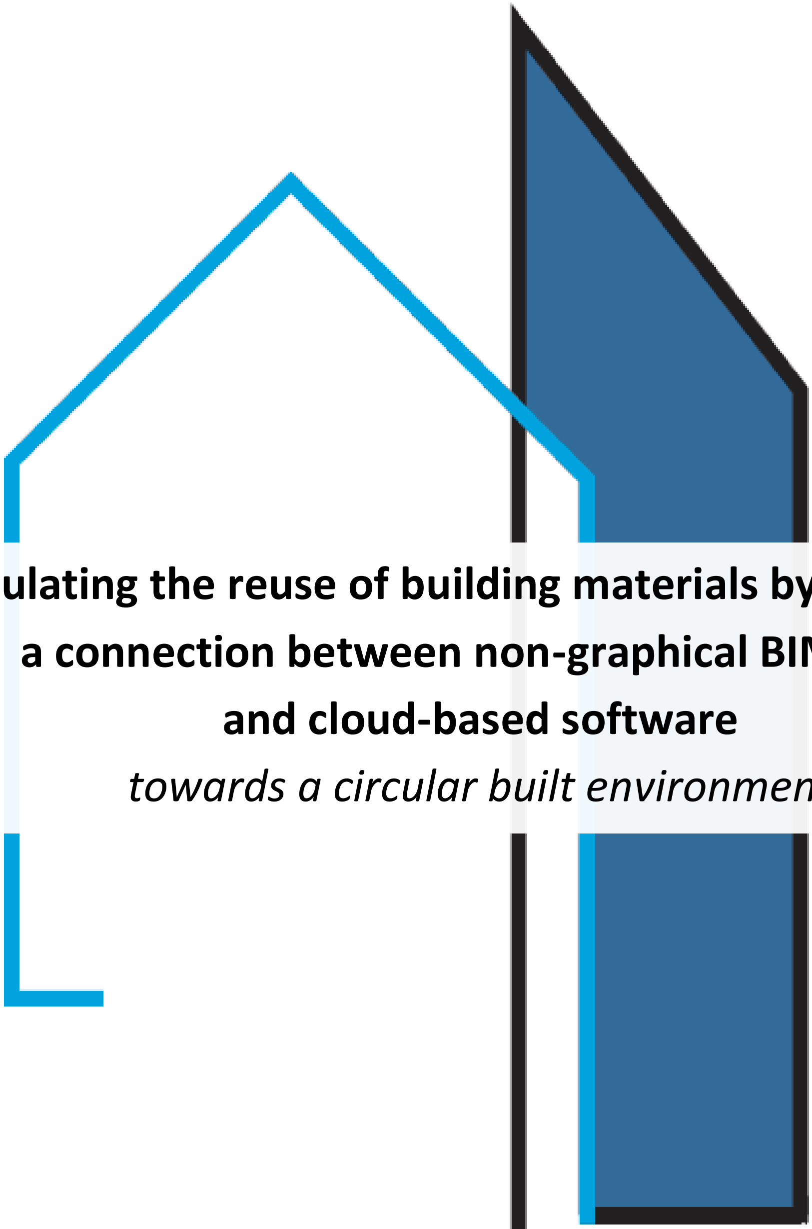
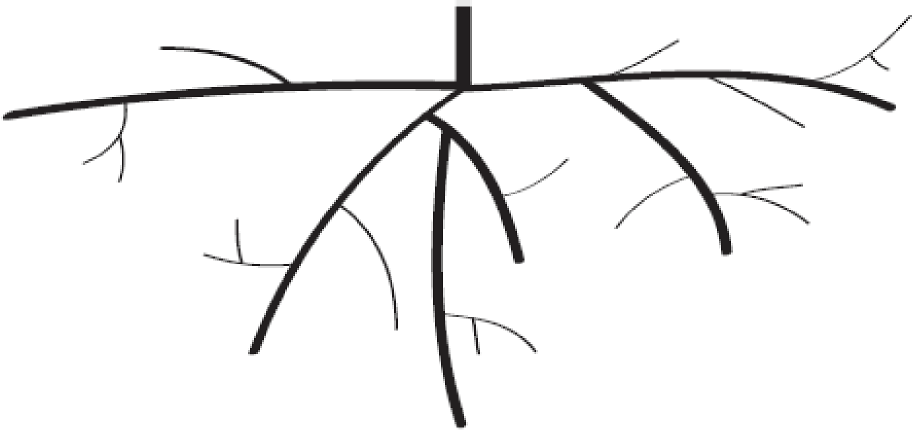


Stimulating the reuse of building materials by developing a connection between non-graphical BIM data and cloud-based software

towards a circular built environment

MSc Construction Management and Engineering
Eindhoven University of Technology

Joran Verhoeven
2020



Colophon

GENERAL

Thesis title:	Stimulating the reuse of building materials by developing a connection between non-graphical BIM data and cloud-based software.
Thesis subtitle:	*Towards a circular built environment*
Date:	11/08/2020
Final colloquium	17/08/2020
Place:	Eindhoven

STUDENT

Author:	Joran (R.Y.M.) Verhoeven
Student no.:	1264583
Email:	r.y.m.verhoeven@student.tue.nl / joran_verhoeven@hotmail.com
University:	Eindhoven, University of Technology
MSc program:	Construction Management & Engineering

GRADUATION COMMITTEE

Prof. dr. ir. B. (Bauke) de Vries
University supervisor | Chairman graduation committee
b.d.vries@tue.nl

Dr. ir. -arch. P. (Pieter) Pauwels
University supervisor | 1st supervisor
p.pauwels@tue.nl

Dr. Q. (Qi) Han
University supervisor | 2nd supervisor
q.han@tue.nl

GRADUATION COMPANY

BASED (BIM management & consultancy)

Ir. A. (Anouk) van Otterlo
Company 1st supervisor | external advisor
anouk@based.co.nl

Ir. M. (Marcel) van Bavel
Company 2nd supervisor | external advisor
marcel@based.co.nl

Technische Universiteit
Eindhoven
University of Technology

< Page intentionally left blank >

Preface

Before my graduation project started, I aimed to conduct a research that would have a contribution to the AECO industry, and especially in its transition towards a circular built environment. Not only for a scientific relevance, but also the societal relevance of the research was important. Our impact on the environment is something we cannot deny, and therefore we must aim to change our mindsets and workflows to change them into circular processes to contribute together in creating a better planet, even if this is done with small steps at the time.

I think this research has great potentials for the circular built environment to be implemented and therefore can be of great contribution. Especially for stimulating the reuse potentials of building materials to reduce the waste generation. Therefore, I am very proud of what I have accomplished during my graduation project. This research finalizes the master Construction Management and Engineering and therefore also the end of my career as a student.

Reflecting on the past six months as well as my student career at the Eindhoven University of Technology, I can say I grew as a person during these years. The opportunity from the University to choose your path of (offered) courses, makes it able to design your personal development during the master. I am very grateful for this chance, because I could follow a variety of courses, which I did. Combining these courses and the knowledge that I gathered, evolved in the final topic for my master thesis. It is a very complex topic, in which a lot of different components could and should work together. Everything I learned in the past years did help me solving issues and scientific problems. This report is a collection of everything I learned during my study, and therefore I look back with pride on all the effort, devotion, knowledge, and positive energy I managed to implement in this very important period of my life.

But of course, this was not possible without the great support from the people I gathered around me for helping me achieving my research goal. First of all, I want to thank Pieter Pauwels for his devoted and strong support before and during the graduation project. This result would never have been possible without your guidance. It was a pleasant collaboration, even though almost the entire process was held online due to the Coronavirus, it was still managed to retrieve a lot of feedback during the project. I am also very grateful to my external supervisor Anouk van Otterlo, who supported me throughout the entire project, always made time for me to discuss ideas and results, which led to valuable insights and helped me in making the right decisions.

I would like to thank my 2nd supervisor Qi Han for the intermediate feedback moments and especially for her contribution to the knowledge of the circular built environment. I also want to thank Marcel van Bavel (BASED, BIM Management & Consultancy) for the chance to collaborate with me on this research topic, the valuable ideas, critical feedback, and support. I also want to thank my colleagues from the company for their input in the research as well as the great time I had with them.

Finally, I would like to thank my family and my girlfriend for always supporting my ambitions and share the same motivation I had for this research and the entire study. Without them, this would never have been possible.

Joran Verhoeven
Eindhoven, August 2020

< Page intentionally left blank >

Table of contents

< Page intentionally left blank >

Summary

The construction sector has a major part in the waste that is generated in the Netherlands. To be able to reduce these numbers, the construction sector tries to be more sustainable by recycling their waste and using recycled building materials. These were the first steps in its transition from a linear to a circular built environment. The construction sector was able to lower the numbers of its waste generation. But recycling is still not the desired feedback loop. The reuse loop has a higher potential and has the lowest negative impact on the environment out of the four technical feedback loops (maintenance, reuse, remanufacture, and recycle). Especially since the Dutch government determined to have a full circular economy by 2050 with only using reuse processes, the reuse potentials of building materials should be stimulated even more. Many initiatives are already introduced, considering the implementation of circular concepts that are concentrated on stimulating reusable building materials. But no initiative, yet, is developed on how to stimulate the reusable building materials for existing buildings.

When implementing this challenge into the Architecture Engineering Construction Operations and Maintenance (AECO) industry, it must be in line with the development that is happening nowadays. The implementation of the Building Information Management (BIM) process increased the collaboration as well as the interoperability possibilities in the AECO industry tremendously over the past years. But since its introduction, the AECO industry is still struggling in how to manage their building data, especially when the AECO industry is shifting towards cloud-based software technologies. Combining the transition towards the circular built environment with the shifting towards cloud-based software in the AECO industry, the following main question was formulated:

"How can the connection between non-graphical BIM data of existing buildings and cloud-based software be realized, in order to stimulate the reuse of building materials to support the circular built environment?"

The first part of the literature review on the circular built environment, did conclude that the reuse cycle loops have the highest potential in a circular economy. To be able to capture the highest value of the reusable building materials, new circular-based concepts were developed. Especially the design phase, which has the greatest benefits to implement circular-based concepts, can contribute to stimulate the reuse of building materials. The circular design, reversible design, and design for disassembly are the three most known new concepts in the circular built environment. Especially the BAMB ('building as materials banks') organization is supporting this new approach of designing a building, that can completely be deconstructed at the end of its life cycle. This enables to capture the highest value of the building materials. When these building materials can be extracted and they do not need maintenance or remanufacturing, they can immediately be reused. To collect the circular building data, the Madaster platform developed material passports, which indicate the circularity of materials in a building, to get insights on the impact the building has on the environment.

The second part of the literature review on the building information management defined the enormous steps the AECO industry took over the years with the implementation of the BIM methods, based on connectivity and data management. The BIM methods enable companies to collaborate on one platform and provide interoperability possibilities to connect different software tools. The 'I' in BIM increased the quality of the formerly 3D models, to not only have graphical data but also non-graphical information available in the BIM models. To be able to retrieve this data, the open standards as the IDM (information delivery manual), MVD (model

view definitions), and the guidelines of the basis-ILS (Dutch: Informatie Leverings Specificatie), stimulate companies to think about the desired output of the project by setting the right requirements upfront. At the same time, the cloud-based software gets introduced in the construction sector. The used standards and exchange file formats have to be more generic, scalable, and highly adoptable, to have the possibility to be connected with cloud-based software. Stated is that especially the JSON (JavaScript Object Notation) file exchange format has high potentials to enable this connection, due to lightweight data exchange format and high parsing efficiency comparing to other exchange file formats. But the connection with the IFC (Industry Foundation Classes), which is the exchange file format that is most commonly used in the AECO industry, is not yet developed.

Based on the conclusions from both parts of the literature review, multiple opportunities arise that can contribute to solve the problem that is defined for this research. The Unique Selling Points (USPs) point out that a platform is desired on which circular-based information of building materials of existing buildings can be collected and its materials can be sold on that same platform as well. This, to accomplish the circular as well as the economic part of the circular economy. With the concepts of buildings as materials banks, the platform can become a library in which existing buildings and their building materials can be presented as material banks. This will evolve in a marketplace where the supply and demand of reusable building materials meet. Combining this with the shifting towards cloud-based software, the following system was proposed: A digital marketplace in the cloud using JSON file format input and Javascript for web development (cloud-based software), with the supply and demand of reusable building materials (circular economy), driven by circular-based non-graphical information (BIM models).

To be able to export the BIM models into single building materials for the cloud-based marketplace, the right non-graphical information must be retrieved from these models. Only the characteristics and properties of the building materials, which are within the circular-based approach of this research, should be retrieved. Therefore, a circularity-based guide was developed which gave insights on which desired circular-based data should be retrieved from the BIM models. This guide is developed as the Re Use Index – ILS and is inspired by the guidelines of the basis-ILS. Circular-based exchange requirements are included in the guide, that aim to attach circular-based non-graphical information to the objects in a BIM model. This, to increase their reuse potentials. The exchange requirements are divided into three topics: object properties, object circularity indicators, and object dismantle indicators. To verify if this data can be extracted from contemporary BIM models, a BIM model data juxtaposition was executed. In this analysis, three BIM models were compared based on matching and circular-based data, that is within the scope of this research. The outcome of this analysis was Excel Spreadsheets, which contain pre-defined parameters, that enable end-users of the cloud-based marketplace to only extract the non-graphical information that is needed to stimulate the reuse of building materials.

The proposed system (cloud-based marketplace) is developed as an online application using the Rapid Application Development (RAD) method. The development was divided into four phases: requirements, design, development, and test phase. The requirements were listed based on the needs of the end-users. The system was designed using an activity diagram to structure the paths of the webpages and drafted scenarios were used to design the first version of the Graphical User Interface (GUI) in the front-end. The Model View Controller (MVC) framework was fundamental in the system design. The MVC framework was implemented using an Application Programming Interface (API), that created the connection between the external database of MongoDB, the web browser, and the cloud-based software (Node.js, Javascript, JSON). In the development phase, the online application was developed based on the requirements and the drafted scenarios. Two different environments were developed in the online application, that distinguishes the different end-users. At the supply side are the project

owners, who have to use the Project 2 Connect side. It enables them to upload their projects, assess reuse potentials, and finally provides the opportunity to add the reusable building materials as products in the webshop. The webshop customers use the Re Use 2 Use side, which enables them to buy reusable building materials and to get more information about the products.

In the test phase, the online application was tested using the Alpha Testing method, in which a real-user environment was simulated to test the online application based on their needs. Optimizations were implemented as well. Finally, the products in the webshop, with the non-graphical information that was retrieved from the BIM model data juxtaposition, was compared with a best-case scenario. An example was developed of a product in the webshop that did contain the desired non-graphical information based on the exchange requirements from the Re Use Index – ILS. It was evident that the best-case scenario did capture a higher value of non-graphical information that increases the reuse potential of building materials, since this circular-based information is not yet included in the contemporary BIM models. To enable the AECO industry with an easier implementation of the circular-based exchange requirements in their processes, is the Re Use Index – ILS developed as being an existing guide. This guide can help stakeholders in project teams to increase the value of circular-based non-graphical information about the objects in the BIM models and therefore stimulate the reuse potentials of the building materials when the building is at its end of life cycle.

The online application did show that software that is used in the AECO industry can be connected to cloud-based software, and that this connection does increase the reuse potentials of the building materials. In this research, it is concluded that the JSON exchange data format in combination with Javascript for web development, has the highest value and potential for realizing the connection between the software used in the construction sector and the cloud-based software. The online application can convert .xlsx files into .json, present the data in the web browser, and sell the objects from the BIM models as products in a webshop. The online application is a real-time prototype, which has positive test results. After some hardcode optimizations, it has realistic potentials to be used in the AECO industry. The online application did realize the desired connection as it is formulated in the main question and the supply and demand in the digital marketplace provides the opportunity for the stimulation of the reuse of building materials. Based on these results, it, therefore, can be concluded that the research goal is accomplished.

Samenvatting

De bouwsector heeft een ontzettend groot aandeel in de totale afvalproductie van Nederland. Om dit aandeel te verminderen probeert de bouwsector meer duurzame methodes te implementeren, door afval te recyclen en zoveel mogelijk gerecyclede bouwmaterialen te gebruiken. Dit waren de eerste stappen in de transitie van een lineaire naar een circulaire bouwsector. Het is de bouwsector hierdoor gelukt zijn aandeel in de afvalproductie te verminderen. Echter is alleen recyclen nog steeds niet het gewenste antwoord. Hergebruik van bouwmaterialen heeft veel meer potentie en heeft de minst negatieve impact op het milieu van de vier technische cyclussen (onderhoud, hergebruik, reproductie en recyclen). Nu de Nederlandse overheid heeft besloten om een complete circulaire economie te realiseren in 2050 met enkel hergebruik processen, moet het hergebruik van bouwmaterialen meer dan ooit tevoren worden gestimuleerd. Vele initiatieven zijn al geïntroduceerd wanneer we kijken naar de implementatie van circulaire concepten die het hergebruik van bouwmaterialen stimuleren. Er is echter nog (steeds) geen initiatief ontwikkeld, dat het hergebruik van bouwmaterialen uit bestaande gebouwen stimuleert.

Wanneer deze uitdaging in de Architecture Engineering Construction Operations en Maintenance (AECO) industrie wordt geïmplementeerd, moet dit in lijn zijn met de ontwikkeling die er tegenwoordig al plaats vindt. Deze ontwikkeling is de implementatie van het Bouwwerk Informatie Management (BIM) proces dat de samenwerking en de interoperabiliteit mogelijkheden in de AECO-industrie de laatste jaren ontzettend heeft versterkt. De AECO-industrie, is echter sinds de introductie van BIM, nog steeds aan het onderzoeken hoe zij hun gebouw data kunnen beheren. Helemaal nu de ontwikkeling van cloud-based softwaretechnologieën voet aan de grond krijgt in de AECO-industrie. Wanneer we de transitie, richting een circulaire bouwsector, combineren met de ontwikkeling richting cloud-based softwaretechnologieën in de AECO-industrie, kan de volgende hoofdvraag geformuleerd worden:

"Hoe kan een connectie gerealiseerd worden tussen niet-grafische BIM-data van bestaande gebouwen en cloud-based software, voor het stimuleren van het hergebruik van bouwmaterialen voor de ondersteuning in een circulaire bouwsector?"

Uit het eerste gedeelte van het literatuuronderzoek kan geconcludeerd worden dat hergebruik van bouwmaterialen de grootste potentie heeft in een circulaire economie. Om de hoogste waarde van herbruikbare bouwmaterialen te behouden zijn nieuwe circulaire concepten ontwikkeld. Met name de ontwerpfase kan bijdragen in het stimuleren van het hergebruik van bouwmaterialen, omdat deze fase de grootste mogelijkheden bevat voor het implementeren van circulaire concepten. Het circulaire ontwerp, omkeerbare ontwerp en het ontwerp voor demontage zijn de drie meest bekende concepten in de circulaire bouwsector. Met name de BAMB ('gebouwen als materiaal banken') organisatie ondersteunt deze nieuwe aanpak in het ontwerp van gebouwen die compleet gedeconstrueerd kunnen worden wanneer deze het einde van hun levensduur bereiken. Dit maakt het mogelijk om de hoogste waarde van de bouwmaterialen te behouden. Wanneer deze bouwmaterialen gewonnen kunnen worden en geen onderhoud nodig hebben, kunnen zij direct hergebruikt worden. Om circulaire gebouwdata te verzamelen, heeft het Madaster platform materiaalpaspoorten ontwikkeld. Deze materiaal paspoorten geven een indicatie over de circulariteit van de bouwmaterialen uit het gebouw, om inzicht te geven in wat de impact is die het gebouw heeft op het milieu.

Het tweede gedeelte van het literatuuronderzoek, over het Bouwwerk Informatie Management, definieerde de enorme stappen die de AECO-industrie heeft genomen door de jaren heen. Dit is gebeurd door de implementatie van de BIM-methoden, gebaseerd op connectiviteit en datamanagement. De BIM-methoden maakten het voor bedrijven mogelijk om samen te werken op één platform en zorgden voor de mogelijkheid tot interoperabiliteit door het kunnen combineren van verschillende softwareprogramma's. De 'I' in BIM zorgde voor een toename van de kwaliteit, ten opzichte van de voormalige 3D modellen. Hierdoor kwam niet alleen grafische data, maar ook niet-grafische data beschikbaar in BIM-modellen. Om deze data te verkrijgen, zijn normen ontwikkeld als de Information Delivery Manual (IDM), Model View Definition (MVD) en de richtlijnen van de basis-ILS (Informatie Leverings Specificatie), die bedrijven stimuleren om goed na te denken over de gewenste uitkomst van de projecten, waardoor de juiste eisen vooraf opgesteld moeten worden. Tegelijkertijd zijn de cloud-based softwaretechnologieën geïntroduceerd in de bouwsector. De gebruikte normen en uitwisselingsbestandstypes moeten meer generiek, schaalbaar en makkelijk toepasbaar zijn, om deze te kunnen verbinden met cloud-based software. Gedefinieerd is dat met name de JSON (JavaScript Object Notation) uitwisselingsbestandvorm de hoogste potentie heeft om deze connectie mogelijk te maken. Dit komt door zijn lichtgewicht gegevensuitwisseling vorm en hoge ontleding efficiency vergeleken met andere uitwisselingsbestandvormen. Echter is de connectie met IFC (Industry Foundation Classes), wat de meest gebruikte vorm is als uitwisselingsbestand in de AECO-industrie, nog niet ontwikkeld.

Gebaseerd op de conclusies uit beide onderdelen uit het literatuuronderzoek, zijn verschillende kansen ontstaan die bij kunnen dragen aan het oplossen van het probleem dat is gedefinieerd voor dit onderzoek. De Unique Selling Points (USPs) wijzen uit dat een platform gewenst is waarop circulaire information de bouwmaterialen uit bestaande gebouwen verzameld kan worden en deze bouwmaterialen op datzelfde platform verkocht kunnen worden. Dit is gewenst om zowel het circulaire als het economische gedeelte van een circulaire economie te volbrengen. Met de concepten als 'gebouwen als materiaal banken', kan het platform een bibliotheek worden waarop bestaande gebouwen en de bijbehorende bouwmaterialen gepresenteerd kunnen worden als zijnde een echte materiaal bank. Dit zal zich ontwikkelen tot een marktplaats voor vraag en aanbod van herbruikbare bouwmaterialen. Wanneer we dit combineren met de verschuiving naar cloud-based software, is het volgende systeem voorgesteld: Een digitale marktplaats in de 'cloud' die gebruik maakt van de JSON-bestandsvorm en Javascript voor webapplicatie ontwikkeling (cloud-based software), voor vraag en aanbod van herbruikbare bouwmaterialen (circulaire economie), gevoed door circulaire niet-grafische informatie (BIM-modellen).

Om het mogelijk te maken dat de BIM-modellen worden geëxporteerd in enkele bouwmaterialen voor de cloud-based marktplaats, moet de juiste niet-grafische information onttrokken kunnen worden uit deze modellen. Echter moeten alleen de kenmerken en eigenschappen van de bouwmaterialen, die binnen de circulaire aanpak van dit onderzoek vallen, onttrokken worden. Daarom is een circulaire gids ontwikkeld die inzichten geeft in welke circulaire gegevens gewenst zijn te halen uit een BIM-model. Deze gids is ontwikkeld als de Re Use Index – ILS en is geïnspireerd op de richtlijnen van de basis-ILS. De circulaire uitwisselingsvoorwaarden zijn in deze gids inbegrepen en hebben het doel om circulaire niet-grafische informatie toe te voegen aan de objecten in BIM-modellen. Dit wordt gedaan om hun hergebruik potentie toe te laten nemen. De uitwisselingsvoorwaarden zijn verdeeld over drie onderwerpen: object eigenschappen, object circulariteitsindicatoren en object demontage indicatoren. Voor het verifiëren van de data die gewonnen kan worden uit hedendaagse BIM-modellen, is een BIM-model data juxtapositie uitgevoerd. In deze analyse zijn drie BIM-modellen met elkaar vergeleken, op basis van overeenkomstige en circulair gerichte data, die binnen het toepassingsgebied van dit onderzoek vallen. De uitkomst van deze analyse zijn Excel Spreadsheets met voor-gedefinieerde parameters, die het de eindgebruikers van de cloud-based marktplaats mogelijk maken om enkel de niet-grafische informatie die nodig is voor het stimuleren van het hergebruik van bouwmaterialen te verkrijgen.

De online-applicatie is ontwikkeld door gebruik te maken van de Rapid Application Development (RAD) methodiek. Het ontwikkelproces was verdeeld over vier fases: eisen opstellen, ontwerpfase, ontwikkelfase en de testfase. De opgestelde eisen zijn gebaseerd op het eisenpakket van de uiteindelijke eindgebruikers. Het systeem was ontworpen door gebruik te maken van een activiteitendiagram om structuur te geven aan de paden van de webpagina's. De geschetste scenario's zijn gebruikt om het ontwerp van de eerste versie van de Graphical Users Interface (GUI) aan de voorkant van de online-applicatie aan te geven. In de ontwikkelfase was het Model View Controller (MVC) patroon fundamenteel voor het ontwerp van het systeem. Het MVC-patroon was geïmplementeerd in de Application Programming Interface (API), die de connectie tussen de externe database MongoDB, de webbrowser en de cloud-based software (Node.js, Javascript, JSON) mogelijk maakte. In de ontwikkelfase is de online-applicatie ontwikkeld op basis van de eisen en geschetste scenario's. Twee verschillende omgevingen zijn ontwikkeld in de online-applicatie, die de verschillende eindgebruikers van elkaar onderscheiden. Het aanbod op de online-applicatie wordt gevuld door de project eigenaren, die gebruik moeten maken van de Project 2 Connect kant. Dit maakt het voor hen mogelijk om projecten te uploaden, eventuele hergebruik potenties te beoordelen en biedt uiteindelijk de mogelijkheid om herbruikbare bouwmaterialen toe te voegen als producten in de webwinkel. De klanten van de webwinkel moeten gebruik maken van de Re Use 2 Use kant van de online-applicatie, die het voor hen mogelijk maakt om herbruikbare bouwmaterialen te kopen en om meer informatie over deze producten te verkrijgen.

Tijdens de testfase is de online-applicatie getest door gebruik te maken van de Alpha Testing methodiek, waarin een 'echte eindgebruiker' omgeving is gesimuleerd om de online-applicatie te testen op basis van hun behoeften. Optimalisaties zijn hierna toegevoegd. Uiteindelijk zijn de producten uit de webwinkel, met de niet-grafische informatie die is verkregen uit de BIM-model data juxtapositie, vergeleken met een best-case scenario. Een voorbeeld is ontwikkeld voor een product in de webwinkel die wel de gewenste niet-grafische informatie bezat, gebaseerd op de uitwisselingsvoorwaarden van de Re Use Index – ILS. Het was duidelijk dat de best-case scenario inderdaad een hogere waarde van niet-grafische informatie bevatte, die ervoor zou zorgen dat de hergebruik potentie van het bouwmateriaal zou toenemen. Dit was met name, omdat deze circulaire informatie nog niet was toegevoegd aan de hedendaagse BIM-modellen uit de analyse. Om voor de AECO-industrie een eenvoudige implementatie van circulaire uitwisselingsvoorwaarden in hun processen mogelijk te maken, is de Re Use Index – ILS ontwikkeld als een daadwerkelijk bestaande handleiding. Deze gids kan de belanghebbende in een projectteam helpen om de waarde van circulaire niet-grafische informatie van de objecten in de BIM-modellen toe te laten nemen. Hierdoor zal het potentiele hergebruik van bouwmaterialen gestimuleerd worden wanneer het gebouw het einde van zijn levensduur bereikt.

De online-applicatie heeft laten zien dat het mogelijk is om software die gebruikt wordt in de AECO-industrie te koppelen met cloud-based software en dat door deze connectie de hergebruik potenties van bouwmaterialen toeneemt. In dit onderzoek is geconcludeerd dat de JSON-uitwisselingsbestandsvorm in combinatie met Javascript voor webapplicatie ontwikkeling, de hoogste waarde en potentie heeft voor het realiseren van een connectie tussen software die gebruikt wordt in de bouwsector en cloud-based software. De online-applicatie kan .xlsx bestanden converteren naar .json bestanden, deze presenteren in de web browser en de objecten uit de BIM-modellen verkopen als producten in een webwinkel. De online-applicatie is een bestaand prototype met positieve testresultaten. Nadat er een aantal optimalisaties plaatsvinden in de codes, heeft de online-applicatie realistische potentie om toegepast te worden in de AECO-industrie. De online-applicatie realiseert de gewenste connectie zoals die is geformuleerd in de hoofdvraag en de vraag en aanbod in de digitale marktplaats draagt bij aan de kansen voor het stimuleren van het hergebruik van bouwmaterialen. Gebaseerd op deze resultaten kan daarom geconcludeerd worden dat het onderzoeksdoel is bereikt.

<   Page intentionally left blank   >

Abstract

The need for stimulation the reuse of building materials in the construction sector is growing. Since the construction sector has a major part in the total number of waste that is generated in the Netherlands, it tries to be more sustainable over the years. But since the reuse feedback loop is defined as the highest potential over the recycle feedback loop, the construction sector needs to focus on the reuse loops to reduce its waste production. This is in line with the vision of the Dutch government to have a fully circular economy, with only using reuse processes by 2050. The implementation of the Building Information Management (BIM) process increased the collaboration and interoperability potentials within the Architecture Engineering Construction Operations & Maintenance (AECO) industry. But it is now shifting towards the usage of cloud-based software technologies. To enable the AECO industry to realize a connection with cloud-based software and with the need for the stimulation of the reuse of building materials, this research aimed combine both problems. By realizing a connection between non-graphical information and cloud-based software and use this connection to stimulate the circular built environment to reuse its building materials. This research is concentrated on existing buildings, since these will be deconstructed soon and have the highest potential for the extraction of reusable building materials.

This research proposed an online application that was developed as well as a guide (Re Use Index – ILS; inspired by the basis-ILS) in which circular-based exchange requirements are listed. The guide can help de AECO industry to attach circular-based non-graphical information to the objects in the BIM models. The online application was developed using an MVC (Model View Controller) framework, which was fundamental in the connection between Node.js, Javascript, JSON, MongoDB, Application Programming Interface (API), and the web browser. The online application is a digital marketplace for reusable building materials, which is in line with the circular as well as the economic part of the circular economy. The online application enables project owners to upload their projects and to add the building materials as products to the webshop. The webshop customers on the other hand can buy these building materials. With the Re Use Index – ILS guide, for increasing the circular-based non-graphical information in BIM models and the online application where the supply and demand of reusable building materials meet using cloud-based software, it is concluded that the research goal is accomplished.

Keywords
Reusable building materials, waste reduction, BIM, cloud-based software, non-graphical information, MVC framework, API, online application, circular-based exchange requirements

List of Abbreviations

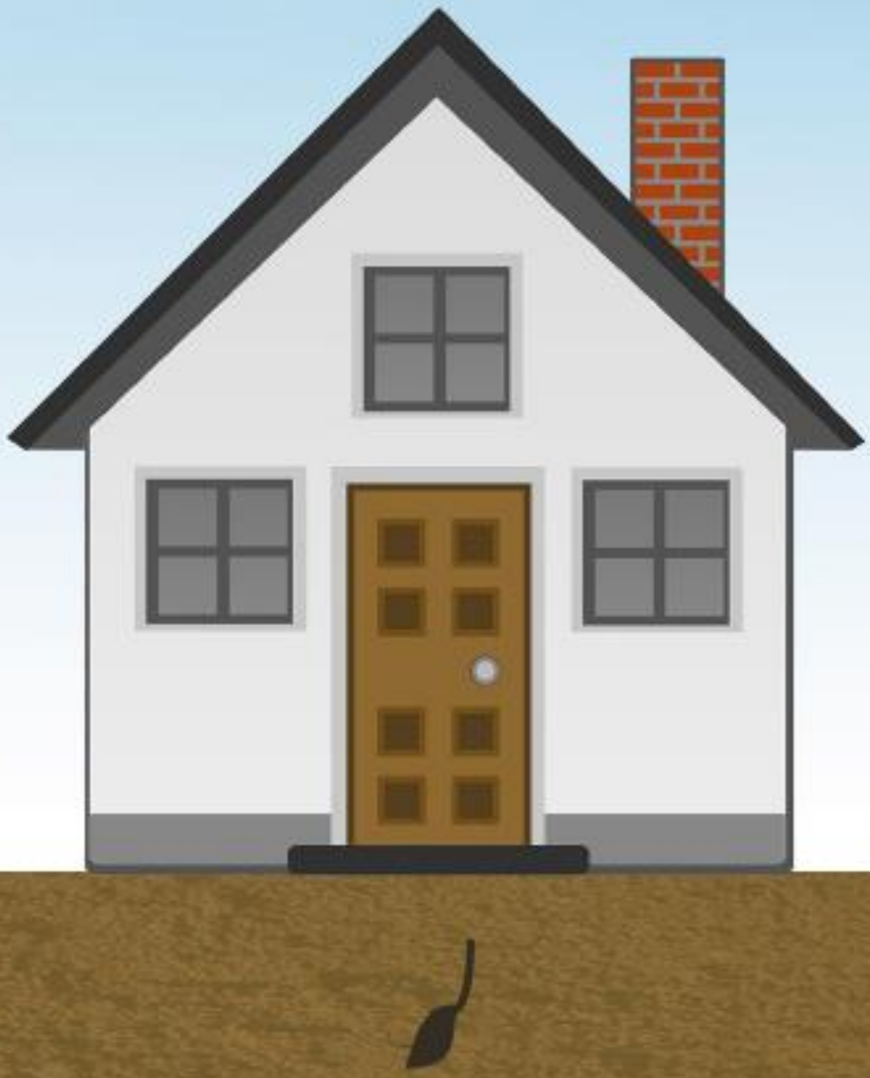
AECO	Architecture, Engineering, Construction, Operations
API	Application Programming Interface
BAMB	Building as Material Banks
BCI	Building Circularity Indicators
BIM	Building Information Modelling
BMC	BIM-based Model Checking
CAD	Computer-aided Design
CEN	European Committee for Standardization
CEPs	Circular Economy Principles
CSS	Cascading Style Sheets
DfD	Design for Disassembly
EDA	Event-Driven Architecture
EJS	Express.js
GUI	Graphical User-Interface
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
LCA	Life Cycle Assessment
iBIM	Integrated Building Information Modelling
IDM	Information Delivery Manual
IDS	Information Delivery Specification
IFC	Industry Foundation Classes
ILS	Informatie Levering Specificatie
IP address	Internet Protocol adress
ITO	Information Take Off
JSON	JavaScript Object Notation
LOD	Level of Detail
LOI	Level of Information
LOIN	Level of Information Need
MVC	Model – View - Controller
MVD	Model View Definition
NL-SfB	Nederlands Samarbetskommittén för Byggnadsfragor
NoSQL	Not Only Structured Query Language
PoC	Proof of Concept
RAD	Rapid Application Development
RDF	Resource Description Framework
SQL	Structured Query Language
URL	Uniform Resource Locator
USP	Unique Selling Point
VSC	Visual Studio Code
XML	Extensible Markup Language

List of figures

List of tables

Section 1

Introduction



1 | Introduction

In this first section, the introduction of this research will be described. The research motive will be defined in sub-chapter (1.1). Here, the challenges where the construction sector is coping with, are defined. These challenges evolve in a problem, which is defined in sub-chapter (1.2). To be able to tackle this problem, the main question and sub-questions are formulated in sub-chapter (1.3). The research design, as well as the schema of the research design, are described and depicted in sub-chapter (1.4). This section will be finalized by defining the relevance of this research in sub-chapter (1.5).

1.1 | Research Motive

The construction sector has a major part in the production of waste. In 2016, in the Netherlands only, nearly one-quarter of all the output generated by the construction sector consisted out of waste (Statistics Netherlands; CBS, 2019). This is higher than any other sector in the Netherlands. Almost half of the total waste production in the Netherlands is on account of the construction sector. And this number does not include the demolition waste from existing buildings (approximately 11 billion kg) (Statistics Netherlands; CBS, 2019).

This a serious problem. The construction sector should drastically stop continuing with these numbers of waste generation. And luckily, they are trying to reduce these numbers. The construction sector started many years ago, to slowly implement sustainability in their processes. The implementation of sustainability goes in line with the noted negative impact that the construction sector has on the environment (Graham, 2002; Grierson, 2009; Opoku, 2015). Especially in the last years, the sustainability processes are widely implemented in the construction sector. In the Netherlands, almost 54% of all recycled materials is used within the construction sector, and nearly 38% of the used materials in the building sector ended up in the recycling process (CBS, 2019). The construction sector did recognize their part in the waste production and is now trying to reduce these number rapidly by closing their life cycle loops. By closing the life cycle loops, the construction sector is changing from a linear to a circular built environment (Ellen MacArthur Foundation, 2016). The generated waste and used building materials do not end up on the landfill or in the waste production processes, but they get a new purpose.

But the ultimate sustainable construction sector, does not end with the recycling processes, because recycling is still not the most environmentally friendly answer. Recycling contains a lot of transportation and reproduction processes, which still has a negative impact on the environment (Ng, Chau, 2015). Even though it is a closed-loop, it is still not the ideal solution. The solution is to create a shortcut where the reproduction process can be excluded from the life cycle loop and the building materials can immediately be reused. In the circular built environment, this is named: reusable building materials. The reuse of building materials is a far more beneficial method than the recycling method. Existing buildings contain a wealth of building materials that can directly be reused after the building gets deconstructed (Akanbi, et al., 2018). Over the years, the construction sector set a clear vision to develop a less negative impact on the environment. With the implementation of sustainability, the introduction of the circular economy and the reduction of waste production due to the change to recycling life cycle loops. The next step is to change to an even more efficient and environmentally friendly approach with the introduction of the circular built environment: increasing the reuse of building materials in the complete life cycle of a building project.

This is in line with the vision of the Dutch government which has set clear objectives about the circular economy in the Netherlands. Their first objective is to make the production process more efficient in which fewer natural resources are needed. The second objective is when it is inevitable that natural resources are needed, this must be extracted from renewable resources. And the final objective is to develop new production processes with the implementation of circular-based products. The Dutch government aims to get a fully circular economy in the Netherlands in 2050, in which no waste is generated, and every process involves reusable resources (Rijksoverheid, 2019).

Besides the transition from a linear to a circular economy, another development is happening. Since the introduction of mobile phones, we can reach the entire world just with a touch by our fingertips on a screen. The world, as we know it nowadays, is about connectivity due to cloud-based software, applications, and the world wide web. It does not matter where you are, you are always connected to anyone from anywhere you want. You can call, post, buy and sell anything, and it just works in the cloud. In just decades, we are connected like never before (Fabian, et al., 2015). This digital era is taking tremendous steps, even harder than most people can keep track of. Data and connectivity are the main topics of this century. Data management, data streams, data structures, making connections between software, apps, and working in the cloud, it is everywhere around us. Every sector around the world is continuously asking themselves which steps to take, which directions to follow, which opportunities lie ahead.

This development also affected the construction sector. The construction sector is continuously implementing, testing, and struggling with the newest digitisation. This, to gain the greatest benefits out of it, and to keep track of this fast digitisation. The connectivity within the construction sector did already largely increase with the implementation of the building information management (BIM) process (Smith, 2014). The traditional 2D and 3D models made room for BIM models, rich in information that goes beyond just the graphical aspect of the model. The BIM model concentrates more on the wealth of information that can be connected to the 3D visualization as well. The implementation of BIM takes the Architecture Engineering Construction Operations & Maintenance (AECO) industry, to higher levels. Especially the collaboration and interoperability possibilities increased enormously (Arayici, et al., 2018). Different software tools, which are connected by open standards and open file formats, make it possible to exchange BIM models within project teams.

The challenge is to connect the digitisation of the AECO industry with the shifting to cloud-based software and data management systems, such as Dropbox, Google Drive, One Drive, Office 365 etc. Within the cloud-based software, data can easier be stored, files can easier be exchanged due to a lower amount of file sizes and the data is always accessible. This requires from the construction sector to extend their collaboration processes and their software tools, to a cloud-based collaboration approach. The data storage and exchange of data, through OpenBIM standards, will in the future be executed on cloud-based BIM platforms (Juan, Zheng, 2014; Ma, Sacks, 2016).

1.2 | Problem Definition

What can be stated is that a lot of developments are happening within the construction sector nowadays. The construction sector is trying to create an environmentally friendly sector that does not live by the 'take-make-waste' disposal but develops a circular life cycle. With 'recycling' as the main purpose in the last years, this is now changing to concentrate more on the reuse possibilities. The reuse of building materials will be the goal within the circular built environment. And this new approach already is gaining ground. New concepts are invented, such as a circular design, reversible design, and the design for disassembly. These concepts provide the opportunities to deconstruct the complete building and still secure the value capturing of the building materials, since the project is designed so that this is made possible (BAMB, 2019; Fifield & Medkova, 2016; Rios, et al., 2015). With the implementation of these concepts, building materials can very easily be reused. The only problem with these concepts is that they are solutions for future projects. The concepts are used in the design phases of new projects, which will be constructed in the future.

However, there is a stock of existing buildings around the world that are not designed to be deconstructed and reused, but finally will also reach the end of their life cycle. In the past 20 years, in the Netherlands only, almost 59,2 thousand hectares were built for living and working (Statistics Netherlands; CBS, 2019). These buildings contain a wealth of building materials that can easily be reused in other projects, but people must be aware that these building materials are available to be reused. The question is: how can we stimulate the reuse possibilities of the building materials of the existing buildings that are built over the last decades?

To be able to answer this question, it is wise to concentrate on the building data that might be available from these buildings. This affects the other development that is happening within the construction sector, based on connectivity and data management. The implementation of the BIM methods increased the building data management for the AECO industry, to add not just graphical, but also non-graphical information to the 3D models. The BIM models could contain a wealth of information about the building materials used to construct the projects. The dimensions, characteristics, property sets, etc., it all is described and defined during the design phase of the projects. To stimulate the reuse potentials of the building materials after a building has reached its end of life cycle, this data must be made available. To make this data available, a connection has to be developed that retrieves the requested data from the BIM models and immediately present it in a way that people can find these building materials. The shifting towards the cloud-based software could be an opportunity to create this connection.

1.3 | Research Questions

To achieve the problem that is defined, the following main question is formulated:

"How can the connection between non-graphical BIM data of existing buildings and cloud-based software be realized, in order to stimulate the reuse of building materials to support the circular built environment?

To be able to answer the main question, this research is divided into two parts: the theoretical part and the practical part. For both parts, sub-questions are formulated, that contribute in answering the main question:

Theoretical Part

1. What is a circular built environment and how does this affect the AECO industry?
2. How is the building data managed within a BIM model and which standards increase the reuse potentials of non-graphical information?

Practical Part

3. Which matching building materials characteristics can be retrieved from the BIM models, which will increase the reusability potentials of the building materials?
4. What requirements must be met, when creating the possible transfer between the non-graphical BIM data and the cloud-based software?
5. What pattern can be designed that connects the cloud-based software with a user-friendly front-end?
6. How to design a system that involves different stakeholders and end-users?
7. Which circular-based approaches for existing buildings, conducted from the literature review, can be applied to the online application?
8. Can a proof of concept be developed that meets the requirements of the stakeholders and end-users, and/or does it need optimizations after it is validated?

1.4 | Research Design

This research consists of eight sections, as they are highlighted in the research design model (Figure 1). The first Section 1 (Introduction) is already discussed in previous sub-chapters. The research motive and context are stated, the problem is defined, and the research questions are formulated. The main and sub-questions will be answered during the execution of this research.

The next Section 2 (Circular built environment) is the first out of three sections of the theoretical part. Section 2 described the change from a linear to a circular built environment, the implementation of circular-based processes, and the future concepts and initiatives of the circular built environment. This literature review will be extended with Section 3 (Building information management). In this section, the implementation of BIM in the AECO industry, the standards and file formats that make the interoperability and collaboration between companies possible, the distinction between graphical and non-graphical BIM information and the openBIM standards and OpenBIM principles are described. Finally, the step towards cloud-based software is discussed as well. After the literature review is conducted, it is time to frame the research and develop the conceptual model for the next part of the research. This is defined in Section 4 (Methodology). This section describes the research design for the practical part of the research, which is necessary to reach the objectives of this research. The Methodology describes what is needed to fulfil the aim of this research: to stimulate reusable building materials by creating a connection between non-graphical BIM data and cloud-based software. This will be the end of the theoretical part.

The second part of the research is the practical part, which starts with Section 5 (Non-graphical information analyses). In this section, an analysis will be conducted based on which circular-based non-graphical information is desired to retrieve from a BIM model to be able to stimulate the reuse of building materials. Next to this analysis, three BIM models will be juxtaposed and the matching non-graphical information which is relevant for this research will be analysed. This analysis will be compared with the desired circular-based non-graphical information output. After the data is analysed and it is clear which non-graphical BIM data can be retrieved from the BIM models, the connection will be designed in Section 6 (System design). First, the requirements that are needed for the connection will be set. Second, the connection will be visualized with a design which consists of drafted scenarios. Finally, the connection will be developed based on the requirements and drafted scenarios. In Section 7 (Proof of concept & validation), the prototype will be tested based on the alpha testing method. During the test, it must be proven that the prototype fulfils the objectives of this research. Optimizations will be adjusted if failures and errors occurred during the test. The prototype will be analysed based on a best-case scenario that consists of the circularity information delivery specification. The section will end by validating if the prototype is usable, scalable and attractive to be launched on the market. This will be the end of the practical part.

The final Section 8 (Conclusion & discussion) reports the conclusions, discussions, and recommendations for future projects. If the reader wants to consult the references, these are listed Section 9 (References). As well as the extra information that attached to this report in Section 10 (Appendices).

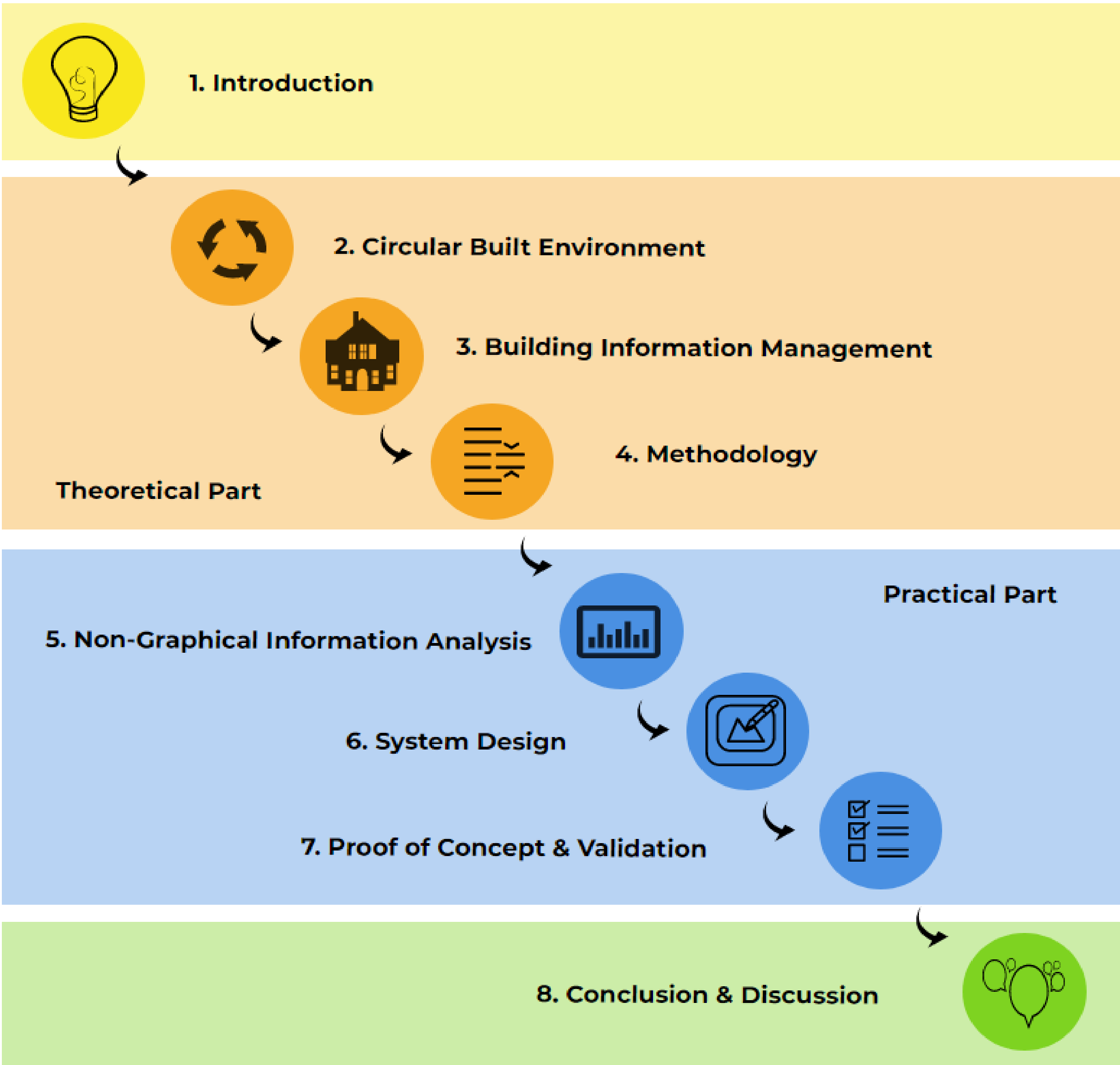


Figure 1: Research Design Model

1.5 | Relevance of The Research

The relevance of this research is stated on different levels. The first level is the contribution to the construction sector in its transition towards a circular built environment. The relevance in creating the awareness of the fact that the reusable building materials are the best possible solution to develop circular economy life cycle loops, which have the least negative impact on the environment. The relevance is concentrated on the change from recycling loops to reuse loops and the reuse potentials the building materials have when a building is at its end of life cycle. Therefore, the view on buildings will change to consider buildings as materials banks. The stimulation of reusable building materials is the easiest solution to reduce waste production in the construction sector. This is in line with the clear vision that has been set by the Dutch government to be fully circular in 2050 in an economy where no waste is generated, and only reusable resources are used. This research will support this vision and will contribute to achieve the objectives that has been set.

The second level is the awareness of the opportunities that can be found in the BIM data could be available. With the implementation of BIM, the non-graphical information of the models within the AECO industry increased. This data has far more opportunities than the construction sector realizes. Therefore, the connection that will be developed will contribute to give the construction sector more insight in the possibilities that can be retrieved from the possible available BIM data.

The third and final level for the relevance of this research is to bring the construction sector in contact with people that are willing to buy the reusable building materials, that will contribute to a cleaner environment. But they do not know yet how to get access to these building materials or even know what potential these building materials have, once they are collected from deconstructed buildings. The contribution to the market as well as the usability of the prototype, that will be developed, will increase the usefulness of creating the connection between both parties.

< Page intentionally left blank >

Section 2

Circular Built Environment

2 | Circular built environment

In the first part of the literature study, the circular built environment is reviewed. In the first sub-chapter (2.1), the difference between the linear and the circular economy is stated, and it is defined what the circular economy is. The circular economy has a few principles where it aims for (2.2) based on primary and secondary production processes. What distinguishes the circular economy from the linear economy is the cycle loops, which are discussed in sub-chapter (2.3). In the next sub-chapter (2.4), the connection between the circular economy and the construction sector is described, in which the circular built environment is described in sub-chapter (2.5). Several concepts are invented to create a circular built environment which can be implemented in the design phase of a construction project. The circular design (2.6) and the design for disassembly (2.7) as well as other initiatives (2.8) to create a circular built environment are reviewed as well in this section. The last sub-chapter (2.9) describes an assessment tool that is able to score a building based on its circularity potentials. The findings of this section will be defined in sub-chapter (2.10) which consists of the conclusion.

2.1 | Linear vs Circular Economy

The world is changing from a linear economy to a circular economy. A linear economy can be defined as a one-way system. Materials and products are extracted from natural resources and will turn into waste after usage. This has a big negative influence on the environment. The waste production evolves into littering and pollution and the natural resources get exhausted. Therefore, changes must be made to protect the environment. By creating a circular economy, these problems may be counteracted, and new opportunities may arise. It rather restores the damages which are made by extracting natural resources, and meanwhile ensures that less waste is generated during the production of new products (Murray et al., 2015). The circular economy is concerned with decreasing the negative impact on the environment. By implementing three 'R' principles in human activities: reduce, reuse, and recycle, the materials will be maintained at their highest utility and value (Ellen MacArthur et al., 2015b). Various actions can be considered during this process: the recycling of packaging, to promote the usage of ecological products, to reduce waste production and the CO_2 emissions, energy-saving opportunities, eco-design, and the use of low-environmental-impact consumer goods (Scarpellini et al., 2019). As the paper by Walter (2016) defines: "A circular economy would turn goods that are at the end of their service life into resources for others, closing loops in the industrial ecosystems and minimizing waste. It would change economic logic because it replaces production with sufficiency: reuse what you can, recycle what cannot be reused, repair what is broken, remanufacture what cannot be repaired" (Walter, 2016). This is the mindset that is the fundamental aspect of the circular economy. Exclude the word waste out of society and find opportunities in creating closing loops.

2.2 | Circular Economy Principles

The circular economy can be distinguished over a set of multiple principles. These principles are clarified by the study of Trevor et al. (2017): reuse at product level (such as 'repair' or 'refurbishment'); reuse at component level ('remanufacturing'); and reuse at material level ('recycling'). These principles are also known as 'secondary production'. This study tells us that building materials consist of three levels: product, component, and material. By order of the reuse possibilities of the building materials, the reuse potentials are divided over these three levels. The secondary production activities are key in the circular economy, because they support the reuse of building materials instead of using only brand-new ones. The secondary production activities are focussed on value capturing rather than producing new building materials. This, therefore, reduces the exhaustion of natural resources, because the extraction of raw materials from these resources is not needed anymore. With the secondary production activities, the value of the reusable building materials can be captured. Just with some small adjustments, the reusable building materials can be added to the product chain again. During the recycling, repairing, remanufacturing, or refurbishment, the value capturing of building materials is important, they must always return to their original state. The secondary production activities make it possible that building materials can return to the product chain without loss of quality.

Besides creating opportunities for the reuse potentials of building materials with the secondary production activities, the first production processes can also implement circular economy principles. Again, three different principles can be distinguished: deconstructable building materials, environmentally friendly production processes, and the use of biodegradable materials. The first principle has the aim to design the building materials in a way, that they can be deconstructed from the building without damaging or loss of quality. The less the building material gets damaged, the less secondary production activities must take place to return the value of the building material to its original state. The second principle aims that, during the production, it is ensured that no harmful substances are emitted. It must be a clean and environmentally friendly process, with a low amount of energy usage and a low rate of CO_2 emissions. Also, the emission of toxic substances must be eliminated. The third circular economy principle, based on first production activities, is the use of biodegradable materials. If the raw materials can be broken down by nature, the building materials will not (partly) remain in the environment, which makes them environmentally friendly (Studios, 2019).

2.3 | Circular Economy Cycle Loops

Within the circular economy, two different cycle loops can be distinguished: the biological cycle and the technical cycle, as is depicted in Figure 2. The difference between these two cycles helps to understand the durability and value capturing of materials. The biological cycle materials, as is clarified by the study of Piscicelli et al. (2016), are designed to re-enter the biosphere and built natural capital, as the technical cycle materials are designed to circulate in closed loops without entering the biosphere. Technical materials, based on fossil fuels, plastics, and metals, are hard to reproduce and have a limited availability. The management of the stock of these materials is an important process in the technical cycle. These materials are only used instead of being consumed, in a circular economy. After the materials have been used, the original value is recovered from residual flows. Biological materials, such as wood, food, and water, can always again be included in the eco-system and be re-generated. It is important to not disturb the eco-system and let it do its work as well as possible. As long as toxic substances do not contaminate the streams and ecosystems are not overloaded, the consumption during this cycle may take place. In that way, renewable organic raw materials can then be regenerated (Het Groene Brein, 2019).

The circular economy system diagram, which is also known as the butterfly diagram because of its shape, is a feedback circle diagram. By providing these feedback circles, the loops can be seen as closing loops, which makes them circular. As mentioned before, the two loops vary from each other but do have the same purpose within their feedback loops. This diagram aims to rebuild and restore the natural capital, loss of natural resources is excluded since they can return into the system, because of the loops. The biological cycle is regenerative and focuses on organic materials that can return to the ecosystem. The technical loop on the other hand is divided over four loops in terms of the feedback process: (1) recycling; (2) remanufacturing; (3) reuse; (4) maintenance. These four loops are restorative, which means that the products and materials retain their integrity and highest utility at all times.

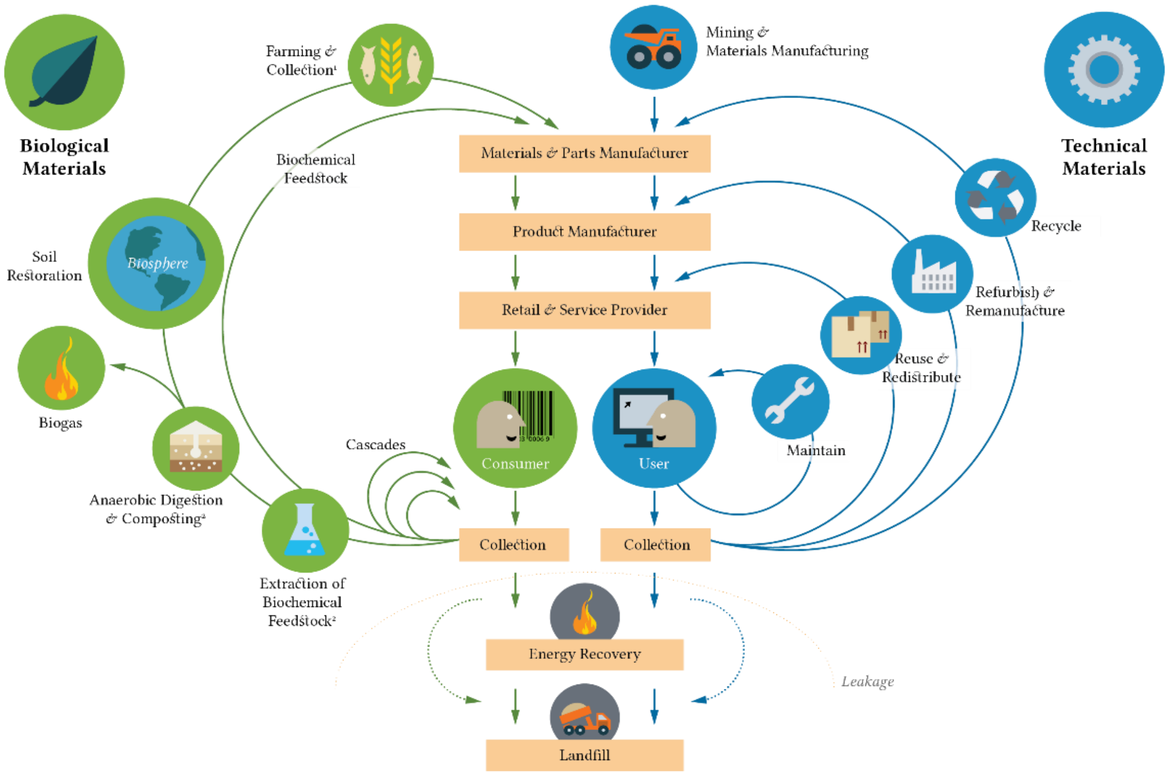


Figure 2: Circular Economy System Diagram (Source: Ellen MacArthur Foundation, 2015)

The first loop is the (1) *recycling loop.* This is the loop, which is most often used in the past years, and therefore is still part of the picture. The loop can be seen as the loop of last resort. When you take a product and return it to the original material of which it is made of, you strip away all the extra value, whether this is labor, energy, or other materials, that were part of the process creating the product in the first place. Therefore, this is the loop of last resort, since only the original material used as input for production will be left over. The product's material, for example, synthetic or minerals will be separated, collected, processed, and inserted into the production process of new products. The materials may, therefore, be of lower (downcycling) or higher quality and functionality (upcycling) than the original material (Braungart et al. 2007; King et al. 2006; EMF 2012; in Lüdeke-Freun, et al., 2018) But going back to the core of the product is not always necessary and/or needed anymore. Therefore, better options are one of the three remaining loops.

In the (2) *remanufacturing loop* the product will be restored to its new or even better standard than when it was produced in the first place. If a product flows through the remanufacturing loop, parts of the product that are failing or expected to be failing soon will be replaced. The remanufacturing of the products ensures the value capturing of the products through the replacement of the main components of the product. The performance of the remanufactured products will be the same as after its first production, based on the original specifications of the manufacturer (Lüdeke-Freun, et al., 2018).

In the (3) *reuse loop* not much is added in sense of energy, materials, extra labor, in terms of returning that product into the economy. The definition of the reuse process is about people selling their old possessions, to directly be reused. As Lüdeke-Freun et al. 2018 clarifies, the products will again be used for which they were originally designed and produced, with little enhancement or change. In commercial settings, ownership is usually transferred from the initial users to the secondhand user.

The last technical cycle loop is the (4) *maintenance loop*. The maintenance process is really valuable for a product. The product does not need a full remanufacturing process, but only parts will be replaced. With little effort, the lifecycle of the product can be extended which ensures the product will last longer in the value chain. The activities involved in the maintenance process will be done by restoring it to its usual functionalities (Lüdeke-Freun, et al., 2018). This loop is also part of the technical cycle, but mainly takes place while the building materials are still part of the building. Hence, the maintenance loop will flow into the remanufacturing loop when the building materials are deconstructed from the building. This loop is the shortest one and will mainly be executed by the owner of the building.

The four technical cycles are described and defined. It is now important to look at them critically and evaluate which cycle is best (meaning: most environmentally friendly). The *remanufacturing* and *recycle loop* can be classified in the same division. This is because they both need secondary production activities while restoring their value. The *remanufacturing* and *recycling loops* do have a positive influence on the environment, and they do protect the natural resources, but it is important to note that this impact is still more negative than the *reuse* or *maintenance loop*. The secondary production activities make it inevitable that energy consumption, use of water, and CO_2 emissions are involved during the execution of these cycles (Rios, Grau, 2020). Recycling products and returning them to their core material, and remanufacturing products to restore their original quality, creates a negative side effect in this circular economy loop. The *maintenance loop* also has this same negative side effect, since the value of the building materials on component or material level must be restored, using new produced components or materials. This again involves the extraction of natural resources and production pollution. Therefore, the *reuse loop* can be seen as being most environmentally friendly in the circular economy feedback cycle loops (Akanbi, et al., 2018). The building materials can immediately be reused without the need for secondary production activities. Now this conclusion is stated, this research aims to focus on the *reuse loops* to get the highest, environmentally friendly result.

2.4 | Circular Economy in the Construction sector

The construction sector is a large consumer of primary raw materials. In the Netherlands, 40% of the extraction of natural resources is used for the construction sector. Almost 50% of the waste that is generated in the Netherlands is related to the architecture, engineering, construction, operations and maintenance (AECO) industry (Landman, 2017; CBS, 2019). The waste generation evolves in a big problem in which the answer lies in the circular economy. The end-of-life of a building contains more opportunities than the AECO industry realizes. Therefore, the perspective that a building is just a building will change into the concept that a building is a stock of materials and products. According to the paper of Zimmann et al. (2016), the construction sector recognizes the need to fundamentally evolve their processes, components, and systems it utilizes to obviate waste and increase efficiency. There is an incredible breadth of opportunity that this will create across the entire supply chain. The construction sector is still literature studies that tell companies how to change to a circular business model. But the reuse and the return of products slowly evolve in regulations which describe and stimulate companies to create these objectives in their processes. Some companies already started designing inventive products that can be reused or be remanufactured. The reuse and recycling of building materials provides opportunities for the change in demolition of buildings as well as for the reuse opportunities for future projects. Another opportunity is sustainable building material replacement in existing buildings. The biggest part of the real estate stock has a low energy label. With the transition to a circular building economy, materials with lower energy usage and a lower amount of CO_2 emissions during the production process can replace materials and products with a low energy label. The philosophy of the circular economy in the construction sector is not only about the closing loops, to eliminate the waste production by reusing materials and recycling resources, but it is also about extending the lifecycle loops of materials and products. By developing materials with a high life expectancy, it also contributes towards a circular built environment. The circular economy approach within the construction sector can be defined as:

"*A lifecycle approach that optimizes the buildings' useful lifetime, integrating the end-of-life phase in the design and uses new ownership models where materials are only temporarily stored in the building that acts as a material bank*" – Leising et al., (2017)

2.5 | Circular built environment

With the change from a linear to a circular built environment, the whole project cycle changes significantly. Every phase of the project requires changes based on the new circular approach. An important aspect of the circular built environment is that the life expectancy of the buildings will be extended in comparison to the buildings constructed in a linear construction sector. The phases of a project in the circular built environment are depicted in Figure 3. The circular approach in the design phase can be distinguished over two levels: on the design level and on building material level. The design level (circular design) is more in-depth described in sub-chapter (2.5), here we discuss the circular design approach based on building material level. This approach aims to design with circular building materials. This can on one hand be building materials that went through the feedback loops (maintenance, recycle, remanufacture or recycle), and on the other hand, newly produced building materials with a circular-based approach. Both approaches are circular: the first to design with used building materials in order to reduce the exhaustion of natural resources, the other one to use environmentally friendly building materials. That can be biodegradable building materials, with a low influence on the environment in the production process or have a high life expectancy. In the construction phase, the aim is to prevent the production of waste as much as possible. This is also possible by still using wrong ordered building materials due to design failures. And if this is not possible, the building materials will be reused in other projects. The waste that is generated, will be assorted on the construction site itself based on its core component (paper, metal, rubble, rubbish, etc.). When the building is in use and therefore in the operation phase, the maintenance of the building materials has the priority. The use of circular-based building materials will evolve in this phase in less maintenance than in a building constructed in the linear economy. The building materials have a high life expectancy, which makes them less sensitive for damaging or depreciation over time. If the building materials need maintenance, it should not be much effort to capture their value again, since the building materials are designed in a way that the components and materials can easily be replaced. In the renovation phase, the circular-based approach aims to avoid unnecessary demolition, and if deconstruction is possible, the building material should be detached from the surface without loss of quality. The final phase is the demolition (deconstruction) of the building. Which is similar to the deconstruction in the renovation phase.

Figure 3: Lifecycle model implementation - circular economy in the built environment (Source: Ministry of Environment, 2019)

Deconstructing for keeping the value maintained. After deconstruction, three options are possible before entering a new circular cycle: reuse, remanufacture or recycle. By executing one of these three cycles, the building materials can be used in future projects (Ministry of Environment, 2019).

2.6 | Circular Design

A new concept in the circular built environment, as is briefly mentioned in the previous sub-chapter, is the circular design. A circular design, as Ellen MacArthur (2012) defines, requires a widening focus from the product to material flows, production processes, and conditions, as well as aspects of use and reuses. It needs an extended system view as well as a profound understanding of ecological principles. The improvements are materials' selection and product design, standardization, modularisation of components, purer material flows, and design for easier disassembly. In this design, resources are continuously cycled in various forms. The construction sector has failed to implement design strategies that prevent the demolition of buildings at their end of life, and therefore did not prevent the generation of waste. The scope of the circular design as is previously mentioned, to design with reused, remanufactured, and recycled building materials on one hand or design with circular-based environmentally friendly building materials. But discussions arise with these approaches, which one is best? Saving natural resources has at all times the main priority. But some natural resources do contain raw materials that are necessary in the production process of building materials with a high life expectancy. The paper of Crus Rios and Grau (2020) mentions about this discussion: "It has been pointed out that the current material resource use rates in the construction sector, paired with global population growth, and consumption trends, will result in the scarcity of certain materials, such as steel and copper, which are prevalent in the construction sector. Steel and copper are highly durable materials, e.g. beams and columns, which implies that many opportunities for their reuse exist in multiple projects" (Crus Rios, Grau, 2020). Taking a critical look at this discussion, the two sides of this story is the distinction on a short-term or a long-term vision. On the short-term: the production of steel and copper has a high CO_2 emission rate and therefore a big negative influence on the environment. On the long-term: the elements can easily be deconstructed from a building, do not need remanufacturing, and can therefore easily be reused. And they do have a high life expectancy, far longer than most building materials. The difference on the short-term and the long-term indicates the difficulty in which decision is better. The circular economy aims to lower the negative impact on the environment and create long-lasting buildings and building materials. A circular design needs therefore always consider what building materials it uses, what the impact is on the environment, and how the highest life expectancy can be achieved. And most importantly, that it generates no waste when the building is at the end of its life, by developing a smart design, which makes it possible to deconstruct the complete building. In the circular built environment, this is referred to as being a design for disassembly.

2.7 | Design for Disassembly

Design for Disassembly (DfD) is, clarified by McLellan (2019): "A design process that allows easier access to the materials, parts, and products of a building when it is renovated and/or disassembled. It provides flexibility whilst renovating, disassembling, or converting. It is intended to maximize value and minimize environmental impact through, reusing, recycling, repairing, and remanufacturing the whole or part of a building" (McLellan, 2019). In the current construction project designs, it is hard to replace or remove parts from the building. Therefore, demolition of a building always turns into waste since building materials get damaged in a way they cannot be reused anymore. In most cases, the various building materials can only be recovered as raw materials during demolition on material level. This leads to value destruction, the inflexibility of the system, and a bigger negative influence in the environment than is necessary. According to literature studies (Rios et al., 2015; Deller et al., 2019), ten principles can be distinguished which need to be considered, during the DfD process:

1. **Document materials and methods for deconstruction**. As-built drawings, labeling of connections and materials, and a ‘deconstruction plan’, all contribute to efficient disassembly and deconstruction.

2. **Select materials using the precautionary principle**. Materials are chosen with consideration for future impact, that have a high quality that will retain value and/or be more feasible for reuse, remanufacturing, and recycling.

3. **Design connections that are accessible**. Visually, physically, and ergonomically accessible connections will increase efficiency and avoid requirements for expensive equipment or extensive environmental health and safety protections for workers.

4. **Minimize or eliminate chemical connections**. Binders, sealers and glues on, or in materials, make them difficult to separate and recycle, and increase the potential for negative human and ecological health impacts from their use.

5. **Use bolted, screwed and nailed connections.** Using standard and limited palettes of connectors will decrease tool needs, and time and effort to switch between them.

6. **Separate mechanical, electrical, and plumbing (MEP) systems**. Disentangling MEP systems from the assemblies that host them makes it easier to separate components and materials for repair, replacement, reuse, and recycling.

7. **Design to the worker and labor of separation.** Human-scale components or conversely attuning to ease of removal by standard mechanical equipment will decrease labor intensity and increase the ability to incorporate a variety of skill levels.

8. **Simplicity of structure and form.** Simple open-span structural systems, simple forms, and standard dimensional grids will allow for ease of construction and de-construction in increments.

9. **Interchangeability.** Using materials and systems that exhibit principles of modularity, independence, and standardization will facilitate reuse.

10. **Safe deconstruction.** Allowing for movement and safety of workers, equipment, and site access and ease of materials flow will make renovation and disassembly more economical and reduce risk.

2.8 | Circular built environment Initiatives

There are multiple initiatives when it comes down to the elimination of waste in the transition towards a circular built environment. One of these initiatives is the Project BAMB (building as materials banks). This is an initiative with 15 partners from 7 European countries, that work together. Their vision is to shift the construction sector to a circular built environment. By creating dynamic and flexible designed buildings, building materials can contain their value, which will not result in generating waste after deconstruction, but create a stock of valuable building materials. This is in line with the previous sub-chapter about the DfD. The Project BAMB has named this as a reversible design, in which the buildings are completely deconstructable. This changes the core perspective on a building. Instead of being a building as a whole, it is now a material bank, rich in valuable building materials that are ready to be reused in other projects after the building is deconstructed (BAMB, 2020).

Another initiative that tries to reduce the waste production in the AECO industry is Madaster. The vision of Madaster is to eliminate waste by creating identifications for building materials. This concept is known as: material passports. With these materials passports, Madaster tries to give more insights in building materials that are used to construct the building, their quantities, the quality of the building materials, and financial and circular value. It is an independent platform that is creating a library in which all material passports will be saved to keep an overview of the existing stock of building materials. The materials passports try to give a circular insight in the buildings, based on the total number of products mass, virgin material usage, recycled reused and renewable materials, mass of waste, etc. Madaster is developing a platform to give insights in the buildings in the Netherlands based on the core materials. (Madaster, 2020). Madaster is one of the platforms that tries to assess buildings based on circularity-based indicators. But in the past years, other studies were conducted in how to assess buildings based on several circularity indicators as well. A tool that can be used as an assessment the circularity of existing buildings is described in the next sub-chapter.

2.9 | Building Circularity Indicators

The last step that is missing in this literature review, after many circular built environment principles, approaches, and methods are discussed, is how to assess existing buildings on their circularity opportunities. To assess the existing building, the assessment tool by M. van Vliet (2018) will be used. The assessment tool that he invented was developed during a graduation project for his master thesis at the Eindhoven University of Technology. The assessment tool uses Building Circularity Indicators (BCIs) to give scores to a building based on its level of circularity. The scores are based on twelve disassembly requirements. M. van Vliet used the Fuzzy Delphi Method, in which an expert panel defined and tested many disassembly requirements. The twelve requirements that will be described here are the twelve requirements that were concluded as being most important in the disassembly process of a building. The BCIs show similarities with the listed DfD factors, which is not unusual since both methods have the same approach in deconstructing a building. The DfD is implemented in the design phase of a construction project, in which the BCI assessment tool is implemented after the building is deconstructed. The twelve requirements are divided into three topics:

1. Technical requirements (technical factors)
2. Preconditions (process-based factors)
3. Drivers (financial based factors)

2.9.1 | Technical requirements

The technical requirements are the disassembly requirements that are based on building material level. These requirements have a practical approach in which the shape, connection, independence, etc. defines the scores of the circularity opportunities for the building. These opportunities are calculated for example: if the building materials can easily be disconnected, transported and if necessary, deconstructed in single components of the building material.

1. Independence

The independence of a building material is an important requirement. The disassembly potential becomes less when it is part of another construction element. If this is the case, the building material must be damaged during the deconstruction process, which results in loss of value, and that must be prevented. The independence of the building materials must be secured in the project and the incorporation with other construction elements must be avoided (Vliet, 2018).

2. Type of relational pattern

The type of relational pattern is the number of connections that the building materials have with other construction elements. The lower the amount of connections, the higher the disassembly potential is of the building material (Vliet, 2018).

3. Disassembly sequence

The disassembly sequence determines in which sequence the disassembly of the building materials of the project takes place. This sounds logical because the disassembly can only be executed in steps. It is wise to set up a disassembly plan to capture the value of the building materials. If materials must get damaged to get disassembled, the building materials with the highest value will be disassembled first. This will be the determinant factor in the disassembly process. When smaller products with less value can be disassembled without influencing other building materials, these of course will be disassembled first (Vliet, 2018).

4. Disassembly shape

The shape of the building materials is also an important factor in the disassembly process. The potential of disassembly depends on whether the building material can be easily deconstructed or not. The easier the shape, the higher the potential. The high potentials differ by having a smaller shape, less angles, can it be deconstructed manually, etc. (Vliet, 2018).

5. Method of fabrication

The method of fabrication distinguishes the building materials which are prefabricated or are composed on the construction site itself. Besides making the products more reusable, prefabrication leads to easier disassembly due to the standardization of connections, easier accessible connections, and the ability to disassemble complete components on-site and further separation of components off-site. The level of prefabrication determines the assessment of the disassembly process (Vliet, 2018).

6. Type of connection

The type of connection is not the number of connections with other building materials but is the type of connection between the building materials. The type of connection determines the accessibility of disassembly. If building materials have a chemical or glued connection with each other, it is difficult to separate and recycle them, and they will certainly get damaged. Therefore, the type of connection is also an important assessment criterion (Vliet, 2018).

7. Accessibility of connection

This requirement is focused on the accessibility of the type of connections from the previous criteria. The highest value of the building materials can be captured when, during the disassembly process, the connection is easily accessible without damaging the building materials. This influences the value of reusability of the building materials, but also makes the disassembly process easier and quicker (Vliet, 2018).

2.9.2 | Preconditions for disassembly

The requirements for the preconditions for disassembly are process-based factors. This means that these requirements are the preconditions in order of the disassembly process rather than the disassembly of single building materials. The preconditions are set to create a safe and overthought process while deconstructing the building. It can be a dangerous process if not planned out well in advance.

8. Deconstruction safety

The deconstruction safety affects different levels in the deconstruction process. The overall deconstruction process needs to be a process where safety comes first. Because the building completely gets deconstructed, the building gets more fragile and movable. The easiness in dismantling building materials, in terms of manual disassembling or the needs of a crane, makes a big difference in the safety assessment. Deconstruction safety is regarded as a precondition during the development of the project, which makes the safety in the disassembly process more viable (Vliet, 2018).

9. Disassembly instructions

Disassembly instructions can help in overcoming process-based challenges and communicate specific technical interventions to enable disassembly at the end of the lifecycle of a building. The core of the disassembly instructions is to provide a deconstruction plan in which the dismantling of the building materials is described with their belonging safety factors, type of connection, etc. (Vliet, 2018).

10. Disassembler expertise

There is a difference between demolishment activities and disassembly activities. Connections can be complex, and the experience of the construction worker may be insufficient, leading to demolishment instead. It is important to distinguish these two activities to get the highest value of the building materials during the disassembly process. Expertise is an important factor in this building circularity indicatior (Vliet, 2018).

11. Number of operations

This criterion is harder to assess on its own, but it is important to consider it. The high potential in the disassembly process should be a low amount of operations. This is because every building material that can potentially be disassembled, comes along with a variety of operations because required tools are needed, different types of connection, accessibility of the connection, etc. Reducing the number of operations can be achieved by carefully planning disassembly operations (Vliet, 2018).

2.9.3 | Drivers for disassembly

The drivers for disassembly are financial-based factors. The drive to deconstruct a building will always be involved with a revenue model. If the costs are higher than the profit when deconstructing a building, this process will probably not take place. Therefore, it is wise to draft a balance sheet to make sure the process is profitable.

12. Disassembly costs

In this criteria, all costs are involved that are necessary for the disassembly of the building. Besides, there is a common perception that the costs of a deconstruction process are higher than the costs of when the building was demolished. But the DfD recognizes that if provided services for dismantling are considered in the initial building design, these costs can abundantly be reduced. And next to that, new profits can be applied to the business model since the extracted building materials can be reused which results in not have to purchase new building materials, or the building materials are sold and this will evolve in more profits as well (Vliet, 2018).

2.10 | Summary

The first part of the literature review of this research gives a clear overview of how the transition from a linear towards a circular built environment takes place. Besides the fact that the linear economy still will be part of the economy in the common years, is the circular economy introduced on a broad scale already. With the sustainable approach, that has been set for many years, is the circular economy a good extension. This to overcome the negative environmental impact, in which the construction sector has a major part in. The implementation of the closing loops already changed the primary as well as the secondary production processes. But, stated is that the reuse loop is most attractive, since it is the easiest and most environmentally friendly loop of the four cycle loops, in the technical feedback loop side of the circular economy system diagram (Figure 2). The reuse loop has the preference since it is concentrated on product level, and therefore can immediately be reused. This is possible without remanufacturing of the components of the products is needed or to conduct a recycling process to the extract the core materials of the products, which both still includes many environmentally unfriendly aspects.

The literature study supports the aim of this research by reducing the waste production of the AECO industry by stimulating the reuse of building materials from deconstructed projects. The solvation of the waste production will immediately close the cycle loops in the construction sector. The end of the life cycle of the buildings becomes therefore more and more important. The buildings in a circular built environment are observed as a stock of building materials. The vision of the reuse potentials of the building materials changes the view on the building as a being a material bank. The only challenge is to extract the building materials from the project when it is at its end of life cycle.

The transition from a linear to a circular built environment is implemented in every phase of the construction project, from the design phase until the demolition phase (Figure 3). In every phase, environmentally friendly changes can be introduced based on the circularity approach. But what can be concluded from the literature study, is that in the past years, especially in the design phase has the highest potential of inventing circular concepts. These directly have therefore a positive circular effect on the other phases of the construction project. A discussion arises in the design phase which building materials should be used, based on its environmental impact in the short-term or the long-term. Biodegradable building materials have a positive influence on the environment and especially on not having to exhaust natural resources for the production process. But these buildings materials need to be replaced more often due to its shorter life expectancy, because of a lower strength of its core materials. Building materials that have a high life expectancy are the opposite, they can exist for the long-term without maintenance is necessary. Structural building materials as steel column can be used for many years without maintenance, but their core materials are extracted from natural resources. This debate will still be part of what is best for the construction sector as long as long lasting building materials, which are not extracted from natural resources can be invented.

The circular concepts and initiatives, which are mainly focussed on the design phase, will have a great contribution to the reuse potentials of the building materials when future projects get deconstructed at its end of life cycle. The value of the building materials is captured during the deconstruction process, when implementing the circular or reversible design, and the design for disassembly principles, as they are discussed in the literature review (2.7). The initiative by BAMB aims to implement these concepts in a wide scale already. This is a very useful circular approach for the circular built environment in the future, but this is not the aim of this research. The aim of this research to stimulate the reusable building materials from existing buildings, to contribute to a circular built environment immediately when existing buildings get deconstructed these days. The initiative of Madaster is the first step towards composing a database of the stock of existing buildings in the Netherlands, based on their circularity indication. The materials will all be provided with a materials passport. These passports give insights into the core materials and the virgin materials usage of the building. This, to understand what the environmental impact is of the building nowadays and when it was constructed. But this is still does not stimulate the reuse potentials of building materials from existing building, it rather just collects circular-based data on a platform.

To be able to assess an existing building based on its circularity indicators, the last sub-chapter (2.10) defines an assessment tool. This assessment tool scores the potential of how well the building can be deconstructed, considering the value capturing of the building materials. The building circularity indicators score how much value of the building materials can be captured after they get assembled from the building. This assessment tool can stimulate the reusable building materials from existing buildings and can therefore be a strength to elaborate on in a further stage of this research.

< Page intentionally left blank >

Section 3

Building Information Management

3 | Building Information Management

The second part of the literature review consists of the management of building data. To get an insight into the stated development (Section 1) in the AECO industry about data management and connectivity, will provide this section insights into how this is implemented and resolved in the construction sector. The Building Information Management (BIM) has a major part in combining the data management and connectivity in the AECO industry. The implementation of the BIM methods is described in the first sub-chapter (3.1). Different maturity levels can be distinguished in how the BIM process is used in a collaboration project. These maturity levels are defined in sub-chapter (3.2). To connect the participants in a BIM process, several standards are developed to create guidelines for a universal collaborative platform. The guidance of the BIM standards and the information sources within the BIM models are described in (3.3). After the standards and different levels of information input in the BIM models are discussed, the next sub-chapter describes the connection between participants that increases the collaboration potentials. This is possible using the openBIM workflow (3.4). An important aspect of openBIM is the information exchange standards and file formats, which provide the actual opportunity to exchange information within project teams. The different open standards that are invented to increase the collaborative aspect and to make the interoperability between software tools possible are defined in sub-chapter (3.5). A literature study is also conducted on how to make the collaboration and interoperability connection with cloud-based software (3.6). To manage the BIM models that are developed in the collaborative project teams, the quality can be controlled using BIM-based model checking software (3.7). To describe the bridge between the two developments in the construction sector (Section 1) and to realize the goal of this research, the connection between the BIM process and the circular built environment is defined (3.8) as well as possible standards that can contribute to this connection (3.9). The last sub-chapter (3.10) formulates the conclusion on this section.

3.1 | BIM & the AECO industry

The implementation of the BIM models improved the possibility to manage the building data in the AECO industry. The focus shifted from the graphical data in 3D-models, to the importance of the non-graphical information data in the BIM models. A BIM model is not just a visual example of the project that will be built but is a database of information about the building materials, their properties, characteristics, and relationships between the variety of objects (McArthur, 2015). The AECO industry was able to increase the quality of their projects, due to the introduction of BIM. The costs of the project can be estimated with BIM software, because quantities can be extracted from the BIM models. The planning of the companies can be connected to the construction sequence in the BIM model. This involves the opportunity to add delivery schedules and to coordinate the materials ordering process. The collaboration and interoperability potential between companies increased as well within the AECO industry. The BIM software made is possible to exchange information and to combine the BIM models in BIM software. Clash detections and conflicts in the design are controlled to avoid interference and exclude failures during the construction phase. The BIM process creates a more effective collaboration potential, more efficient projects, higher design quality, be able to manage the project in its full life cycle, and an easier way in communication between the stakeholders involved in the project teams. The implementation of BIM changed the traditional work processes in the AECO industry and provided the opportunity to collaborate in a digital environment (Azhar, et al., 2011; Bui, et al., 2016).

3.2 | BIM Maturity Level

To measure on which quality level companies use BIM, the BIM maturity levels are invented. These aim to define the construction supply chains' ability to operate and exchange information. These levels were invented by the UK BIM Taskforce (CAD & Company, 2019). In the UK it is mandatory to build with BIM methods since January 1st, 2016. Being the pioneers in providing BIM standards, the results from the taskforce were incorporated by the rest of Europe over the years. The BIM maturity model is depicted in Figure 4. As can be stated from the figure, the BIM maturity has four levels:

In BIM Level 0, which contains CAD (computer-aided design). In this level, no standards or processes are used, only 2D drawings and a brief exchange of electronic information. BIM level 1 also uses CAD (but it can also be done without), but in a managed way. BIM level 1 is still executed in a private environment where the exchange of models with other stakeholders in the project team is still not accessible. At this level, most of the time the design will be in 2D, but the 3D design is already implemented briefly. The models are, in contrast with BIM level 0, upgraded with 3D information and graphics. It still uses the exchange of document and the collaboration between companies is still low.

BIM level 2 is a managed environment, but this time in a fully 3D environment. This level is the first level in which we can actually speak of a BIM model. The BIM models contain graphical and non-graphical data, but they are still designed in a private environment, separately from other companies. This is the BIM level that is tried to be achieved in the AECO industry as we know it nowadays, but this is still poorly done due to a lack of collaboration and design quality. The different disciplines in the project team design their separate models, which are assembled in a later status of the project. The separate models together form the complete BIM model of the building. These files will be exchanged, and software programs are used to combine these single models. This will be elaborated in sub-chapter (3.5). The cost and planning aspect can be connected to these models as well (CAD & Company, 2019).

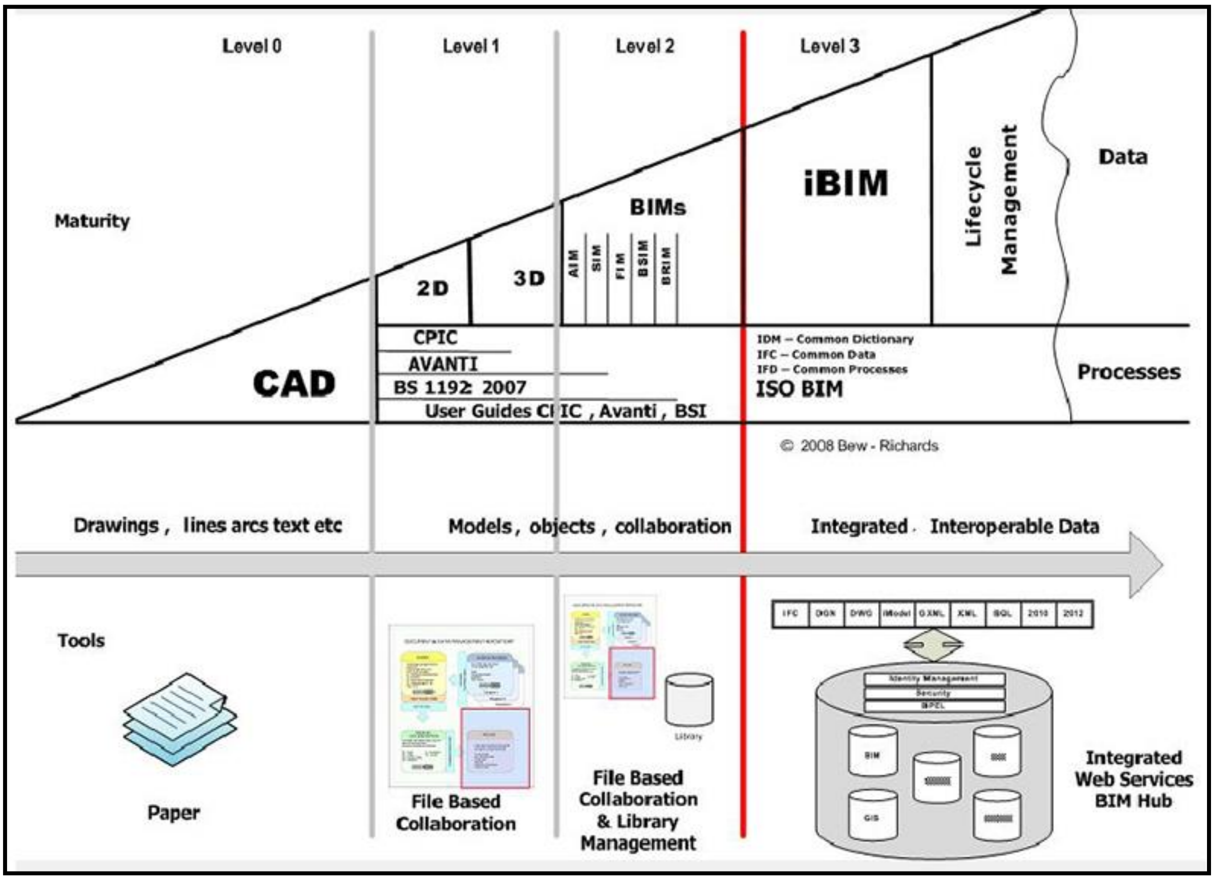


Figure 4: BIM Maturity Levels (Source: Bew & Richards, 2008)

The final level in the BIM maturity model is BIM level 3. The first steps towards this level are briefly implemented in the AECO industry. This integrated form of BIM, sometimes known as iBIM, contains one collaborative project model with construction sequencing (4D), cost (5D), and project lifecycle information (6D). It is also important that open standards as well as the ISO norms, concerning the BIM methods, will be implemented. With BIM Level 3, the highest value of the BIM methods is tried to be captured. This process is focused on the full life cycle process of the building, and it all will be captured in the design phase. Within the circular built environment, this is the BIM level which shall contribute to a major advancement reaching this circular status (Tekla, 2019).

3.3 | BIM Guidance

To accelerate the universal implementation of the BIM maturity levels, international standards are developed, to increase the interoperability in the AECO industry. In the past years, several documents were developed to create a more effective collaboration and management of the data throughout the different project phases in a BIM process. The international standard that is commonly used in the AECO industry is the ISO 19650 (British Standard Institution, 2019). This BIM standard replaces the PAS 1192, which was invented in the UK to reach the BIM Level 2 maturity, as it was compulsory by the government. The ISO 19650 is an extension of the old standard and affects the same principles and requirements. The difference between both documents is that the ISO is the international approved standard. The ISO (International Organization for Standardization) develops and publishes international non-governmental standards. It brings together experts to share knowledge and provide solutions to global challenges (ISO, 1946). Every country is involved with processes in the construction sector, with variations in laws and cultural approaches. With this worldwide collaboration of standardization, the processes on an international level will be easier and more profitable. Especially when seeking guidance in the BIM environment, international standardization is important. The ISO 19650 is an international standard developed to move on to the management of information during the operational phase of assets, and the adoption of a security-minded approach to the management of information relating to sensitive assets (National BIM Standard, 2020). Besides the standardization of BIM requirements, interoperability in collaboration is only possible if the exchange of the BIM models and its information is applicable by the stakeholders involved in the projects. Therefore, the BIM information exchange standards are invented to increase the collaboration potentials (sub-chapter 3.5). Another important international group that provides standards is the CEN/TC 442 (European Committee for Standardization). The CEN/TC 442 operates in close collaboration with for instance the ISO committee. They try to achieve the same: increasing the collaboration and interoperability potentials by developing international standards. The CEN/TC 442 has the objective to deliver a structured set of standards, specifications, and reports with a specified methodology to manage the BIM data. It advises EU Commissions on how to implement BIM. Their working groups have a variety of topics on which they are willing to develop standards for, such as: terminology, exchange information, information delivery, support data dictionaries, chairperson’s advisory group, infrastructure and horizontal role of BIM (CEN/TC 442, 2019). Together with the ISO 19650, it provides specific requirements on the delivery status of graphical (level of detail) and non-graphical (level of information) data. It is important to define these levels before starting a collaboration project.

3.3.1 | Level of Detail

With the distinction between the properties in a BIM model, based on the graphical and non-graphical information, the level of detail (LOD) is defined on the graphical part of the BIM model. The graphical information, which consists of the geometric data, is different than the level of development, which also is defined as LOD in the AECO industry. In this research, the LOD refers to the level of detail. The level of development is published by the American Institute of Architects (AIA) and differs from the level of detail in the requested output. If a certain level of development is agreed between parties, not only the graphical part but also the non-graphical part of the object in the BIM model should be designed with at least that level of development. The level of development defines a broader range of design quality (AIA, 2013). The level of detail is it is described in the PAS 1192 and the ISO 19650, only defines the level of graphical and geometric design quality in the BIM model. A BIM model with a high level of detail, does not immediately consist of a high level of development. On the other hand, if a BIM model has a high level of development, this immediately consists of a high level of detail. The level of development is the level of detail combined with the level of information. (Hijza, Omar, 2017).

The LOD can be divided over five levels. Between multiple actors, defining the LOD is used to coordinate the modelling efforts (Gigante-Barrera, et al., 2017). When an actor wants to give a certain quality of graphical information to an object in the model, he can give a LOD status to the attribute. If this status is attached to the object, a standardized checklist is attached to the object as well. In this manner, it is guaranteed that the object will be modelled based on this level of detail and contains the requirements from the checklist (Grytting, et al., 2017). The visual object would provide graphical content. The LOD is in other words the level of graphical content that will be developed in the model. The higher the level of detail, the more graphical content is available. An example is illustrated in Figure 5. A window is designed based on the difference between the five levels of detail. The five levels that are mentioned below are the average definition of the LODs but is still is not clearly defined which guidelines truly belong to which LOD (EC, 2018). To five levels give an indication on the average expectation when is requested for a certain LOD:

- LOD 100: Only the volume of the object is visual.
- LOD 200: Concept of the areas, volumes, orientation and cost
- LOD 300: Quantities, sizes, shapes, location, and orientation is defined.
- LOD 400: Coordinated objects, with an accurate cost estimation
- LOD 500: Suitable objects for the maintenance and operations phase

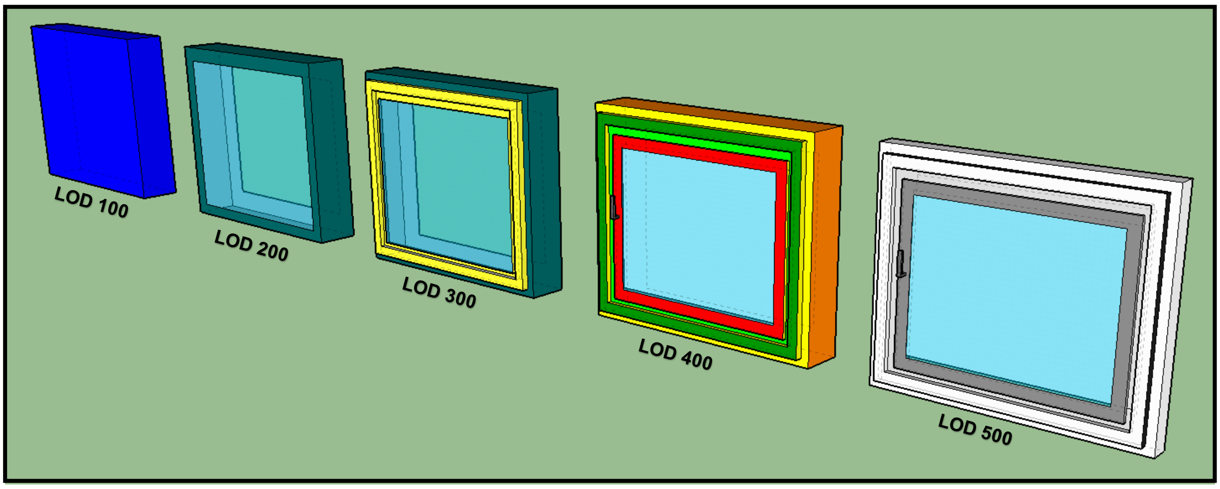


Figure 5: Example of a window with the stepwise change in its Level of Detail (LOD)

3.3.2 | Level of Information

In which the graphical information of a BIM model can be defined with the LOD, the non-graphical information can be defined as well. The level of information (LOI) is responsible for the level of non-graphical content that is attached to the objects. The non-graphical information is the information which not involves geometric data, but information about the properties and characteristics of the objects. E.g. for insulation this could be the thermal resistance and/or fire and acoustic rating. The LOD gives us information about the height and the thickness of the insulation, in which the LOI has a more textual-based content. This is the distinction between the BIM Maturity Level 1 and Level 2. In which the graphical content that was already briefly implemented in Level 1, is the non-graphical content is only involved in Level 2. This immediately increases the quality of the BIM model, which has the priority to fulfil the I (information) in the BIM model (NBS, 2015). The different steps in the LOI are also not clearly defined which information input belongs to which LOI. It therefore is wise to pre-define what is expected to exclude misinterpretations and lack of desired information.

3.3.3 | Documentation

A (sometimes) third source of information in a BIM process is the documentation. Besides the graphical and non-graphical information, the documentation is also a source that is typically presented in a Common Data Environment (CDE) Figure 6. The CDE is the virtual environment that collects every information source of a particular project. This environment manages the information sources, updates them, and can exchange the information among the parties involved (BIM & ICT, 2016). Not every BIM software tool includes the documentation source as well, but if it does, the documentation contains useful information. The documentation consisting of a BIM model is based on contact details of the participants and companies involved, documents related to the project involving contracts and agreements as well as documents about products and materials.

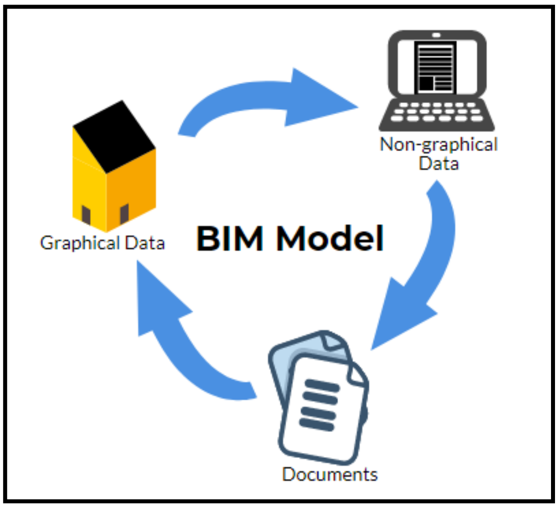


Figure 6: The three information resources in a CDE (Redrawn from: NBS, 2016)

3.3.4 | Level of Information Need

The ISO & CEN committee introduced a new convention, the level of information need (LOIN). With their aim to create a standardized AECO industry, the LOIN will replace the LOD and LOI, and therefore also the misinterpretations about the acronyms and definitions of both concepts. As was previously mentioned in sub-chapter (3.3.1), other definitions are used for the same acronyms. This creates misunderstanding and that is must be reduced and avoided by the committees. Therefore, the LOIN will define both the graphical and non-graphical information need. The LOIN will specify the maturity levels of the objects that are modelled, the levels are not yet defined on which scale this will be outlined. LOIN has not yet gained broad international support. But the ISO 19650 and the CEN/TC 442 will in the near future customize their guidelines and desired terminology to create a more standardized international AECO industry (Churcher, et al., 2019; Baldwin, 2019; Moses, 2019).

3.4 | OpenBIM

From previous literature review, we know that a BIM model is a digital 3D presentation, with graphical and non-graphical data which makes the objects more informationligent. As well as the BIM maturity levels and the invented standards that are used worldwide for implementing the BIM concept in the companies' workflow. The next step is to combine this information and to increase the collaboration potentials. Therefore, openBIM is invented and has the aim to tackle collaboration problems between the project teams in the AECO industry. By providing definitions, requirements, and tools the collaboration between stakeholders such as architects, engineers and contractors will be realized (CADalyst, 2012). OpenBIM is an initiative, by the buildingSMART organization and software suppliers. BuildingSMART defines OpenBIM as:

"*OpenBIM extends the benefits of BIM by improving the accessibility, usability, management, and sustainability of digital data in the built asset industry"* (buildingSMART International, 2020).

OpenBIM provides specific protocols and workflows to get a very strict control on the data exchange. It has the aim to connect the private designed models to create collaborative BIM models. The exchange of BIM models is possible using open file formats. This, to make them accessible for the members of the project team which use different BIM-bases software. The complexity of exchanging project information and sharing BIM models created with different software systems, is sometimes underestimated.

An open information exchange approach is necessary to create a transparent process to increase the interoperability and therefore the quality of the project. As the paper of Juan et al. (2014) clarifies: "The characteristics of the building information, such as huge quantity, complex types, wide resources and scattered storage, the efficiency and effectiveness of information communication and exchange between each sub-system and various project participants is critical to the successful implementation of the project" (Juan, et al., 2014).

3.5 | BIM Information Exchange Standards

OpenBIM provides the first steps towards information exchange possibilities due to the protocols for data exchange and the workflows for an easier collaboration process, but open and neutral standards must be implemented as well to achieve the desired interoperability. As was briefly mentioned in the previous sub-chapter, the exchange of information in a BIM environment, is only possible with an open file format. In the AECO industry, the IFC (Industry Foundation Classes) is the open file format that is most commonly used for exchanging BIM models. This format is recognized as the BIM neutral data exchange format for interoperability in the AECOO industry (Juan, et al., 2014). A strength of IFC as an open standard, is that it is independent from software systems and that it can easily be exchanged between different applications. It can therefore be adapted by different software systems. This is also a weakness from the IFC open file format. It only is an exchange format, that only exchanges data between other software tools. If an IFC file is opened in a software tools, it will be converted to the format of that software tool. But, with the development of the open file format, the participants in the project team can easily exchange their BIM model, characteristics, properties, geometry, and other related construction management data.

In an IFC file, the graphical and non-graphical data are included as well as the objects and their connections, including all data that is structured over the different stages of the building (NBS, 2015). But, IFC does not the collaboration problem. To design on a certain LOD and/or LOI, requirements are defined to achieve this status of detail and information. These requirements are not included in an IFC file. To overcome this problem and exclude misinterpretations or designing on different scales, a manual was invented where these requirements are listed: The Information Delivery Manual (IDM). The IDM is inextricably linked to the IFC standard. With the IDM, the collaboration process can be controlled by specifying the information exchange requirements. A more detailed user-defined specification delivery can be created with the IDM (Zhang, et al., 2012). To be able to get a really specific request for particular data in a BIM model, another standard is developed. The IFC and IDM standards together aim for the total project requirements, the Model View Definition (MVD) predefines what information needs to be transferred by who and at what moment during the project. It demands for specific packages of information. It is a powerful concept with a clear and structured way of working. To avoid big data exchange for specific requests in a later stage of the project, the pre-defined packages can be used as an IFC file export, which only includes the information that is requested within that package (buildingSMART International, 2019).

With these standards, the exchange of information, collaboration, and interoperability potential increased enormously. But still development takes place to increase these processes. The CEN/TC 442 started different committees which all have different scopes in increasing the interoperability and collaboration processes in the AECO industry. The CEN/TC 442/WG2 committee is concentrated on the exchange of information. The CEN/TC 442/WG3 committee is focussed on the Information Delivery Specification (IDS). The IDS is based on the process, the IDM, and the MVD (Clemen, 2017). In Figure 7 is illustrated how these standards are located in the BIM collaboration process. The collaboration layer is defined by the IDM. The participants exchange their information based on the exchange requirements. The functional part and the MVD support the software development of the exchange requirement model. The

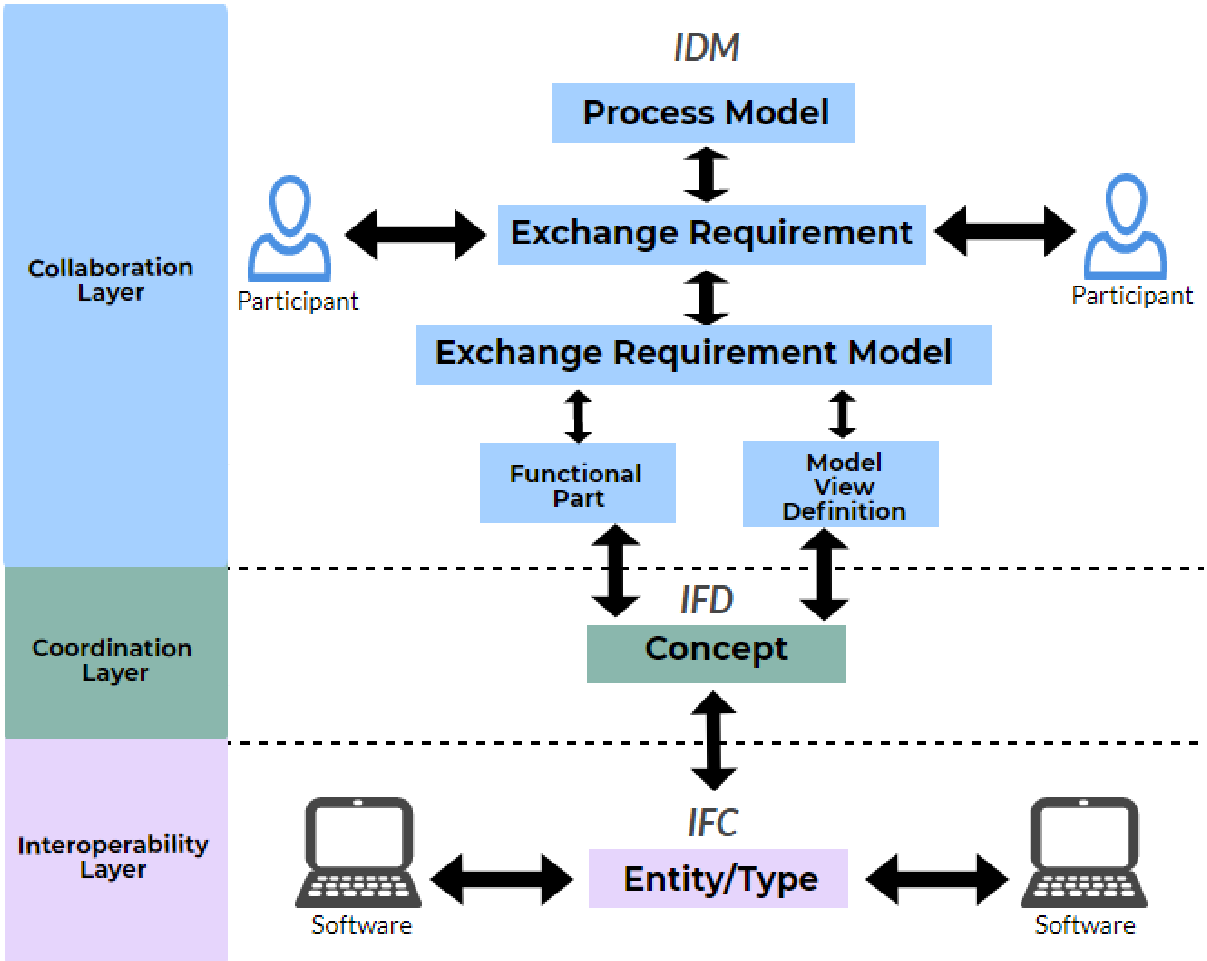


Figure 7: BIM information exchange standards (Redrawn from: Zhang, et al., 2012)

interoperability layer is defined by the IFC, which makes it able to create this connection between different software tools. The connection between the collaboration layer and the interoperability layer is specified by the International Framework of Dictionaries (IFD). These two layers can be coordinated by the IFD. This defines the use of particular names, properties, types, and entities as a standard is especially used as a coordination tool within the project (Zhang, et al., 2012).

Now the exchange standards are discussed, the next step is to widen the scope and try to find the connection outside the AECO industry, with the cloud-based information exchange standards.

3.6 | Cloud-based Information Exchange

OpenBIM increased the interoperability for the AECO industry. But, the connection with cloud-based formats was still not provided and therefore new standards were developed. This to increase the interoperability with cloud-based formats. As is depicted in Figure 8, a scale is developed in which the standards are categorized based on (left) high threshold or (right) higher adoption. BuildingSMART developed this roadmap, to widen the scope in terms of exchanging information not only within, but also outside the AECO industry. With new technologies like Artificial Informationligence (AI), blockchain, APIs (Application Programming Interface), difficulties arise when trying to develop connections for the exchange of information. As is stated for the scope of this research, connectivity is an important topic. The connection between software tools, programs, systems, and cloud-based processes. The problem that was tackled within the AECO industry with the implementation of openBIM can now be used to solve problem with the connection to cloud-based software. Neutral open file formats and open standards that are able to connect the software tools within the AECO industry with cloud-based software. To lower the threshold, Figure 8 highlights that the former standards like IDM, IFC, and other used open standards try to find their way to the right side of the scale, which in their case become more generic, scalable and easier adoptable by companies (buildingSMART International, 2020).

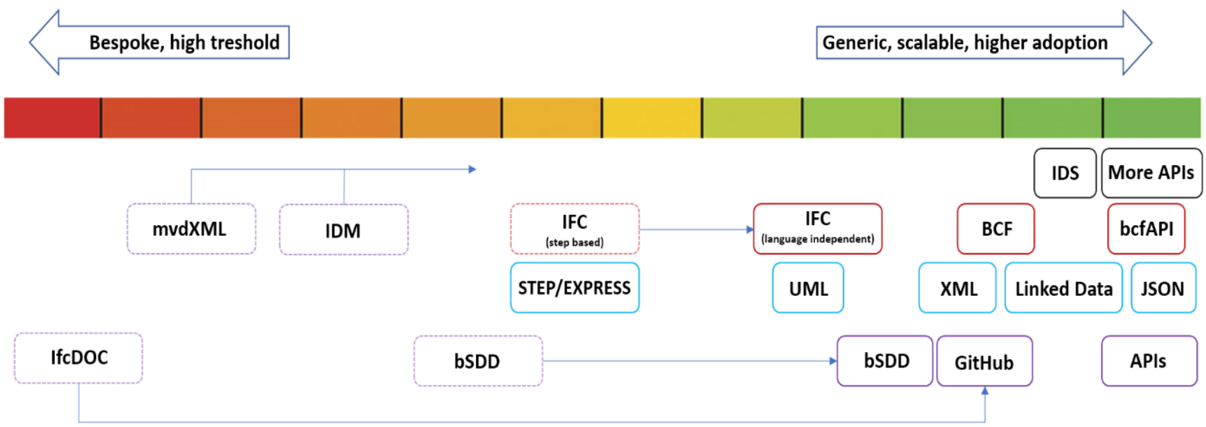


Figure 8: Technical Roadmap buildingSMART (Source: buildingSMART International, 2020)

It is necessary that buildingSMART developed this roadmap, because the open standards and formats need to stay up to date. With new open standards and file formats like XML (Extensible Markup Language), JSON (JavaScript Object Notation), and RDF (Resource Description Framework) the interoperability connection between (cloud-based) software programs can be realized. With these structured file formats, the data can be connected (linked data). This lowers the threshold in exchanging data and create a higher adaption potential. The AECO industry is moving to this cloud-based environment, and therefore needs to move to the right side of the roadmap.

The step for the AECO industry towards a cloud-based construction sector, demands for new requirements for their used standards. The IFC data-exchange standards need to be improved to build this connection towards the new cloud-based file formats. The way an IFC file is structured is a challenge when providing this connection. An important software solution for this problem could be the API. If an IFC file is used with an API, the IFC file can be accessed, maintained, and exchanged (buildingSMART International, 2020).

With the transition towards cloud-based software, the data serialization formats XML and JSON are widely adopted in the data transmissions towards web applications. As is depicted in Figure 8, both data exchange formats are generic, scalable and easy adoptable, which makes them strong formats to work with. Both are platform-independent languages which are used for the development of cloud-based applications. Between the two data exchange formats, JSON gained the highest preference to be used in web development. According to the paper of Afsari et al., the state that: “JSON is a key-value style lightweight data exchange format that has higher parsing efficiency than XML and due to the inadequacies of XML, JSON has been widely used in web applications, specifically in Javascript web services” (Afsari, et al., 2017).

3.7 | BIM-based Model Checking Software

The levels of detail and information together with the BIM maturity levels are defined, and the information exchange format and standards are described, all to increase the collaborative and collaboration potentials in project teams. But, the quality of the BIM models must be controlled and managed as well, to achieve the desired quality. The quality check of BIM models is executed by BIM-Based Model Checking (BMC) software. Each construction project starts with a list of requirements for designing the building. If this list of requirements is added as a ruleset to the BMC software, it is possible to control and ensure that the requirements are met in the BIM model. BMC software tools make it able to define the quality of the BIM models, to detect design flaws and finally to avoid failures in the construction phase of the project (Hjelseth, 2015a). The quality of the graphical as well as the non-graphical information in the BIM models can be assessed using BMC software.

The BMC software provides a cross-disciplinary collaboration platform, in which the interoperability possibility with the IFC files (if their quality is ensured as well) is the most important aspect. The privately designed models can be exchanged as IFC file formats and uploaded into one BMC software tool. Once they are connected, quality checking, calculations of quantities, simulations of time and planning, and other actions are possible to conduct on the entire project (Dimyadi, et al., 2016). The pre-defined requirements can be implemented in the BMC software using rule sets that provide clash detections based on the rules (the requirements) that are attached to the software. The rulesets are also augmented with the overall requirements that try to avoid design flaws if objects clash with each other based on mistakes in the geometry. Other requirements for the rule sets can be adjusted as well, depending on the project (Nawari, 2012; Hjelseth, 2015).

Many BMC software tools are developed for the AECO industry. Examples of commonly used BMC software tools are: Navisworks (Autodesk), Solibri Model Checker (Solibri), and BIMserver. The distinction between the BMC software tools is based on its variety of possibilities and extra added features. Some tools are more simplified than others, which are just developed to execute small quality checks. These tools are less complex and can therefore have the preference if a faster quality check is required. The difference between the Solibri Model Checker (SMC) and Navisworks is that Navisworks supports the input of many file-formats, which makes the interoperability potential more attractive. The SMC is easy in use and still contains a broad scale of specifications and features added to the software. It is a widely adopted tool in BIM collaboration projects to check the quality of the BIM model and it is possible to check if the non-graphical information, attached to the BIM model, is correct as well. Which BMC software tool has the preference in a project team depends on their requirements and objectives (Hjelseth, 2015a; Dimyadi, et al., 2016).

The BIM maturity levels did increase in the AECO industry with the implementation of the BMC software, since collaboration and interoperability between companies in a digital 3D environment became possible. Especially the clash detections and the immediate possibility to communicate these clashes with the other participants, increased the quality of the BIM models, and therefore, the complete BIM collaboration process. The BMC software tools do not only increase the quality of the graphical information, but also make it possible to check the quality of the non-graphical information. The BMC software tools also provide insights into the classifications of the non-graphical information in the BIM model. Due to this possibility, a higher quality on the LOI can be requested in a BIM model. Especially with the focus on the circular built environment, the non-graphical information becomes more important. The BMC software tools play an important role to get more insight into the operations and maintenance phase as well as the end of life cycle expectation of a building. A higher value of the non-graphical information in a BIM model plays an important role with the transition towards a circular built environment (Nawari, 2012; Hjelseth, 2015b; Dimyadi, et al., 2015).

3.8 | BIM & Circularity

The BIM process, as the name implies, increased the management of the building information tremendously. With the change from a digitalized 3D environment, to a digitalized BIM environment, the opportunities for the operations and maintenance increased as well. The information in the BIM model is not only important to construct the building, but also for the operations and maintenance phase. The non-graphical information availability in the BIM model makes it possible to conduct a Life Cycle Assessment (LCA) (Gao, et al., 2019). The LCA is a method that can calculate the impact that the building has on the environment. Based on the information that is adjusted to the BIM model, the energy consumption, CO2 emission, water usage, natural resource extraction, and other environmentally-based factors can be calculated. The BMC software tools provide the possibility for the AECO industry to get a broader insight in their BIM models. It can now even consider the impact on the operations and maintenance phase as well as the impact on the environment in the complete life cycle of the building (Soust, et al., 2017).

With the transition towards a circular built environment, the AECO industry is continuously investigating the opportunities to take the next step in the BIM process. The circular and reversible design, as well as the usage of biodegradable materials (Section 2), are concepts which concentrate on the complete life cycle of the building. They aim to create a new purpose for the building materials after the building is deconstructed. The possibilities are endless, but this demands for a high quality of information input in the BIM model. The ‘I’ in BIM already captured non-graphical information value in the AECO industry, but the transition towards a circular built environment requests for a higher level of information input. The AECO industry must implement a higher LOI, in which the data of project life cycle information is added to the BIM model as well (sub-chapter 3.2).

Therefore, the non-graphical information input of the BIM model process should be increased with specific requested information. If this specific requested information is applied, it can contribute in developing the LCA and can create circular loops when circular-based information about the building materials is added to the BIM model. This process is a concept that defines the complete state of the building over its full life cycle. Most of the benefits for the information about the life cycle of the building can be implemented by the stakeholders involved, during the design phase (Bicarri, et al., 2019). What can help by exchanging data based on a universal information structure, are the guidelines of the basis-ILS (Dutch: Informatie Leverings Specificatie). It is developed in a collaboration between multiple contractors and stakeholders in the AECO industry, in which there was a growing need for certain universal guidelines (BIM Loket, 2017). The basis-ILS is a two paged document (Appendix I), which aims to create an efficient and effective information structure in the BIM model, which has pre-defined agreements that must be met by all stakeholders involved in the project.

3.9 | BIM Model Information Delivery Guidelines

The basis-ILS guidelines have three main principles: collaboration, exchanging data, and increasing the quality of BIM models. The guidelines provide a structured workflow for project teams, in which the information input will be customized by certain requirements. The guidelines aim to deliver the information in a clear and structured way, in which every stakeholder can retrieve and store data, as long as the requirements are met. The exchange of information will be done, using the IFC file format, which contains the graphical and non-graphical information. The guidelines are intended to make universal agreements for defining the description of the classifications, entities, names of the objects, etc. In the exchange requirements, you can define the object information that is required e.g.: the load-bearing, internal or external usage, fire rating, etc. The object information part of the basis-ILS is focused on the LOI of the BIM model. This part is concentrated on the LOI of a single object and can therefore contain specific requirements (BIM Loket, 2017). Following the guidelines of the basis-ILS can create a transparent workflow for companies which aim to have a structured and uniform data output. The value of the LOD and the LOI is widely adopted and the need for a structured data input and universal way of exchanging data is the next step in the collaborative project teams.

The basis-ILS guidelines can always be customized by the needs of the project team. The agreements and requirements of the basis-ILS will always be the fundament of the BIM collaboration, but the customization of the guidelines can contain specific agreements and requirements, which also have the be met in that particular project. This can be provided as a customized basis-ILS with the needs of the goals of that particular project, but still using the same guidelines for structuring the information input (Duurzaam Gebouwd, 2018).

3.10 | Summary

BIM changed the entire process in the AECO industry. The implementation of the BIM methods provided many possibilities for collaboration and interoperability potential within project teams. BIM makes it possible to manage the building data and to control the quality of the design of the building. The BIM maturity levels give a clear indication on which level the BIM process is executed in a project and what needs to be improved when higher levels want to be achieved. The BIM process did not only increase the quality of the graphical part of the 3D model but especially applied the non-graphical information to the 3D model. The non-graphical information is the part of the BIM model which is most valuable for this research. The research must concentrate to retrieve circular-based information from the BIM models. To be able to stimulate the reuse possibilities of the building materials, the characteristics and properties are more valuable rather than the graphical data that is attached to the objects.

What can be concluded with the implementation of the BIM methods is that the AECO industry is continuously changing their workflows into more structured ways to upload the data and develops more universal agreements to lower the threshold in the collaboration projects. With the open standards provided by the ISO and the CEN/TC, the AECO industry received guidelines to increase the quality of the BIM models and the workflow for the collaboration process. The LOD and LOI are important concepts for defining the level of quality for the BIM model. It, therefore, is important to pre-define the requirements in front of the project, to ensure the right level of detail and information is included in the BIM model.

The openBIM workflow took the AECO industry to higher levels based on the collaborative approach, but the IFC open file format was needed to get the interoperability to a higher level as well. The IDM and the MVD requires from the AECO industry to concentrate more on the process before the design phase starts, rather than set the requirements during the design phase. This is in line with the BIM approach and the standards that are developed by the different committees and organizations. The focus is to concentrate upfront on the desired output of the project, and not only until it is constructed but on its full life cycle. If the desired output is defined and especially it is ensured to capture the value of the BIM model based on the information needed to create a circular built environment, the right agreements and requirements can be set. Especially the guidelines of the basis-ILS can be important tool to achieve the goals of this research. Pre-defining which information must be included in the BIM models to retrieve this data when the building is at its end of life cycle, is an important factor in the reuse possibilities for building materials. Knowing which data is needed when the building gets deconstructed requires a different approach in front. To retrieve a high level of customized circular-based non-graphical information from the BIM model, the development of circular-based guidelines for this information output is required to be able to achieve this goal.

The roadmap of buildingSMART International (Figure 8) is an important figure for the connection that will be realized in this research. It shows to which cloud-based software the IFC file formats must be connected to be able to shift the AECO industry to this current development. Insights are required in the way the data in structured in an IFC file and cloud-based software file formats in order to create the connection.

< Page intentionally left blank >

Section 4

Methodology

4 | Methodology

In the last section of the theoretical part, the research will be framed towards the practical part of the research. What can be concluded from the previous literature reviews as well as the problem that is defined will be discussed on the first sub-chapter (4.1). The findings from the literature will be defined and in how they will contribute to achieve the research goals. After this is determined, a system is proposed (4.2), which could be an answer on how to realize the connection between non-graphical BIM data and cloud-based software. The method that will be used to develop the proposed system is described in sub-chapter (4.3). The architecture of the system that will be developed is divided over different concepts and frameworks. These are explained in sub-chapter (4.4). The last sub-chapter (4.5) summarizes the conclusions that are determined in this section. It frames the research by providing a conceptual model, in which the approach to execute the practical part of this research is visualized.

4.1 | Unique Selling Points & Opportunities

What can be concluded from previous literature is the that waste production around the world is a serious problem, especially since a major part is caused by the construction sector. But the sustainable approach already reduced the waste that is generated over the years. With the usage of recycled materials and recycling the waste output, the first steps towards a circular built environment were taken. Intending to close the cycle loops, the construction sector tries to become circular. But what can be stated from the literature is that the recycling processes are superseded. The most attractive cycle loop, which has the most positive impact on the environment in its feedback loop, is the reuse of building materials. This is in line with the vision of the Dutch government, to have a fully circular economy in the Netherlands by 2050, that only includes reusable resources in their process.

The first part of the literature review shows that the implementation of the reuse loops obtained a lot of support and this provided a new mindset in the construction sector with their perspective on buildings. The buildings can be seen as materials banks, in which they are a wealth of reusable building materials, when the building gets deconstructed at its end of life cycle. To be able to capture the highest value of these reusable building materials, new circular-based concepts were developed. Especially for the design phase, in which the greatest benefits for a circular process can be extracted from when implementing new concepts. The circular design, reversible design, and design for disassembly are the three most known new concepts for the circular built environment. Especially the BAMB organization is supporting this new approach of designing a building, that can completely be deconstructed at the end of its life cycle, in which the highest value of the building materials can be captured. When these building materials can be extracted and they do not need maintenance or remanufacturing, they can immediately be reused. To collect the circular building data, the Madaster platform developed material passports, which indicate the circularity of the materials and the building as a whole, to get insights on the impact the building has on the environment. Both initiatives, as well as the new design concepts, are supporting the reusable building materials. But this is done for future projects. The unique selling points (USPs) that can be formulated taking a critical look at these initiatives, will be to set the focus on the extraction of building materials of buildings that will be deconstructed over the next years. The stock of existing buildings is enormously, and those buildings will reach their end of life cycle in the next years / decades. The waste reduction must be tackled right away, and not in the future. Second USP is that the Madaster platform collects the circular-based data of the stock of buildings in the Netherlands. This will increase the knowledge of how circular the existing buildings are but will not stimulate the reuse of building materials. What is missing is the possibility of an acquisition option to purchase reusable building materials. A unique selling point would be to not only collect the circular building data, but also be able to buy and sell reusable building materials on that same platform. This is a very important part to not only become circular, but also fulfil the economic part of the circular economy. By implementing closed cycle loops and to turn them into profitable revenue streams at the same time, will make it economically interesting for the construction sector and contribute to a circular built environment as well.

The second part of the literature review on the building information management defined the enormous steps the AECO industry took over the years with the implementation of the BIM methods, based on connectivity and data management. The most important aspects of this research that can be concluded here, is that the BIM concept developed a digital platform where companies can collaborate on and different software tools could be connected due to the interoperability possibilities. The 'I' in BIM increased the quality of the formerly 3D models, to not only have graphical data but also non-graphical information available in the BIM models. The literature review reveals, that especially with the non-graphical information possibilities arise due to be able to conduct an LCA and to retrieve project life cycle information. To be able to retrieve this data, the open standards as the IDM, MVD and guidelines of the basis-ILS, obligate companies to think about the desired output of the project before setting the requirements upfront. The desired output, especially on the project life cycle information, can only be extracted if this data is included in the BIM model. It therefore is important to clarify for the construction sector which information is required and desired for the project life cycle information. This, to be able to increase the opportunities of the BIM models for especially in the operations & maintenance phase as well as the opportunities of the building materials when the project is at its end of life.

Based on the conclusions from both parts of the literature review, multiple opportunities arise that can contribute in solving the problem that is defined for this research. The USPs point out that a platform is desired on which circular-based information of building materials of existing buildings can be collected and the materials can be sold on that same platform as well. This to accomplish the circular as well as the economic part of the circular economy. With the concepts of buildings as materials banks, the platform can become a library in which existing buildings and their building materials can be presented as material banks. By presenting the circular-based information of these building materials, their reuse potentials will increase, and therefore their reuse will be stimulated. But the second part of the literature review concludes that this circular-based information might not be implemented in the BIM models of existing buildings. It, therefore, is advised to analyze what circular-based information is desired to extract from BIM models if a project is at its end of life cycle that is needed to be able to stimulate the reuse of its building materials.

A final step that can be added to the defined possible solution is to get the platform to a higher level due to the stated shifting, in the literature, towards the usage of cloud-based software in the construction sector. This, to make the exchange of data more generic, scalable and higher adoptable (Figure 8). By using cloud-based software technologies, the platform can become a digital platform in the cloud on which the previously mentioned opportunities can be collected. A possible digital marketplace in the cloud (cloud-based software), with the supply and demand of reusable building materials (circular economy), driven by circular-based non-graphical information (BIM models), can be the answer on the main question that is formulated for this research.

4.2 | System Proposal

The proposed system for this research will be a cloud-based digital marketplace with the input of the circular-based non-graphical information from the BIM models about the potentially reusable building materials and the output of selling these materials to future buyers. Therefore, three components can be distinguished that together form the basis for the system: the platform, the input, and the output.

The cloud-based platform is the most important component of the system. This platform can solve the problem that is defined in how to realize the connection between the non-graphical BIM data and cloud-based software. By using BIM data in the cloud, the project information is always accessible. The data is applicable at any time by anyone. This accessibility, together with easy storage and data exchange possibilities, substantiates that a cloud-based platform can be the right system for the digital marketplace with the desired input and output. The cloud-based software that can be used is presented by buildingSMART (Figure 8). The most generic, scalable, and highly adoptable cloud-based data exchange formats are XML and JSON. But the literature review did conclude that JSON has the highest preference due to its lightweight data exchange that has a higher parsing efficiency (Afsari, et al., 2017). It is a widely adopted format, specifically in combination with Javascript, for web application development. Therefore, both will be used in the development of the cloud-based system, which will be an online application.

The online application will be a digital marketplace where supply and demand meet for the reusable building materials. The key group of the supply side (input) for the online applications will be project owners that have the availability of a project that will be deconstructed soon. The size of the project is broad, this can be from an owner of a private house to an investor with apartment complexes or offices. The end-users of the supply side for the online application are contractors, investors, real estate owners, clients, housing corporations, freelancers, and private house owners. The most important aspect of the project is that the building gets deconstructed soon and therefore its materials become available for reuse potentials. The size is not important, this can be a large amount of just a few building materials. Every key user for must be able to use the online application. But the main focus is for project owners which have a BIM model available of the project that will be deconstructed soon. This to achieve the research objective to realize the connection between non-graphical BIM information and cloud-based software. But to be able to export the BIM models into single building materials presenting as products on the online application, the right non-graphical information must be retrieved from these models. Only the characteristics and properties of the building materials, which are relevant for this research, should be retrieved. Therefore, a circularity-based guide must be developed which will give insights which desired circular-based data should be retrieved from the BIM models. Once this circularity-based guide is defined, it can be analysed if this data can be retrieved from the BIM models. This aims to ensure the value capturing of the desired non-graphical information in the BIM models. To be able to compare this data, an analysis will be conducted (Section 5) on BIM models which were designed over the past years. Analysed will be, which non-graphical information is already available in the BIM models and which of the desired circular-based information is not available.

The third component of the system is the demand side (output). The key group of the demand side (output) for the online applications will be the potential buyers of the reusable building materials. The key users of this group can be a variety of people, from people who are building a private home to clients that will build a large project. The end-users of the online application that want to buy materials are the webshop customers. The customers have the similarity to build a project with the input of reused building materials. Their project will be adjusted based on the building materials they can purchase. For bigger projects, architects can design a future project based on the materials that will become available in existing buildings nearby that project. The key users can be the same group as the input side, only this time they have demand for potentially reusable building materials instead of supplying them. This will close the reuse cycle loop. When the potential buyer can enter the application in the cloud, this will increase the usability and accessibility of the application. The input of the non-graphical information of the reusable building materials must be transformed in a way that they can be presented as products in a user-friendly webshop, to substantiate the idea of a digital marketplace.

To create a clear distinction between the three components for the system, the components will be defined in the next part of this research as follows:

The cloud-based platform, which is a digital marketplace, will be defined as 'the online application'. The supply side of the online application, with the clients who own a BIM model of a project, will be defined as 'the project owners'. The demand side of the online application, with the potential buyers of the reusable building materials in webshop, will be defined as 'the webshop customers'.

4.3 | Rapid Application Development

The development of the online application will be executed based upon the Rapid Application Development (RAD) method (James Martin, 1992). The methodology involves an incremental application design and the development of a prototype. The method aims at a fast application development process. This method is a good method to apply when developing a framework for web applications (RAD method, 2017). The method concentrates on the functionality and performance to enable faster outcomes and a proof of concept of the objectives of the application. By focusing less on the peripheral issues of the application, more time can be spent on achieving the aim of the research.

This method has four phases that can be distinguished: requirement phase (1), design phase (2), develop phase (3), and test phase (4) (Figure 9).

The first phase is to set the requirements for the application. The goals that are desired to be achieved by the application will be determined. This will evolve into a list of requirements that will form the basis for the design and development phase. These requirements have to be met during the other phases. In the design phase, the system will be visualized by drafted scenarios. The system design is the process that is necessary to develop the application. Visualizing the design makes it possible to understand, modify, and eventually approve the system that meets the requirements. When the systems' design is finished, the development phase starts. The scope of this phase is to program the script of the application, integrate codes and functions that help to meet the requirements, program the back-end and front-end, create user-friendly interface, and meanwhile test the system. It will be a continuous feedback loop, in which small tests evolve in new solutions to be able to solve the problems. Once the system does not fail or contain errors and the requirements are met, the proof of concept is achieved and it will be ready for the last phase, the test phase. In this phase, the application will be evaluated, validated, and if necessary, optimizations will be applied to the system. The value of the prototype will therefore increase. The last step will be to determine if the application is a success, might be close in being a success, or a complete failure. Depending on which result is achieved can be determined if the application is ready to be launched on the market or not.

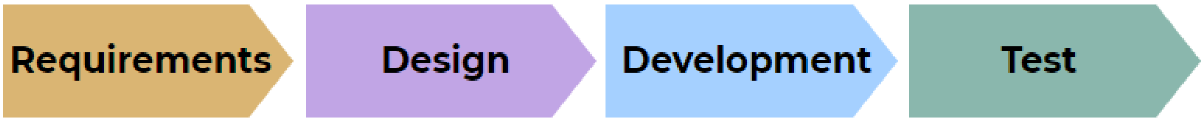


Figure 9: Four phases of rapid application development method (RAD)

4.4 | Application Architecture

For the architecture of the online application, a variety of software tools, methods and approaches will be used to create the final system design. The system design will be elaborated on in Section 6, but first, the decisions for the application architecture will be defined in this section.

4.4.1 | Event-Driven Architecture

The application architecture is the design of the pattern in how the application interacts in the system. The interaction between middleware, databases, and other applications is defined in the application architecture. Based on the requirements and desired output of the application, certain architecture patterns will fit or not. The most applicable architecture for an online application is event-driven architecture (EDA). An EDA distinguishes itself from other application architectures, due to its need of user input. The application is driven based upon the events that the users of the application need or request for. The EDA can be defined as an application that is designed to react to the actions of the users (Theorin, et al., 2015). An EDA is developed to serve the users with quick and accurate responses to create a better user experience. It, therefore, is a complex system, because it must facilitate great responsiveness. The complexity of this kind of system is a great benefit, because it makes it very intuitive, flexible, and can control internal as well as external events (Helendi, 2018). The pitfall of this architecture pattern is that you can get lost in the complexity of the application. Therefore, at all times, the requirements of the end-users are the scope of the development process, you need to fulfil their needs. Also, a clear system architecture diagram is needed to be able to keep track of components and overall structure during the development process. A disadvantage is that the privacy of the data is hard to assure. The EDA has a non-linear design character, which makes the security level of the application more complex. The privacy assurance of the data is for this research not involved. But if the application will be launched in the end, it is advised to cooperate with a security company to secure the data storage safety and avoid therefore avoid hacking potentials.

The event-driven architecture consists of three domains: event creator, event manager, and the event consumer. These three domains can perfectly be applied to the specific requirement need of the online application. This makes the EDA a good solution for the system design. The source of the event is applied by the event creator. The event creator develops an impulse and broadcasts a signal to indicate that an event is created. The event manager receives this impulse and functions immediately as the controller of the events. The event manager is the intermediator between the event creator and the event customer. After an event is detected by the event manager and it is controlled, it triggers an action that is related to the event consumer. The event manager can apply some rules to the process of the event. After an event proceeds to the event consumer, a single event can evolve in countless activities for the event consumer. But an event consumer needs to be triggered by an event from the event manager before it is able to use it (Ayanoglu, Aytas, 2016; System Innovation, 2019).

4.4.2 | Scripting Framework Architecture

Before a scripting framework architecture can be implemented, it is wise to summarize the first requirements that are already stated in the previous sub-chapters of Section 4. After the requirements are listed, a better decision can be made of which scripting framework is best applicable for this research.

(1) The online application will be cloud-based, which means a connection must be made with a web server to be able to run in a web browser.
(2) The project owners need to be able to upload their project data and afterward be able to add the reusable building materials as products.
(3) The non-graphical information analyses (Section 5) will evolve in a certain data output, which must be implemented in the system design to retrieve the desired data from the project that will be uploaded.
(4) The webshop customers need to be able to view the products in a webshop and to be able to purchase them, the contact information of the project and its owner need to be applied to the products as well.

What can be determined discussing these requirements is that a connection must be developed to connects the project owner with the webshop customer. They must be able to store, upload and download data. A dynamic database must be added as well. Therefore, the following scripting framework will be implemented in the system design. This, to provide the necessary connections to meet the requirements. This research uses the Model-View-Controller (MVC) framework (Reenskaug, 1970), which is one of the most used frameworks in the development of online applications or web development. The MVC framework (Figure 10), as its name implies, consists out of three components: the *model*, the *view* and the *controller*. This framework makes the development of an online application much easier by creating the right connection in the code which forms the basis of the script. This is done by splitting the code into the three components, which all have their purpose. In the next sub-chapter (4.3), a more in-depth insight is given per component, but first, the overall description of the MVC framework is defined.

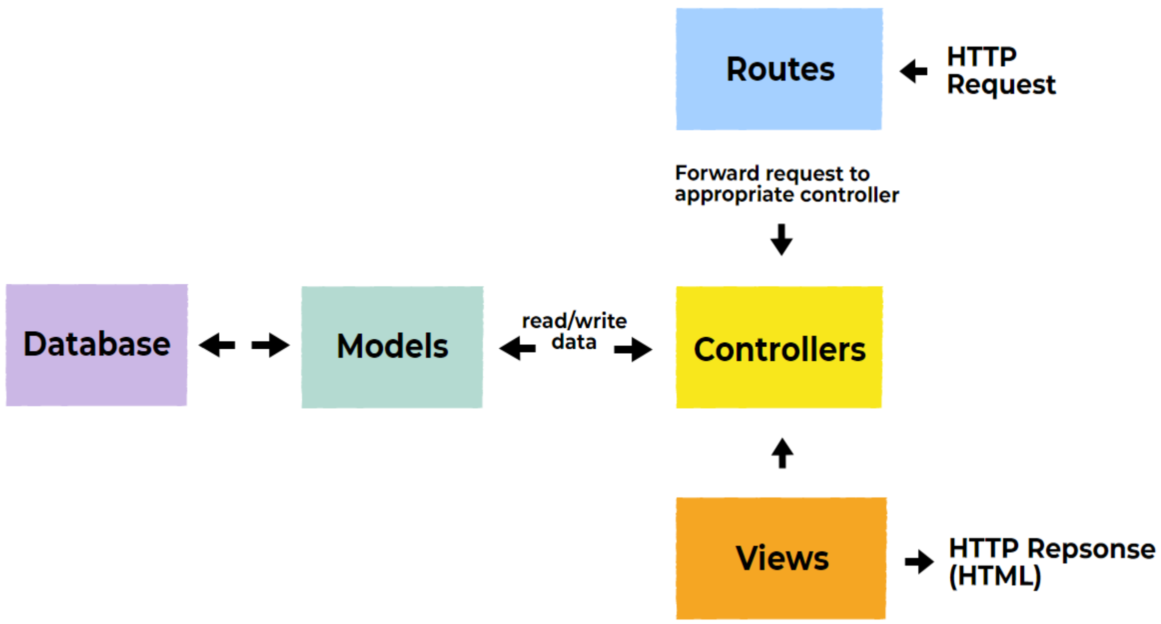


Figure 10: Model-View-Controller-(Routes) Framework (Redrawn from: Reenskaug, 1970)

To structure the script of the online application, the MVC framework will provide the connection between the front-end (*view*) of the application, the back-end (*controller*) of the application, and the database (*model*). It is a logical separation of concerns, making sure different parts of the code do different things, and it is clear which part is responsible for what.

The *model* is the part of the script that is responsible for representing the data in the script and allows us to work with that data. The *model* handles: saving data, fetching data to or from a file, and storing it in its connected database. The *view* is responsible for rendering the right content in the HTML (Hyper Text Markup Language) documents and sending it back to the user. This is decoupled from the script of the online application and handles just light integrations, regarding the data that is needed to generate the *views*. The *controllers* are the connection between the *model* and the *views*, as is depicted in Figure 10.

The *controllers* are the middleware, meaning that the *views* and the *models* can work together inside the script of the application. The *controllers* work with the *model*, saving the data or triggering the saving process. The *controllers* are the middleware, meaning that the *views* and the *models* can work together inside the script of the application. The *controllers* work with the *model*, saving the data or triggering the saving process. This makes it possible to pass the data, that was fetched, to the *views*. An extra component of the MVC framework is *routes*. *Routes* are part of the *controllers*. *Routes* define in the script upon which path which *controller* code should be executed. The *controllers* need this to know to define with which *model* to work with and which *view* to render.

This MVC framework is a pattern that is highly focussed on the middleware, the *controller*. The *controllers* define the logic in the middleware and, therefore, the connection within the online application between the components (Pop, et al., 2014; Masoud, Halabi, 2006; Shan, 1989). The MVC framework will be implemented in the back-end of the online application. The API will be the connection between the back-end and front-end of the online application. This will be described in the next sub-chapter.

4.4.3 | Application Programming Interface

The API can be seen as a messenger, it receives a request, tells the system what action to execute, and sends back the response. For example (Figure 11), you want to buy building materials at a construction shop. You want to buy screws but are not sure which wants to buy. On the desk, there is a list available with which screws are available in that shop. The screws are stored in the warehouse, which is not accessible for you. The only thing that is missing to complete the order, is the shop employee. You order the screws at the employee. He or she goes to the warehouse, collects the screws, and returns your screws. The warehouse, in this case, is the database. You are the customer/user, which goes to the shop to view the possibilities. The employee is the controller, he or she decides if the order can be completed or not. The employee is the API. In this case, the employee and the warehouse are visible. But in the case of the online application, the API and the database are not visible for the customer. The API (employee) is the connection between the back-end (path to the warehouse) and the front-end (shop) of the application. In the database (warehouse), the data (screws) can be stored and be fetched as well, but by using an API.

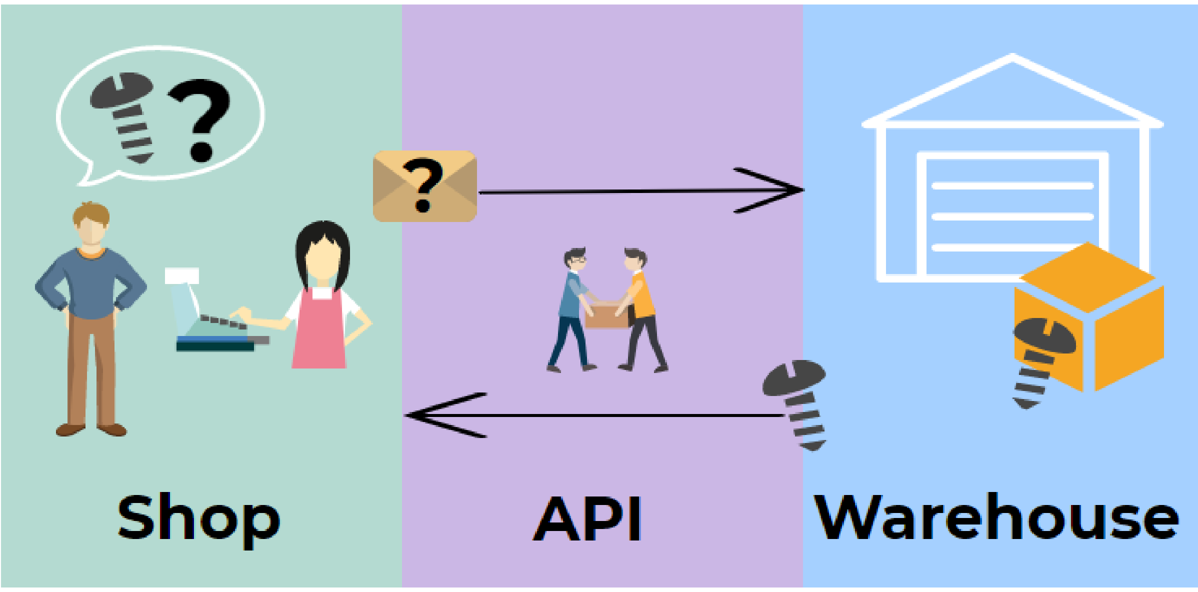


Figure 11: The visualization of the API example

The back-end of the online application shall be developed using Node.js. Node.js is an open-source platform that makes it possible to use the Javascript program language, outside of a web browser. Javascript makes it able to manipulate the web page as it presented in the web browser. It provides small scripts that can run in a web browser. Javascript is widely adopted as the most commonly used program language for web development and the JSON exchange data format can easy be applied (sub-chapter 4.2). An interactive user interface must be developed in the web browser, to fulfil the needs of the end-users. Using the Node.js platform, it provides the opportunity to access the Javascript (outside of the web browser), make changes, and finally be able to run in again in the web browser. Node.js is developed with the Javascript approach (the .js in Node.js is Javascript), which makes this connection possible. Node.js is an extension on Javascript but has more features which are not able when only using Javascript. The possibility to access Javascript outside of the web browser, makes Node.js a deliberate choice for this research. Nodejs allows you to run Javascript in your computer and add functionalities and features that were formerly not possible by just using Javascript (Node.js, 2020).

The features and extra functionalities added to the Node.js software, to increase the possibilities with Javascript, are described in Section 6 (sub-chapter 6.3). In the next sub-chapter (4.4.3.1 - 4.4.3.3), the pattern of the API, using the MVC framework, will be explained in-depth as well as the connection between the component of the MVC framework with external software tools. The MVC framework is already briefly described (sub-chapter 4.4.2), but step by step will now be described how the three components work together within the scope of this research.

4.4.3.1 | Controller

Once the user enters the web browser (Google Chrome, Safari, etc.), he or she requests for a specific web page. Depending on which web page he or she asks for, an URL (the address of a worldwide web page) message is sent to the server, that is running in the back-end of that web domain. The server in the back-end receives and responds to the information that is requested. In this research, the web server is the API that provides that the web domain can run in the cloud. Once the user asks for a URL message, he or she actually sends an HTTP request (Hyper Text Transfer Protocol). This is a protocol that the browser uses to transfer data between the API and the computer that is used. This data needs to be fetched, and therefore sends an HTTP request to receive images, text, pages, etc. Once the user is requesting for a specific page in the web browser, the API will send all the requested information to the *controller* of the API. The *controller* is responsible for handling the entire request from the client. The *controller* will tell the API what to do with the request. It acts as the middleman, between the two other components, *model* and *view*.

A sidestep between the HTTP request from the web browser and the *controller* is the API *routes*. *Controllers* use *routes* as middleware between the HTTP request and actions that need to be executed. The *routes* match the requests with the right paths in the script. The *routes* point the direction, the guidelines for the *controller*, that enables the *controller* to use the right path in the back-end. Without the *routes*, the *controller* is not able to use the right path and the request will end up as a failure. The *controller* and the routes do not contain very much code in the script. In Figure 12, the first steps in the connection between the web browser, the HTTP request and response, and the connection between the *routes* and the *controller* are depicted.

Internal Database

Some internal storage is acquired as well. The outcome of the BIM model data juxtaposition will be multiple templates (Section 5). The analysis will result in a certain list of characteristics that are available in a BIM model. Creating a template with all the available and/or required characteristics for the objects in the BIM models, results in the ability to export a project report with the same information input for every project that is uploaded into the online application. Therefore, an internal storage needs to be facilitated where these templates and other files, images, text documents (logos, background information, images used for the design of a webpage, etc.) can be stored. Therefore, the *controller* has to have the possibility to interact with the data from the databases, as is depicted in Figure 12.

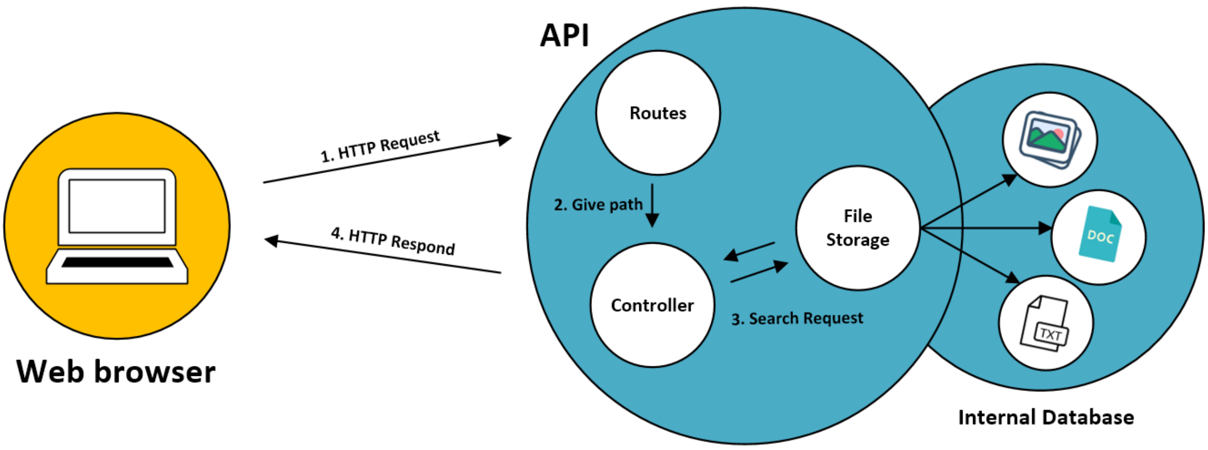


Figure 12: Connection between web browser and the API (controller & routes)

4.4.3.2 | Model

After the *controllers* receive the HTTP request in the right path from the *routes*, the *controller* asks the *model* for information, based upon this HTTP request. The *model* is responsible for handling all of the logic of this request. The *controller*, as mentioned before, is just the middleman. It does not add or change the request or response, it just sends through the message that is received. The *model* is responsible for the interaction with the request. The *model* decides whether this request can be fulfilled or that it will end as a failure, once the requested information is not available. The *model* can send information back to the *controller*, because the *model* interacts with a database. The interaction with the database provides the handling of validation, saving, updating, and deleting data. The *controller* will never interact directly with the database, it will only use the *model* to perform these interactions. To clarify, the *controller* never worries about how to handle requests and responses, it is just the messenger. The *controller* only tells the *model* what to do and responds whatever is returned by the *model*. This means on the other hand that the *model* never has to worry about handling the user requests. It just responds to the request, and if it is possible, it will send back the information available. Whether this fulfils the complete request or just partly. What to do with a failure or success is managed by the *controller*. The *model* only interacts with the request of the *controller* and the data in the database.

The *model* from the MVC framework will be connected to an external databases to meet the requirements of the online application that have been set (sub-chapter 4.4.2). The *model* in the API will interact with an external database to store and fetch data.

External Database

The reason you want to connect your application with a database software tool, is because you want to store data efficiently and retrieve it as well. External database software is developed to store lots of data. They are developed and used because the application would become very slow once a lot of data is stored in the internal database of the app. Therefore, the external database software will help to store countless amounts of data, be able to retrieve it, and still enables to not slow down the online application. It also enables searching for single pieces of information without reading the entire file. In choosing a database software tool, a distinction must be made between SQL (Structured Query Language) and NoSQL (Not Only Structured Query Language) databases. The queries are commands that are used to interact with the database. Every command can be different, but it is a function which enables requesting or retrieving data from the database.

To decide which database is best applicable for this research, first, the difference between an SQL and a NoSQL database needs to be clarified. The objectives are to make the data that is stored, easily accessible and available, and after all, the data flow needs to go fast. For the SQL database, MySQL is very often used as a software tool for this type of database. MongoDB is the most popular software tool for the NoSQL database. An SQL database works with tables and columns that define that table. The rows in the tables where the data is added are called records. An important aspect that distinguishes SQL from NoSQL databases, is that the tables can be related to other tables. An SQL database works with relations and has a strong data schema. The strong data schema means that for each table it is clearly defined how the data should look like and which fields are involved. It is a strictly defined schema and all data in the table must fit the schema.

A NoSQL database, on the other hand, has other strengths that enable itself to be less strict with relations or strong data schemas. The NoSQL database does not work with tables but calls it collections. Each collection does not contain records, but documents. It is the same, but different type of terminology. The NoSQL does not have a strict schema, two documents in the same collection can have different structures and still belong to the same collection. On average the same fields of data storing are used within the application, but the possibilities to store documents that are generally equal but still differ in some fields must be possible. And especially on scalability, the NoSQL database has the main advantage. The ability to scale the servers, when there are multiple or very complex joins, the SQL database reaches a limit, where the NoSQL database easily can expand his server horizontally or vertically (horizontal means multiplex connected servers; vertically means extending the memory capacity). This gives the NoSQL database a great benefit over the SQL database (Gafaar, 2017; Shen, 2019).

Combining the pros and cons of both databases to determine which database is best applicable for this research, the NoSQL database shall be used for storing the data of the online application. The projects that will be uploaded need a connection with the products that will be added to the project, but the projects do not need a relation to other projects. Also, the characteristics of the products can differ, due to the level of information differences within BIM models. Therefore, a schema-less database is necessary to add information about the products in which it is uncertain what is available. Furthermore, the data will not flow across multiple tables but will be nested in a few collections. This therefore results, that the preference for this research will be for the NoSQL database. Finally, the scalability expansion is a great benefit for the online application. Thousands of projects can be uploaded, which means an incredibly large amount of complex data. The NoSQL servers assure that this is possible and data requests and responses still go fast (Gafaar, 2017; Shen, 2019). In Figure 13, an example is visualized of how the class diagram and therefore the structure within the NoSQL database could look like. The project owner can contain zero or multiple projects, but a project can only have one owner. A project can have zero to multiple products that can be reused. And the products cannot exist on their own. Without being connected to a project, no product can be added to the online application.

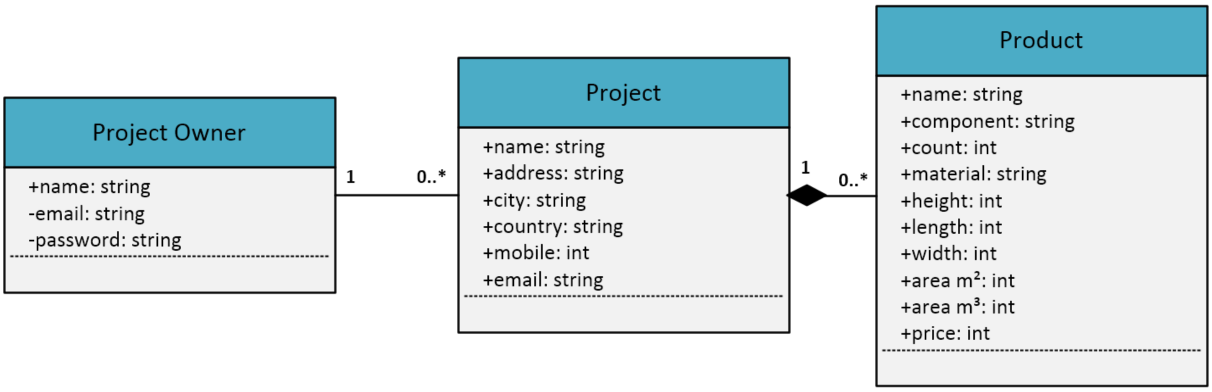


Figure 13: Class diagram of the possible NoSQL database structure

MongoDB

A database solution for the NoSQL database approach, as is previously mentioned, is MongoDB. This is a very efficient and popular software tool to run NoSQL databases. The core of the MongoDB software is that it is built for large scale applications, to quickly query data, store data, and interact with data, which makes the software tool fast with large amounts of data input. MongoDB uses the JSON program language to store data in the collections, so it follows the Javascript object notation format. This makes it easily applicable for the online application (Figure 14), because it runs Javascript using the Node.js software. The data in the collections can contain arrays, strings, numbers, etc. (MongoDB, 2020). This flexibility is necessary, because it is not defined yet what kind of data will be stored in MongoDB.

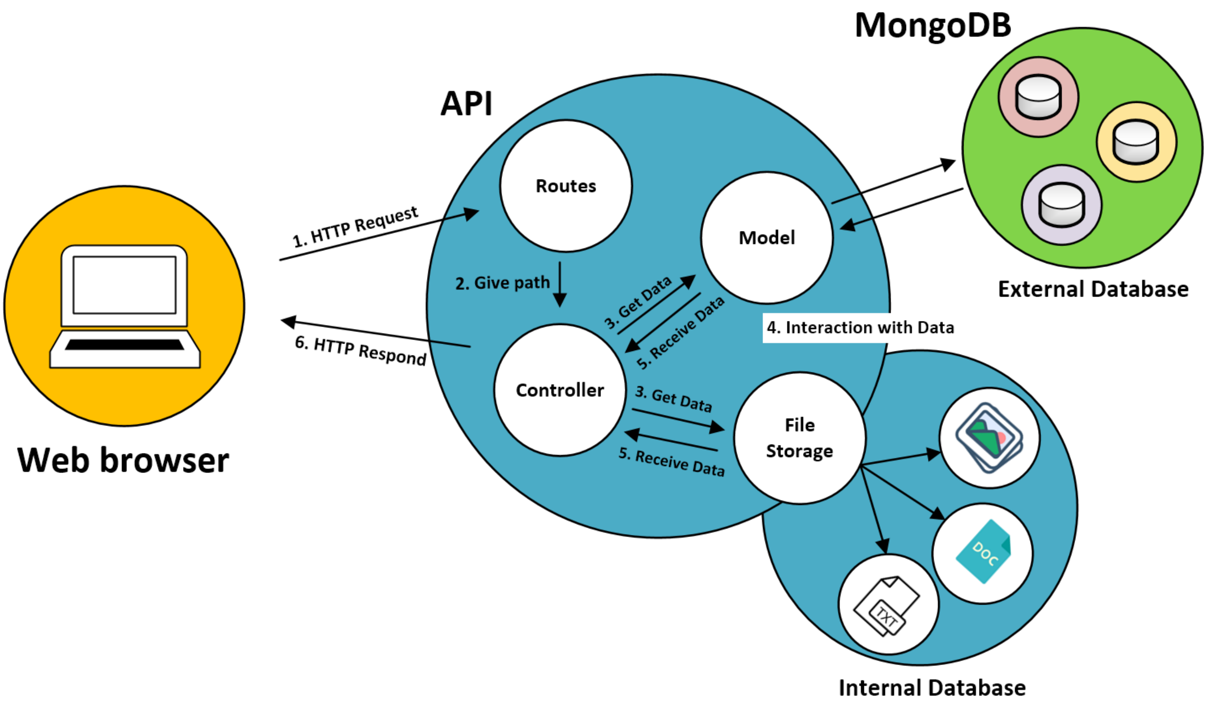


Figure 14: Extension of the API schema with the external and internal database

4.4.3.3 | View

The third and final component is the *view* (Figure 15). Before the data is received from the *model* and can be sent to the user on the web page by the *controller*, the *controller* first needs to interact with the *view.* During this interaction, the data will be rendered by the *view*, before it will serve as the HTTP response. The *view* is only concerned, as its name implies, in how to present the information that the *controller* sends to the webpage. The *view* is a template file that dynamically renders the HTML files. The *view* does not care about how to handle the data, as is similar to the *model*, but only cares about how the data will be presented. After the data is received by the *view* and rendered into its final presentation, the *controller* will handle the final step and responds to the user. An important notation about the MVC framework is that the *model* and the *view* never interact with each other. The interaction between both components is only done by the middleman, the *controller*. In Figure 16, the complete MVC framework is depicted in how the pattern is implemented in this research. Every step from the HTTP request to the *routes*, through the *controller, model,* external and internal database back to the *controller*, to render the data in the *view,* to respond the requested data with a user-friendly presentation back to the end-user in the web browser.

The *view* will transform the complex program language in the back end of the API into a Graphical User Interface (GUI). The GUI is a user-friendly interface that creates a better layout of the online application in the web browser. The style sheeting of the single web pages in the back-end in the *view* is done, using Cascading Style Sheet (CSS). The CSS language provides possibilities added to the HTML document to upgrade the layout. With CSS, you can change font sizes, background colors, add images, change text container styles, etc. All is necessary to create a user-friendly interface that evolves in returning customers to the online application. A not user-friendly, not attractive application will scare off customers and will therefore ruin its possible opportunities on the market.

A final addition that is used which is connected to the HTML *view* pages, is that for this research Express.js (EJS) is used. The EJS software packages provide more possibilities than just using the HTML coding (Similar to the usage of Node.js over Javascript). EJS is a web application framework for Node.js, and therefore perfectly applicable to the design of the online application for this research. EJS has some features, which are not possible with HTML, that create cleaner Javascript code in the back-end of the online application. For instance, you can add navigations bars and footers to the pages, which refer to the same EJS page in the script. By excluding unique coding and replacing it with single navigation lines, the pages get less complex. The program can run faster, because it does not need to read the code over and over again, which can only result in errors that will let the system crash.

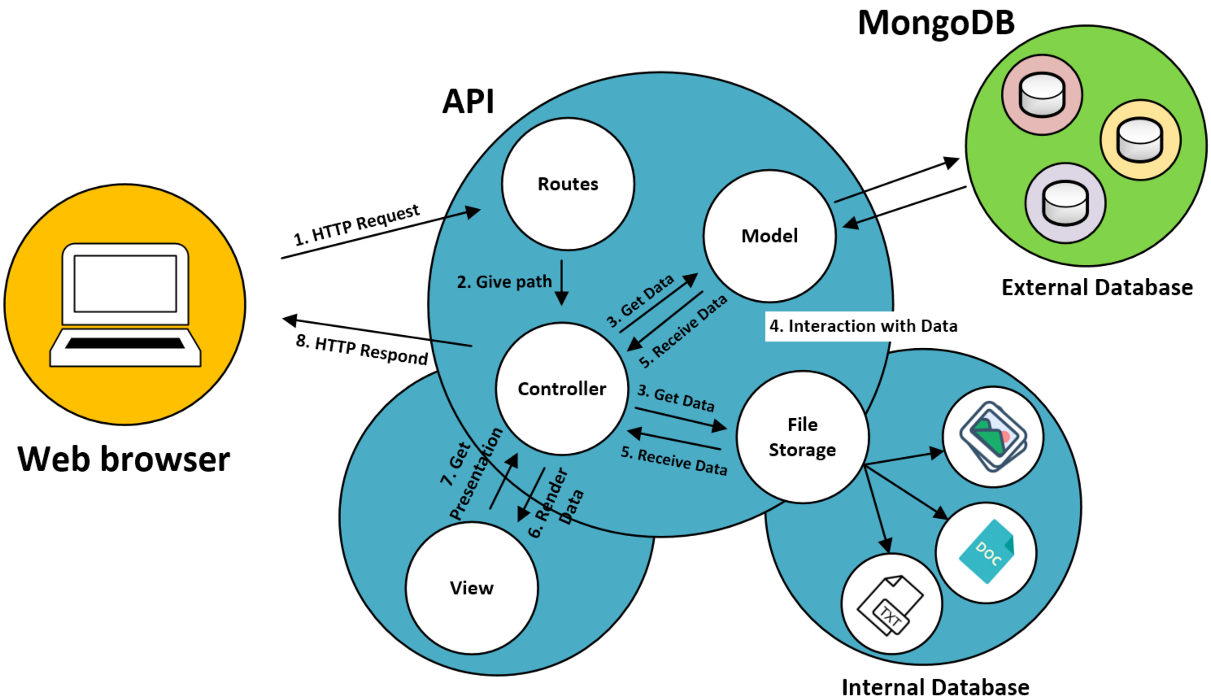


Figure 15: Complete implemented MVC framework in the research

4.5 | Conceptual model

To summarize the conclusions that are defined in the Methodology and describe what can be expected in the following sections of this research, a conceptual model is composed (Figure 16). The conclusions are based on the problem definition, conducted literature review, the RAD method, frameworks, and the variety of software tools that will be used to achieve objectives of this research. By ending Section 4, the theoretical part of the research is executed, in which Section 5, 6 and 7, will be the practical part of the research. Each section will be highlighted now, and a description is given in what can be expected in Section 5, 6, and 7.

In Section 5 (Non-graphical information analyses), the desired characteristics of the objects in a BIM model will be determined, which are relevant for its presentation in the webshop. A circularity-based guide will be customized, which clarifies what parameters and properties are necessary to give a quality to the reusable building materials based on its availability of non-graphical information. This circularity-based guide will be compared with the BIM model data juxtaposition. This analysis shall juxtapose three BIM models and shall analyse what data is matching, and finally which data is relevant for this research to be retrieved from the BIM models. The BIM model data juxtaposition is connected to the internal database, as is depicted in Figure 16. The result of the analysis will be a Solibri Role and a Project Report Template, which will make sure that the right data will be retrieved from the BIM models, once a project is uploaded unto the online application. This will be explained in detail in the next Section 5, but the connection is already visualized in the conceptual model.

Section 6 (System Design) will be the start of the RAD and will execute three of its four phases. The requirements, design phase and the development phase will be devised in here. The requirements will be listed before the design phase can start. The requirements will be set, based on the perspective of the two end-users: the project owner and the webshop customer, as well as the demands of the developer. The design phase shall be a visualization of how the requirements can be met and to see which requirements must be adjusted or changed once the scenarios are drafted. Also, an activity diagram will be developed which highlights the paths of different web pages. The development phase will be the most important phase which consists of the connection between the web browser, the API, and the databases. The MVC pattern will be the fundamental part. The development of the back-end and its connection to the front-end in the web browser, and with the NoSQL MongoDB software tool. The development phase, and the three domains involved, are highlighted on the right side of the conceptual model. This to provide a better understanding of how the MVC pattern, event-driven architecture, software tools, and requests and responses will be handled by the API. As was defined in sub-chapter (4.4.1), a clear system architecture diagram is needed to be able to keep track of the components and the overall structure during the development process. Therefore, this diagram is highlighted to substantiate this requirement. After the development phase, the prototype is ready to be tested.

In Section 7 (Proof of Concept & Validation), the prototype will be tested and will be validated if the online application will prove the concept of achieving the connection that was tried to be realized to stimulate the reuse possibilities of building materials. This is done, by connecting the non-graphical data with cloud-based software. The prototype will be tested using the Alpha testing method, in which a simulation by end-users will describe the results of the online application and if needed, optimization will be adjusted to the online application. Section 7 will end with an evaluation of the desired non-graphical BIM data that is retrieved from the BIM models and it will be validated if the research aim is achieved or which steps should be undertaken to achieve this goal in future research. The usability, extensibility, and contribution to the market will be discussed as well.

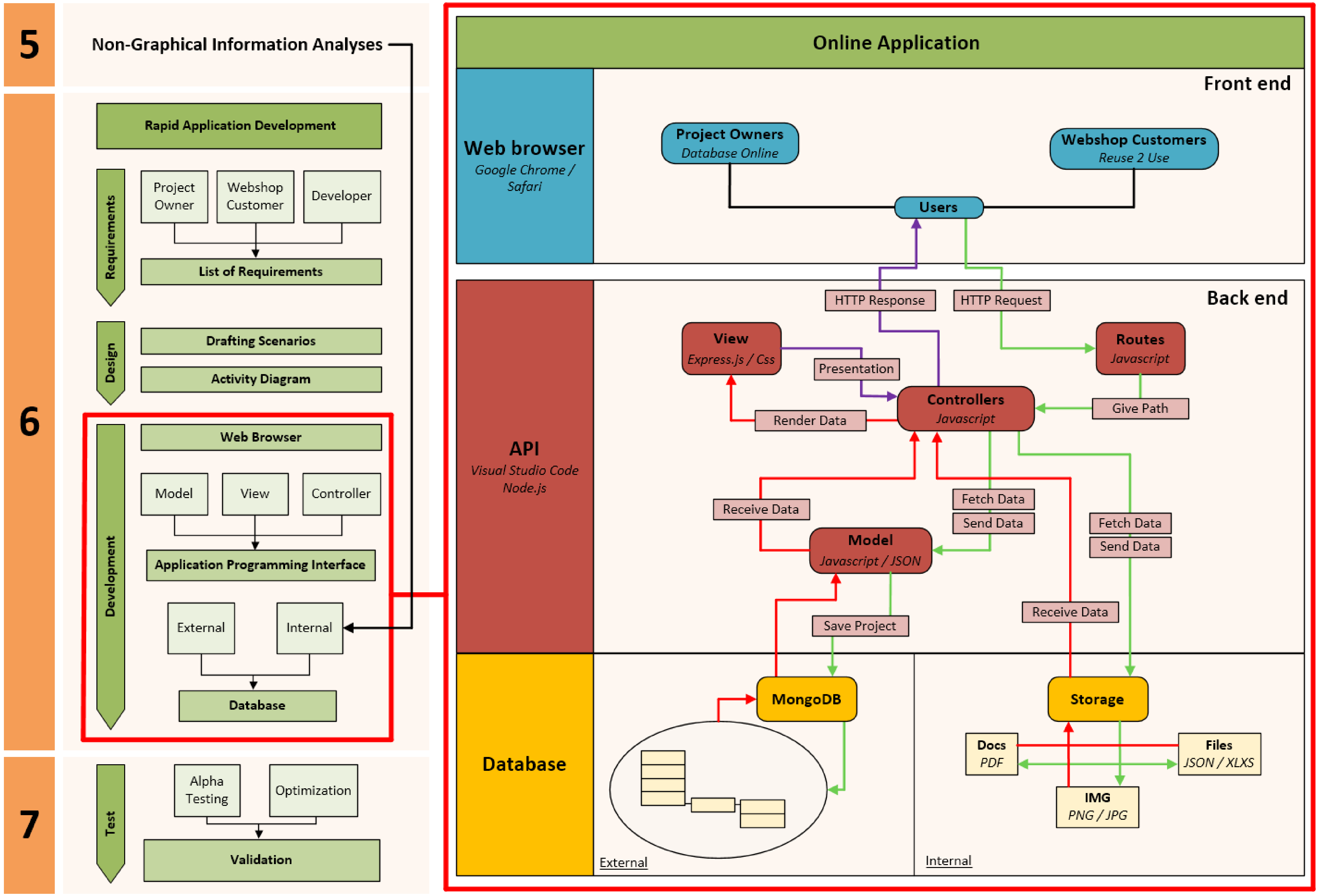


Figure 16: Conceptual model, with the architecture system diagram of the MVC pattern highlighted

Section 5

Non-Graphical Information Analyses

5 | Non-Graphical Information Analyses

The first section in the practical part of this research is the non-graphical information analyses. This analysis will be divided into two analyses. The first analysis is concentrated on the development of a circular-based guide that will be able to request for the desired exchange requirements that are needed to stimulate the reuse of building materials. In the first sub-chapter (5.1) this circularity-based guide is proposed, in which it will be explained in more detail in sub-chapter (5.2). Based on the conducted literature review (Section 2) and external resources, a list of desired exchange requirements for this guide could be composed. Once the desired output of the BIM model is defined, the second analysis of this section starts. In the BIM model data juxtaposition (5.3), three BIM models will be analysed based on their value of non-graphical information that is within the scope of this research. The input for the analysis is described, the data is analysed, an Information Take Off is conducted, and finally, this report is analysed as well, to validate if the extracted parameters do contain the value that is required for this research. Finally, the output of the BIM model data juxtaposition will be compared with the exchange requirements of the circular-based guide (5.4). Concluded can be if the BIM models do contain the desired non-graphical information that is within the scope of this research. Once it is stated which data can be retrieved from the BIM models, that serve as input for this research, the final output of the non-graphical information analyses is determined in subchapter (5.5). The output of this section will be two templates, which will be included in the internal database of the online application and will provide the opportunity to retrieve the requested data, in a customized environment, once the project owners will upload their project unto the online application.

5.1 | Proposed Circularity-Based Guide

As was stated in the literature review (Section 3), especially the 'I' in BIM increased the value of the projects since the implementation of the BIM methods over the years. This increased the possibilities that are able with BIM models by not only having the graphical information of the project, but especially the non-graphical information of every object and its characteristics and properties available. Based on this type of data, project life cycle information can be retrieved from the BIM model, which will help the construction sector with its transition towards a circular built environment. Concluded was that, being able to retrieve the right non-graphical information from the BIM models, this specific data input must be requested for. Receiving the desired output of the BIM model, especially to conduct an LCA or be able to get insights in the full life cycle of the building, specific exchange requirements must be set. By requesting for these specific exchange requirements, the circular-based desired values can be captured from the BIM model. The only claim is that these must be pre-defined before the design phase starts, otherwise these specific requirements will not be included in the BIM models. An extra check to make sure the data in included during this phase is therefore advised as well.

A structured and universal guide will help to request for these specific exchange requirements, ensures the involved stakeholders will include these exchange requirements, a guide is proposed in sub-chapter (3.8). The basis-ILS (BIM Loket, 2017) (Appendix I) is especially developed to increase the value of the BIM models in the collaboration process of construction projects. These guidelines ensure that certain information in the BIM model will be included, based on a set of universal requirements to get a structured, efficient and effective data input, which is applicable by all stakeholders involved. These collaboration exchange requirements will increase the quality of the complete process of a building project. The basis-ILS should be mandatory in the AECO industry, to standardise the information input in the BIM models, in which every company shall design the model based on the same universal and structured naming of entities, classifications, and objects. The basis-ILS can be extended by a customized

ILS (sub-chapter 3.9), which will pre-define the exchange requirements based on specific project goals. Due to this possibility, this research proposes (sub-chapter 4.2) to develop a customized ILS to capture the circular-based value of a BIM model, based on the desired LOI. This circularity-based ILS will be developed, which clarifies what parameters and properties are necessary to give a certain quality to the reusable building materials based on its availability of non-graphical information. By pre-defining the exchange requirements needed for the output of reusable building materials, the moderators are obligated to include these values to the objects in the BIM models. Therefore, a high LOI can be achieved which will stimulate the reuse potentials of the building materials after the building is deconstructed. The circular-based guide, with the desired non-graphical information output, will be defined in the next sub-chapter.

5.2 | Re Use Index – ILS

The circular-based guide, which will be inspired on the guidelines of the basis-ILS, will be the Reuse Index – ILS. The Reuse Index – ILS shall contain exchange requirements that will request for as many as possible non-graphical information parameters that are desired for the output of reusable building materials. These exchange requirements are based upon the conducted literature review (Section 2) as well as external resources:

- Circular design (Crus Rios, Grau, 2020).
- Design for Disassembly (McLellan, 2019; Rios, et al., 2015; Deller, et al., 2019).
- The circular built environment initiatives (BAMB, 2020; Madaster, 2020).
- Building Circularity Indicators (Vliet, 2018).
- Material Circularity Indicators (Ellen MacArthur Foundation, 2015).

The exchange requirements for the Re Use Index – ILS will stimulate the reuse potentials of the building materials, after they are deconstructed from a building. The exchange requirements are specifically focused on retrieving only the data that will stimulate the reuse potentials of the building materials, general exchange requirements e.g. project information, general name, identification, materials, location, and building physics are not taken in consideration in the Re Use Index – ILS since they are already requested in the basis-ILS. The building materials are defined as objects in basis-ILS in which they will be defined in the Re Use Index – ILS as objects as well. To frame the terminology that is used in the Re Use Index – ILS, the following terms can be distinguished. (1) The object is defined as the complete building material. A building material can have different (2) components, and the components can be made out of different (3) materials. For example, a door (1) is the object. The door has the components: door, window, door handle, and joints (2). The materials are wood, glass, aluminium, and steel (3). The exchange requirements are divided into three main topics: Object Properties, Object Circularity Indicators, and Object Dismantle Indicators. The three topics and their specific exchange requirements will be described in the next sub-chapters.

5.2.1 | Object Properties

The object properties will define the characteristics of the building materials. This does not have a circular-based approach but gives a complete indication of how the building material is composed due to different components, finishing, and accessories. The exchange requirements will be listed with a short description and an example is given of what is expected as non-graphical information input for that specific requirement. The object property requirements are determined to get as much as possible information about the characteristics of the objects. If the graphical information of the BIM model is not on a high LOD, the non-graphical can still describe the object very detailed, which gives a high LOI. The object properties aim to describe as much detailed information about the production, components, materials, finishings, and dimensions of the object as possible. If this information is added to the objects, the value capturing of the objects from the BIM model is very useful as input for BIM models if the object gets reused in future projects.

Table 1: Object Properties Exchange Requirements (Re Use Index - ILS)

Exchange Requirements	Description	Example
Production Date	Release date of the object after its production process.	14-10-2001
Manufacturer	Name and contact information of the manufacturer of the object.	Stamo BV
Warranties	Add documentations of the warranties of the object if these are still valid.	Is expired
Components	Describe the components which composite the object.	Door, window
Material Types	Describe the types of materials that are used in-depth.	MDF, Hr ++
Finishings	Describe the finishings of the components.	Interior paint, metal coating
Finishing Color Codes	Describe the color codes of the finishings.	Ral 9001
Accessories	Describe the accessories that are applied to the object.	Handle, joints
Height	Describe the height of the object in mm.	2300 mm
Length	Describe the length of the object in mm.	-
Width	Describe the width of the object in mm.	930 mm
Thickness	Describe the thickness of the object in mm.	40 mm
Area m^2	Describe the m^2 of the object.	2,14 m^2
Volume m^3	Describe the m^3 of the object.	0,85 m^3

5.2.2 | Object Circularity Indicators

The object circularity indicators give a more in-depth description of the objects, based on the different feedback loops in the circular built environment (maintenance, reuse, remanufacture, and recycle). If these exchange requirements are available in the object, the reuse potentials will increase enormous, because this defines the impact the reusable building materials have on the environment and which parts can be reused, recycled, or naturally decomposed. It is in fact a personal library of the object, from his previous usage in projects up to its expected end of life. The more detailed the description is, the higher the value can be for a LCA of the complete building and the value it has for the operations & maintenance phase, based on the information that can be extracted from the BIM model.

Table 2: Object Circularity Indicators Exchange Requirements (Re Use Index - ILS)

Exchange Requirements	Description	Example
Used in Previous Projects	The number of times and years the object is used in previous projects.	Used in 2 projects
Expected Durability	Define the expected durability of the object.	41 years
Depreciation	Define the expected depreciation of the object without maintenance, based on a 10-year scale.	23%
Average Status	Describe the status of the object compared to an industry average object of a similar type.	Status is average
Composition of the Object	Describe the composition of the components of the object based on its virgin materials.	Timber fibers, quartz, sand
Replaced Components	If parts of the object are replaced, describe the parts that are replaced and if this is done with: virgin materials, recycle materials, reused components.	Nothing is replaced
Destination after use	Describe the destination after use of what part will go to landfill; what part is for recycling; which components are for reuse.	Components can both be reused

5.2.3 | Object Dismantle Indicators

The object dismantle indicators indicate how easy the object can be dismantled to its components and materials. These indicators do not assess how easy or not the building material can be disassembled from the building but describe how the object itself can be dismantled. When describing these exchange requirements, an indication is provided in how easy parts can be maintained or replaced during its life cycle.

Table 3: Object Dismantle Indicators Exchange Requirements (Re Use Index - ILS)

Exchange Requirements	Description	Example
Accessible Connections	Describe if the connections between the components of the object are easily accessible.	Easily accessible
Material of Connections	Describe of which materials the connections between the components are made.	Screws, acrylic construction kit
Type of Connections	Describe the types of connections that are used, is just one tool needed or multiple?	Screw machine, utility knife
Human-Scale Components	Describe if the components can be replaced based on human-scale with	Can be replace manually

	standard mechanical equipment or does is need to be remanufactured to replace the components?	
Interchangeability	Describe if the components are easily interchangeable due to the use of standardized components, or are they uniquely designed and produced?	Both components are standardized

5.3 | BIM Model Data Juxtaposition

In the first part of Section 5, it is defined which non-graphical information is desired to retrieve from a BIM model to be able to stimulate the reuse of building materials. The object properties, object circularity indicators, and object dismantle indicators create a complete list of parameters customized on the non-graphical information need for the reuse potentials of building materials. This list is the best-case scenario if it can be completed based on the retrievable data from the BIM models. To be able to determine if these parameters are already available in a BIM model or to analyze what other relevant circular-based non-graphical data there is available in BIM models, an analysis will be conducted.

In this analysis, three different BIM models, that were developed in construction projects in the past years, will be juxtaposed, compared and analyzed. The three BIM models are provided by the supervising company BASED BIM Management & Consultancy. Their clients approved that the models could be used as input for this research. The BIM models represent the refinement of the average execution of the BIM models in the AECO industry, as it can be expected within contemporary projects.

The analysis of the BIM models shall be focused on the non-graphical information, the graphical information will not be taken into consideration. The graphical data of the object can, in a later stage of the research, be rectified by adding an image of the building to the webshop of the online application. The analysis is focused on the LOI, which consists of the characteristics, dimensions, materials, finishing, etc, of the objects. The non-graphical information that is included in the BIM models will be described and compared with the non-graphical information of the other BIM models. Only the usable data will be elaborated on, the unusable data will be deleted out of this analysis.

The exchange format that has been used in the three construction projects, to exchange the BIM model, is an IFC file. The BIM-based Model Checker that will be used for the analysis is the Solibri Model Checker. Solibri is widely used as BMC in project teams, supports the IFC format, and has a user-friendly, easily accessible interface. Especially, for people with less experience with BMC software. The SMC will, therefore, used as a software tool for this analysis. The SMC can provide a clear insight into the non-graphical information that is attached to the objects in the BIM models. The characteristics and property sets will be reviewed and extracted from the BIM models. From a few randomly chosen objects, every single parameter will be extracted and described in an Excel Spreadsheet. This, to easily compare the parameters of the objects with the ones from the other BIM models, to conclude similarities and dissimilarities. The non-graphical information that is in line with the Re Use Index – ILS and/or is relevant for this research to extract will remain. Not matching data will be deleted. Data that might be relevant will be revised if it will stay in the analysis or if it finally will be deleted as well.

Finally, the analyzed parameters that are relevant and available in the BIM models can be extracted with an Information Take Off (ITO). This is a tool from the SMC to export data from a BIM model into an Excel .xlsx file. This ITO Report will be the output of the BIM model data juxtaposition and will be offered in two different formats: Solibri Role and the Project Report Template. Every step and definition will be explained in detail in the next-subchapters, but an overall visualization of the steps, conducted in the BIM model data juxtaposition, is already depicted in Figure 17.

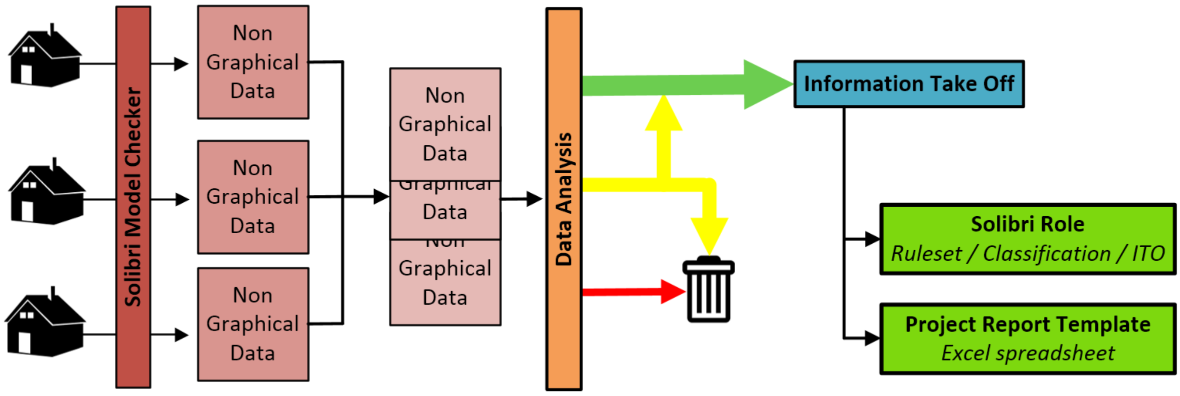


Figure 17: BIM Model Data Juxtaposition Steps

5.3.1 | Analysis Input

The BIM models are provided with censoring approval clients from BASED BIM Management & Consultancy. The presented data will, therefore, at all times not be traceable to their clients or the stakeholders involved. The BIM models are developed over the past years: 2016, 2018, and 2019. This makes the BIM models reliable based on the LOD and LOI that on average is used in the AECO industry nowadays. Due to the timeframe of this research, it was decided to analyse three BIM models to create the average perspective on the data input of the contemporary BIM models. The reliability of the data juxtaposition can always be extended by adding more BIM models to this analysis.

5.3.2 | Data Juxtaposition

The first step is to analyse the BIM models, based on the non-graphical data that is available. To be able to execute this step, the BIM models are reviewed with the SMC. For every project, four objects, which are randomly chosen, are in its entirety described. Every single parameter that was attached to this object was extracted and added to an Excel Spreadsheet. In Figure 18, a brief example of this first step is shown.

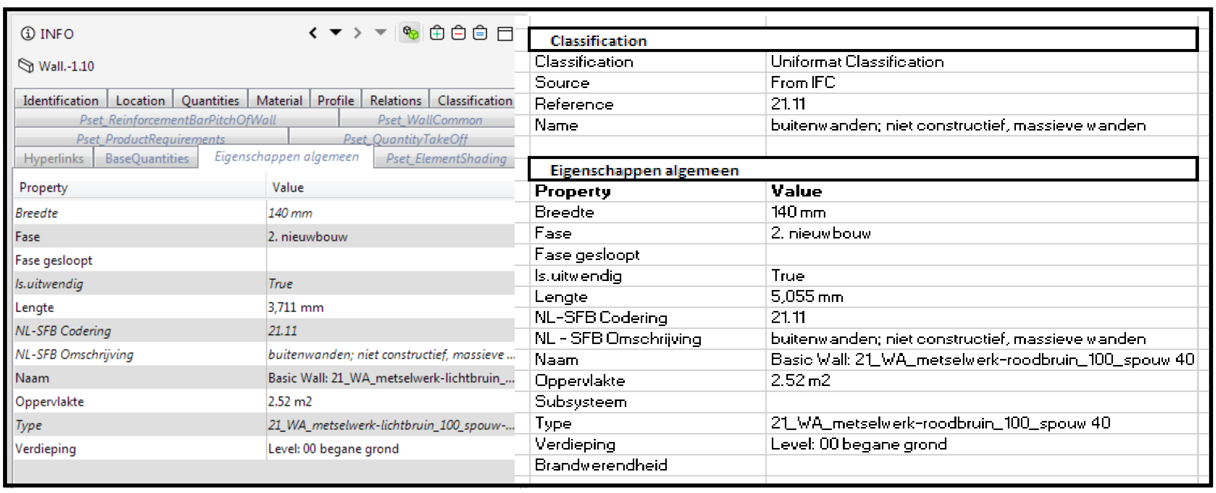


Figure 18: Example of the description of non-graphical data juxtaposition

After a total of twelve objects from three different projects were extracted, the input for the analysis is completed. The complete Excel Spreadsheet with the description of the twelve randomly chosen objects, is provided in Appendix II. The twelve objects were juxtaposed and analysed. In this part of the analysis, three different classifications are distinguished: matching data, semi-matching data, and not matching data (Table 4). The matching data are the fields that have a value in every field of every object (coloured in green). The semi-matching data are the fields that have a value for at least six of the twelve objects (coloured in yellow). In the semi-matching data, the difference of the LOI between the three BIM models can already be noted. The higher the amount of non-graphical data, the higher the LOI that is used during the design phase. The not matching data are the fields that do have a value in 5 or fewer fields, or do not contain a value at all (coloured in red). But also, fields were classified as not matching data if the parameters did not have any contribution to the scope of this research. In Figure 19, an example is depicted in how the value from the parameters is classified. The complete list with classified parameters is provided in Appendix III.

Identification	
Property	**Value**
Model	Censured
Discipline	Architectural
Name	Basic Wall:16_WA_beton ihw 400_gen:3579201
Phase	
Type	16_WA_beton ihw 400_gen:3579201
Type Name	Basic Wall:16_WA_beton ihw 400_gen:3579201
Predefined Type	
Description	
Material	00_Beton_generiek_400 mm
Layer	A-WALL-___-OTLN
System	
Filled	
Building Envelope	True
Geometry	Extrusion
Application	Autodesk Revit 2018(ENU)
IFC Entity	IfcWallStandardCase
IFC Type	IfcWallType
GUID	0Q5Wh_pZLBReCbPD96svMm
BATID	3579201
Model Categories	

Figure 19: Classifications: matching/semi-matching/not matching non-graphical information in the BIM models

Table 4: Classifications used to distinguish the data from the analysis

Matching Data	Semi-Matching Data	Not matching Data
All fields have a value; 100%	At least six fields have a value; 50% <	5 or fewer fields have a value; > 50%. Or the data is not matching for this research.

Identification	
Property	**Value**
Model	CENSURED
Name	44_OOA_Multiplexplaat_trap1:44_OOA_Multiplexplaat_trap1:6079441
Layer	A-GENM-___-OTLN
IFC Entity	IfcCovering
GUID	0Q5Wh_pZLBReCbPD96svMm
BATID	3579201
Quantities	
Property	**Value**
Bounding Box Height	1,365 mm
Bounding Box Length	1,597 mm
Bounding Box Width	290 mm

Figure 20: Matching data classification

After executing this part of the analysis, it was concluded that, out of 240 fields, only 9 fields were classified as matching data. (Figure 20). But, because these 9 fields give too little value to this research, the semi-matching data was validated as well, to analyse if more parameters could be added to the result of this analysis. If the semi-matching data was relevant for the research, it was determined, based on the added value that characteristics of the objects could have. This had to be in line with the stimulation of reusable building materials. After this validation, the following semi-matching parameters were added to the template:

- Component
- Count
- Nl-SfB Code
- Material
- Height
- Length
- Width
- Thickness
- Area m²
- Volume m³
- IFC Type
- Type
- Requirements Category

Component is the brief description of the object, such as wall, door, window, etc.; Count is the total amount of objects in the BIM models which are the same; The NL-SfB (Dutch) coding is added as well, because this code stands for the classification of construction elements (and is also requested in the basis-ILS). It is an international standard, but this is the Dutch version. The code refers to a certain classification used in the AECO industry in the Netherlands; The material is added, because it is the most important parameter in the BIM model. And hopefully gives a value that comes close to the Re Use Index – ILS exchange requirements; The height, length, width, thickness, area m², and volume m³ are the dimensions and will be added as well, since they are also determined in the Re Use Index – ILS and are mandatory to attach to the object when you want to stimulate the reuse possibility of it; IFC Type, Type and Pset_Category are also possible parameters that could retrieve non-graphical information from the BIM model, that is relevant for this research.

Once these semi-matching parameters were added to the list of matching parameters, this list was reviewed again. Several matching parameters were deleted from this list, since they are not relevant for this research: model, BATID, bounding box height-length-width. Excluding these parameters and adding the relevant semi-matching ones, creates the following list:

- Component
- Count
- Nl-SfB Code
- Material
- Height
- Ifc Type
- Pset_Category
- Layer
- Guid
- Length
- Width
- Thickness
- Area m²
- Volume m³
- Type
- Name
- IFC Entity

To be sure every relevant matching and relevant semi-matching parameter was added to this first draft, a second juxtaposed analysis was conducted. This time, three similar objects from the different BIM models. The analysis had the same approach: extracting the parameters to an Excel Spreadsheet. But this time, to ensure if there were not more values matching or semi-matching, the input of the object was not randomly chosen but this time similar objects were juxtaposed. In Appendix IV, this analysis is provided. Three similar objects were analysed: door, wall, and a pipe. What can be concluded from this analysis is that this analysis had much more fields that contained values of matching data, compared to the first analysis. The extracted parameters from this analysis were also classified based on the classifications from Table 4. This second analysis with the classification is provided in Appendix IV. Although, this analysis did contain more values of matching data, unfortunately, not more parameters could be added to the first draft of the template. The extra matching values were not matching to add, within the scope of this research.

What is important to note, after the second analysis was executed, that one of the three BIM models did make use of the basis-ILS during its design phase. An example of two different objects is depicted in Figure 21. But the parameters of the basis-ILS do not contain much added value for the scope of this research. The Re Use Index – ILS is of course, therefore developed.

ILS	
Property	**Value**
Afsluitbaar	
AdresId	
BagPandId	
BagVerblijfsObjectId	
Bediening	
BouwlaagCode	
ComplexId	
Diameter	80 mm
Deurdranger	
Draairichting	
EenheidId	
Exterieur	
GUID	
Hoogte	
IfcExportAs	IfcPipeSegment
KozijncombinatieId	
Lengte	2,840 mm
Materiaal	Metaal
MateriaalAfwerking	
NIsfb4	
Objecttype	Hemelwaterafvoer
Oppervlakte	0.36 m2
Orientatie	
RuimteRelatieId	
TerPlaatseVan	
Vluchtroute	
Breedte	
ObjectAfwerking	
Geisoleerd	

ILS	
Property	**Value**
Afsluitbaar	
BagPandId	63415653
BagVerblijfsObjectId	3533151
Bediening	Handmatig
BouwlaagCode	Level: 01 eerste verdieping
ComplexId	BB-1629A
Deurdranger	False
Draairichting	SINGLE_SWING_RIGHT
EenheidId	OGEH-0022756
Exterieur	True
GUID	63545315454
Hoogte	
IfcExportAs	IfcDoor.DOOR
KozijncombinatieId	351dfa31df1a3sd51f
Lengte	
Materiaal	Hout
MateriaalAfwerking	
NIsfb4	31.31
Objecttype	Deur
Oppervlakte	1.41 m2
Orientatie	NW
RuimteRelatieId	N/A
TerPlaatseVan	N/A
Vluchtroute	False
Breedte	
ObjectAfwerking	
Geisoleerd	

Figure 21: Example of the basis-ILS guidelines used in one of the BIM models

5.3.3 | Information Take Off

After finishing the non-graphical data juxtaposition of the BIM models, it is time to extract the values of the parameters that are listed as being matching/semi-matching and relevant data inputs. Besides the possibility of reviewing the model, execute clash detections by rulesets and be able to communicate, the SMC has another important feature: The Information Take Off (ITO). After the clashes are resolved and design mistakes are communicated and revised, the ITO makes it possible to take off all quantities of the objects that are included in the BIM model. Through this feature, it is very easy to extract certain values the objects from the model, based on the parameters that you want to retrieve from the BIM model. In the ITO, you can add the parameters by yourself, of which you want to extract the value from. Once the ITO takes off all values, it is possible to export the ITO report to an Excel Spreadsheet. With this option, extra calculations can be executed using the Excel file. It is only possible to export the ITO report as an .xlsx file. No other file formats are possible. Based on the change towards cloud-based software (sub-chapter 3.6), an ideal scenario would be if the ITO Report could be exported to an .JSON file. Especially, to develop an online application based on Javascript and JSON file structures. But that will be a requirement for the development of the online application to provide this transition, since it cannot be resolved with Solibri.

The conducted list of parameters, from the matching and semi-matching data analysis, was added to the ITO environment of one of the used BIM models in the SMC software (Figure 22). Once the values of the parameters of all objects in the BIM model were extracted from Solibri into the ITO environment, an ITO report, including this data, was exported to an Excel Spreadsheet. A part of this ITO report is provided in Appendix V.

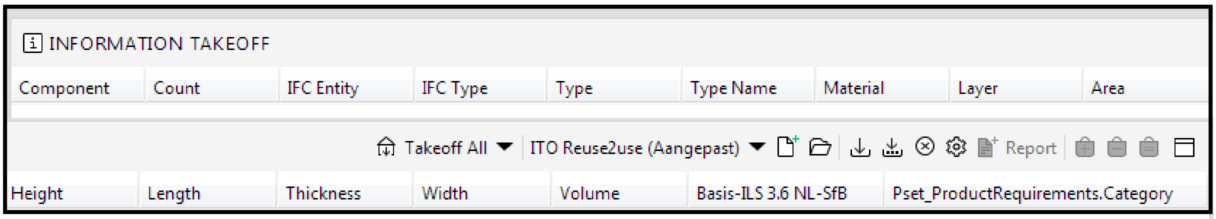


Figure 22: First draft of the ITO Report parameters.

5.3.4 | ITO Report Analysis

The values of the analysed parameters that were extracted in the ITO report were validated in the Excel Spreadsheet. The ITO report analysis aims to validate the values that are extracted now, based on the first draft of the ITO template parameters. If the analysis shows that the values of these parameters are eventually not matching for this research, or similar values of data in different parameters were found, they were excluded from the ITO template. (Some parameters are not included in this ITO report analysis. It is defined that their values contribute towards the scope of this research and do not contain similarities with the values of other parameters from the ITO report).

The parameters are juxtaposed and analysed based on their unique data, to increase the value of the ITO Report. The analysis showed that especially the elements: IFC Entity, IFC Type, Type, and Type Name, had similarities based on their values of data input. As is depicted in Figure 23, the parameters: IFC Entity and IFC Type contain similarities in their values of data input. Stated can be, that the IFC Type values are less complete, because fields are left empty in the ITO Report. Since they are similar in data output, IFC Type is deleted from the template because it is less complete and does not add extra value to the ITO report. The same thing can be concluded for the parameters: Type and Type Name. Type Name is missing values and has similar values of data input, comparing to the parameter Type. This parameter, therefore, is deleted from the ITO Report as well. The parameter Type will be renamed into 'Name' in the next part of this research, which creates an easier understanding of its values in the ITO Report.

IFC Entity	IFC Type	Type	Type Name
IfcDoor	IfcDoorStyle	32_DO_binnenkozijn_Svedex_Match_BO1_2315_930_2640 2	32_DO_binnenkozijn_Svedex_Match_BO1_2315_930_2640 2
IfcDoor	IfcDoorStyle	32_DO_draaideur_Svedex_SL01_opdek	32_DO_draaideur_Svedex_SL01_opdek
IfcEnergyConversionDevice		61_GM_PV paneel_30graden	
IfcFlowTerminal		52_hwa_afvoer_noodoverstort	
IfcFlowTerminal		52_hwa_afvoer_stadsuitloop	
IfcFurnishingElement	IfcFurnitureType	73_aanrechtblok hoek_generiek	73_aanrechtblok hoek_generiek

Figure 23: Comparing the values of the parameters: IFC Entity, IFC Type, Type and Type Name.

The second ITO Report analysis analysed the parameters: IFC Entity, Layer, Pset_ProductRequirements.Category. Figure 24 shows the juxtaposition of the values of these three parameters. IFC Entity contains the highest value and has the most reliable information. The parameter Layer contains, after all, not matching non-graphical information for this research. The parameter Pset_Category is excluded as well because the values are not reliable of specific enough (a lot of values were placed as 'Generic Models', which is not adding any value to the object). Also, values were extracted in the ITO Report, in which they show dissimilarities with the IFC Entity. An example, as is depicted in Figure 24, Windows are placed in the same line as IfcFlowTerminal. Therefore, this parameter is excluded as well.

IFC Entity	Layer	Pset_ProductRequirements.Category
IfcDoor	A-DOOR-____-OTLN	Doors
IfcDoor	A-DOOR-____-OTLN	Doors
IfcEnergyConversionDevice	A-GENM-____-OTLN	Generic Models
IfcFlowTerminal	A-GLAZ-____-OTLN	Windows
IfcFlowTerminal	A-GENM-____-OTLN	Generic Models
IfcFurnishingElement	I-FURN-PNLS-OTLN	Furniture Systems

Figure 24: Comparing the values of the parameters: IFC Entity, Layer and Pset_Category

After conducting the ITO Report Analysis into the ITO Report Template parameters, it can be concluded that the following parameters can have an added value, once they are extracted from different BIM models:

- Name
- IFC Entity
- Count
- Material
- Area m²
- Height
- Component
- Nl-SfB
- Length
- Width
- Thickness
- Volume m³

5.4 | Re Use Index – ILS vs. BIM Model Data Juxtaposition

It can be concluded that the desired non-graphical information, that could hopefully be retrieved from the BIM models, is not available in the BIM models. Or at least, not in the BIM models that are used for this research. A few parameters from the result of the BIM model data juxtaposition are in line with the desired exchange requirements: Name, Component, Material, Height, Length, Width, Thickness, Area m² and Volume m³. But most of these parameters and exchange requirements are based on the dimensions. Not one single circular-based parameter was included in the BIM-Models. Not within the matching, semi-matching, or not matching data input.

The desired non-graphical information would be beneficial for this research, to elaborate on during the development of the online application. But this non-graphical information cannot be retrieved from the BIM models, which serve as the input for the online application. As they are still contemporary BIM models from the AECO industry nowadays. The reusable building materials can still be stimulated with the resulting parameters from the BIM model data juxtaposition. Unfortunately, this will be less attractive and no circular-based non-graphical information will be added. But, the stimulation of reusable building materials is not the only aim of this research. Realizing a connection between the non-graphical BIM data and cloud-based software is a goal as well. These two are linked with each other, in which the stimulation of the reusable building materials is most important. But, if the connection still can be realized, projects, which do have the desired non-graphical information available in their BIM models, can be uploaded in a later stage of the online application. First, the connection must be realized, and afterward, the bridge to increase the desired non-graphical information based on the exchange requirements of the Re Use Index – ILS, can be developed. Therefore, the parameters, which are the result of the BIM model data juxtaposition, will be the input for the development of the online application in the next Section 6. This, to be able to retrieve data from the BIM models and to be sure the connection can be realized. Once this connection is developed and the test results are positive, the bridge between the online application and the request for the desired exchange requirements of the Re Use Index – ILS will be created in Section 7.

5.5 | Final Output of the Non-Graphical Information Analyses

The final result of this section will be two files, which will serve as templates. This to retrieve the requested non-graphical information from the projects, that will be uploaded by the project owners. The first template file will be concentrated on the Solibri software, since this is very often used in project teams in the AECO industry. To enable the project owner to get the same ITO environment with the parameters from the result of the BIM model data juxtaposition, a final feature in the Solibri software is proposed to create this option. A Solibri Role can be developed, in which the complete SMC environment will be customized based on rule sets, classifications, and a desired ITO environment, as they are all pre-defined. By uploading a Solibri Role in the SMC, the complete environment of the SMC will be changed based on the pre-defined input of that Solibri Role. This provides the opportunity to create a Solibri environment, based on the needs of this research. The only thing that needs to happen, is that a project owner can download the pre-defined Solibri Role. This to upload that file in the SMC and therefore is able to download the ITO Report with the customized parameters from the BIM model data juxtaposition. An example of this pre-defined SMC environment is depicted in Figure 25. The classifications are provided by the supervising company BASED BIM Management & Consultancy. The classifications, as the name implies, classify the objects in the BIM model into different classifications. These classifications as well as the ruleset, uses the structure of the basis-ILS, in which the universal naming of objects, classification, and entities are used. The ITO file is depicted as well. This ITO file will customize the ITO environment, based on the parameters from the BIM model data juxtaposition, once it is uploaded in the SMC.

The second file that will serve as a template, is the Project Report Template (Appendix VI). This file is similar to the ITO file, only this template is not a .ito file but an Excel Spreadsheet (.xlsx file). If a project owner does not use the SMC checker as a software tool to review his BIM model, he or she is still able to add the value of the objects from the BIM model into a template. The Project Report Template is the same customized environment, in which the parameters from the BIM model data juxtaposition are pre-defined, to ensure only these values will be completed, and no other not relevant parameters are added.

Both templates will be stored in the internal database of the online application, as it was previously discussed in the sub-chapter (4.4.3.1) and is depicted in Figure 16 (Section 4, Conceptual Model). The non-graphical information is analyzed, which forms the basis for the Rapid Application Development process of this research. In the next Section 6 (System Design), the first three phases of the RAD will be executed, based on the conclusion and output of the two templates from this section.

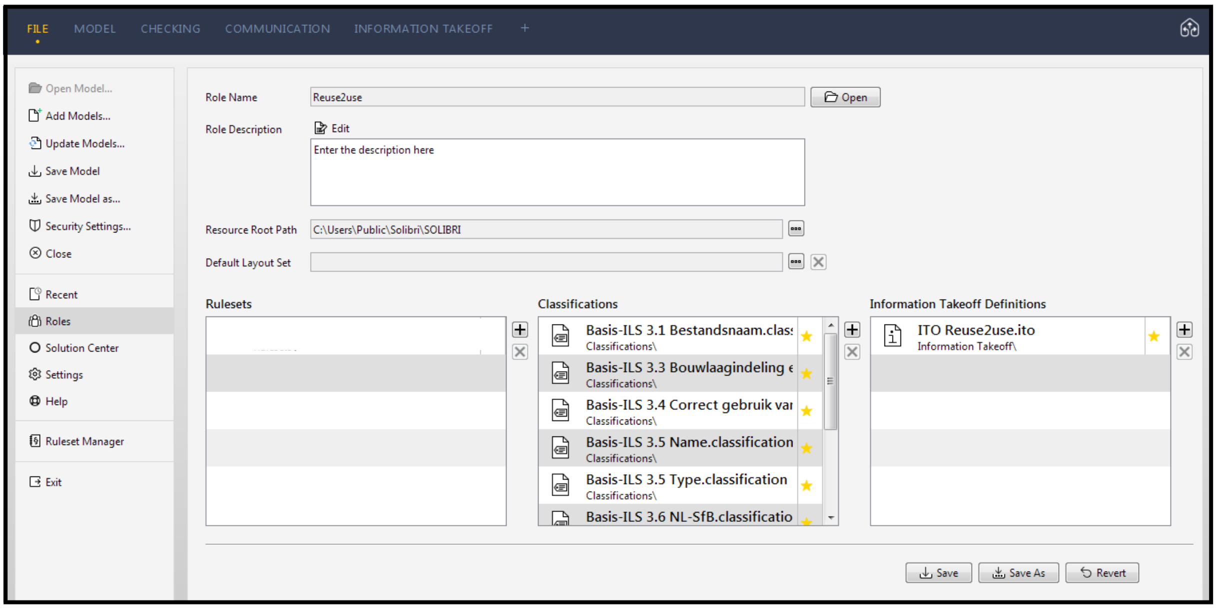
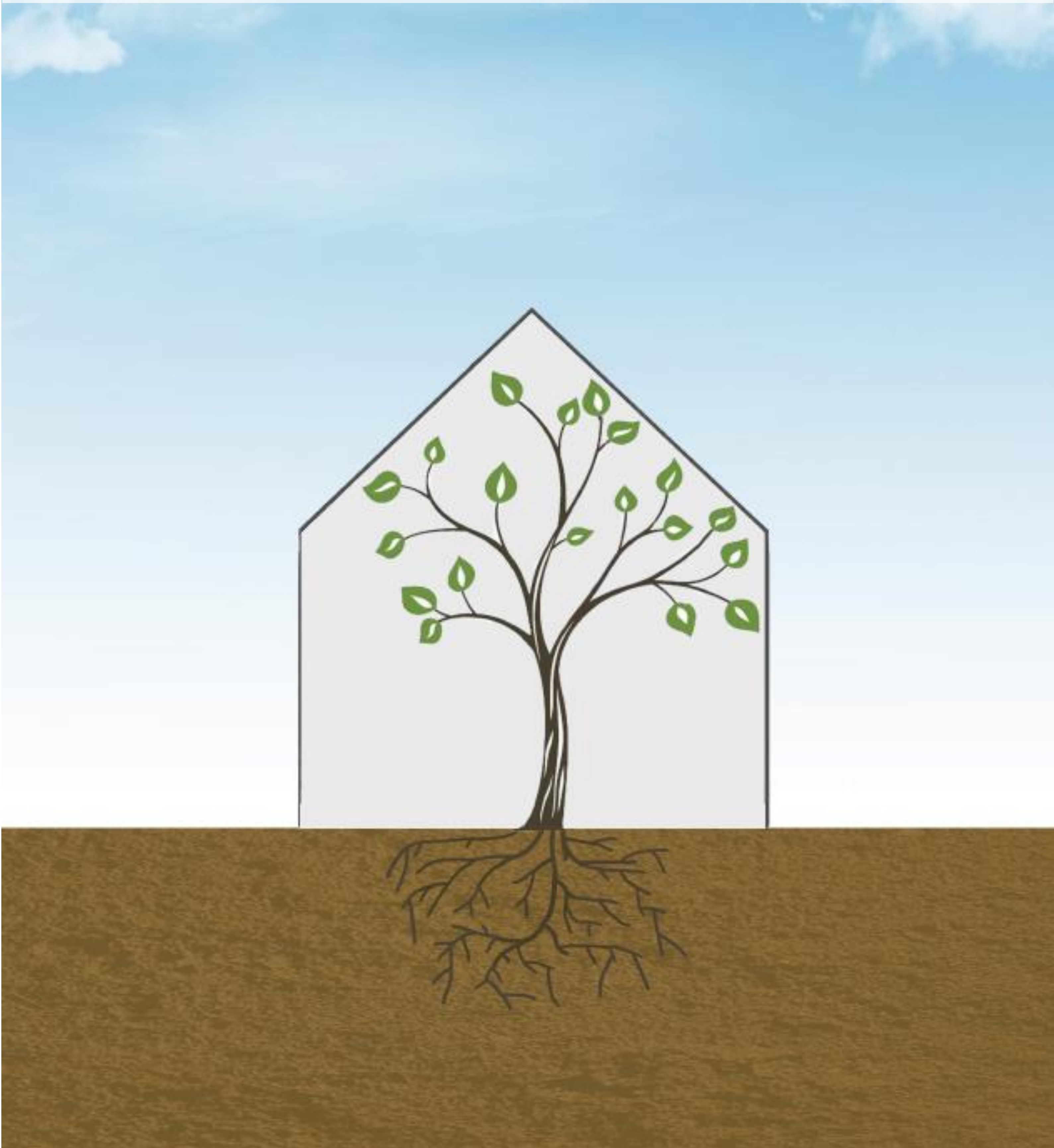


Figure 25: The layout of the Solibri Role for this research

Section 6

System Design



6 | System Design

In the methodology (Section 4) was defined that the connection between the non-graphical BIM data and future customers of the reusable building materials can be made with a cloud-based application. The system, which is the online application, will be designed based upon the Rapid Application Development method. The first phase is to set the requirements (sub-chapter 6.1) for the online application. The requirements will be from the perspective of the end-users of the application, which are the project owners and the webshop customers, since the needs of these end-users define the structure of the application. In Section 4 (sub-chapter 4.4.2), some requirements were already briefly described. The first sub-chapter will elaborate on these and add more requirements to the list. Once this phase is finished, it is time for the second phase: the design phase (sub-chapter 6.2). The design phase will define the actual design of the online application. The layout, style sheeting and display will be defined and visualized using a drafted scenario. In the drafted scenario, all the requirements from the first sub-chapter must be met. This forms the fundament for the most complex phase of the RAD: the development phase (sub-chapter 6.3). Based on the requirements, the drafted scenarios, the MVC pattern, and the challenge to connect the API with the web browser and MongoDB, this sub-chapter will describe the development of the back-end of the API. The development aims to meet all the requirements and at the end tries to match the drafted scenarios by style sheeting the API. This user-friendly style sheeting is named as the: Graphical User Interface. The first three phases of the RAD will be executed in this Section 6, the last phase (test phase) will be executed in the next Section 7.

6.1 | Application Requirements

When executing the first phase of the RAD, we take a look at what the online application aims for, which is connecting the project owner and its non-graphical BIM data with the webshop customer. Therefore, it is important to change the perspective for creating the design of the system, from the developer to the end-users. This will increase the value capturing of the online application, based on their needs. It is difficult to develop an online application and not considering the needs of the end-users. They have other needs and their requirements might differ from the one you expect as a developer. In this sub-chapter, stepwise, beginning at the start until the final step of the online application will be discussed. This will create a list of requirements that is adequate as fundamental for the design and development phase of the online application.

In the methodology (Section 4, sub-chapter 4.4.2), four requirements were already defined based on the conducted literature review (Section 2 & 3) and decisions made in the methodology. These are briefly listed again:

- The online application will be cloud-based, which means a connection must be made with a web server to be able to run in a web browser.
- The project owners need to be able to upload their project data and afterward be able to add the reusable building materials as products underneath this project. The data must be stored in MongoDB, which makes it able to retrieve and store this data.
- The output of the Non-Graphical Information Analyses (Section 5), consists of two templates which enable to retrieve requested data from the BIM models. These files need to be implemented in the design to provide these to the project owners.
- The webshop customers need to be able to view the products in a webshop and to be able to purchase them, the contact information of the project and its owner need to be applied to the products as well. This to create the possibility for an acquisition.

Now these requirements are set, it must be noted that the building materials from the projects will switch into products, from the moment they will be added to the online application. This explanation is needed to exclude misinterpretations in the next part of this research.

The first requirement that can be added to this list is the problem, as was stated in sub-chapter (5.3.3), that only an Excle .xlsx file can be exported as an ITO Report. The cloud-based software Node.js runs on Javascript and uses the JSON file structure as input. The .xlsx file cannot be uploaded if this will not be converted to an .json file. Therefore, a tool must be developed that enables converting the .xlsx file to an .json file:

- A converter must be developed that is able to convert .xlsx data into .json data.

As is already stated, two end-users of the online application need to be distinguished: the project owner and the webshop customer. Both end-users have a different purpose, using the online application. The project owners fulfil the supply side in which the webshop customers are the demand side. This evolves in the fact that two different environments must be created in which both end-users can fulfil their purpose without having to deal with the information needed to fulfil the needs of the other side of the application. This will make the online application user-friendly for both end-users, which results in a better end-user experience:

- Two environments must be developed in which both end-users can fulfil their purpose without having troubles with consuming information that belongs to the needs the other end-user.

Once the project owner has its own environment, the different project owners must be distinguished based on (1) if they have a BIM model available, and (2) if they have, which BMC software tool do they use to view the BIM model. This research concentrates on the project owners who use the SMC to view their BIM model. But to lower the threshold of the online application, other project owners, who do not have this software available, should also need to be able to use the online application. Therefore, different project owner paths must be created to provide this opportunity:

- Besides the Solibri project owner, other paths have to be developed to lower the threshold and make the online application more attractive and usable for other project owners.

After the desired path is chosen by the project owner, and he or she did convert the project report into an .json file, the project can be uploaded unto the online application. This step requires two requirements that are connected (front-end / back-end). The front-end requirement is the addition of contact information about the project, that provides insights into the location, mobile number, company/owner, etc. The back-end requirement is that this information must be stored, to be able to use it on different pages and finally to show this as contact information in the webshop. The storage of this information will be done using MongoDB. An extension of this step will be that the project report must be added to this project as well. While uploading and storing the project contact information, the project report must be stored underneath this project as well. This enables the project owner to add products underneath this project as well.

- While uploading the project's contact information, a tool must be created for uploading the project report template in the front-end, that stores this data in the back-end.

An overview page must be created that shows the uploaded projects. This makes it possible to switch between projects, if products want to be added on a later point of time. The project should always be accessible to the project owner.

- A web page must be designed, which shows every project that has been uploaded to the online application. This makes it possible to add products in a later phase to the project.

The next step is to add the potentially reusable building materials to the project as being the products for the webshop. An environment must be created in which the non-graphical information will be visualized to enable the project owner to add this data to the fields for the product's information.

- A table must be developed which visualizes the non-graphical information from the project report, that is retrieved from the BIM model. This will simplify the activity of adding the product's characteristics in the fields of the product's information.

An extra feature must be added here, which increases the attractiveness of the products in the webshop. Adding a picture to the product, that will be sent to the webshop, will increase the sales attractiveness, since a buyer always wants to see what the product looks like.

- Adding a picture to the product to increase the attractiveness of the products.

The product must be stored underneath the right project in MongoDB. An overview web page must not only be developed for the projects, but also for the products that are added. This, to provide the opportunity for the project owner to always have an insight into the products he added to a project. The webshop customer can see this web page as well. If he or she wants to buy a product from that project, the customer is able to see if there are more building materials from that project he or she wants to buy.

- Development of a product's overview web page for the project owners as well as the webshop customers.

The last step is to transfer the product data to the webshop. A user-friendly webshop must be developed, in which the products are visible as well as their added picture, counts, etc. If the webshop customer wants to get more information about the product, a web page must be developed as well, on which all the added characteristics of the products will be presented to give a clear overview of the product. A user-friendly interface is the key to keep the webshop customers coming back to the online application in the future.

- User-friendly interface for the webshop that shows the detailed characteristics of the products, the added picture, and creates therefore an attractive environment for the webshop customer to proceed to the acquisition of the products.

A final feature will be added to the design of the online application, is the reusable label. This reusable label will be an assessment tool that provides the opportunity for the project owner to assess his or her project and its building materials based on the reusable potentials. The assessment model is based on the Building Circularity Indicators (Section 2, sub-chapter 2.9) from the study of M. van Vliet (2018). The indicators are an easy tool to measure if a building material can be reused after it gets deconstructed from a building. The BCIs assess the shape, independence, type of connections, etc, which will give a score on how much of the value of the building material can be captured during is disassembly process. If this value is high, the potential to be reused is high as well. If this value is low, the building material will unfortunately not be attractive to add to the webshop to be reused.

- An extra feature will be added to the online application to enable the project owners to assess their building materials for their reuse potentials. This will be named as the reusable label in the development process of the online application.

To summarize the previously stated requirements that are necessary to be met during the development process of the online application, the requirements are illustrated in Figure 26. The requirements are classified over the different domains, as they are described in the Methodology (Section 4). The numbers of the requirements refer to the order in which they are executed during the development phase in sub-chapter (6.3). This sub-chapter will refer back to this figure, once a requirement is met. The MVC pattern is also included in Figure 26 of which certain requirements can only be met, during the implementation of this pattern. Some requirements are connected to two domains, which are placed on the dotted line. Requirements that are part of another requirement, being a sequence, are connected. Single requirements are listed within the domain in which they will be developed. Figure 26 will be fundamental for the application design phase in the next sub-chapter.

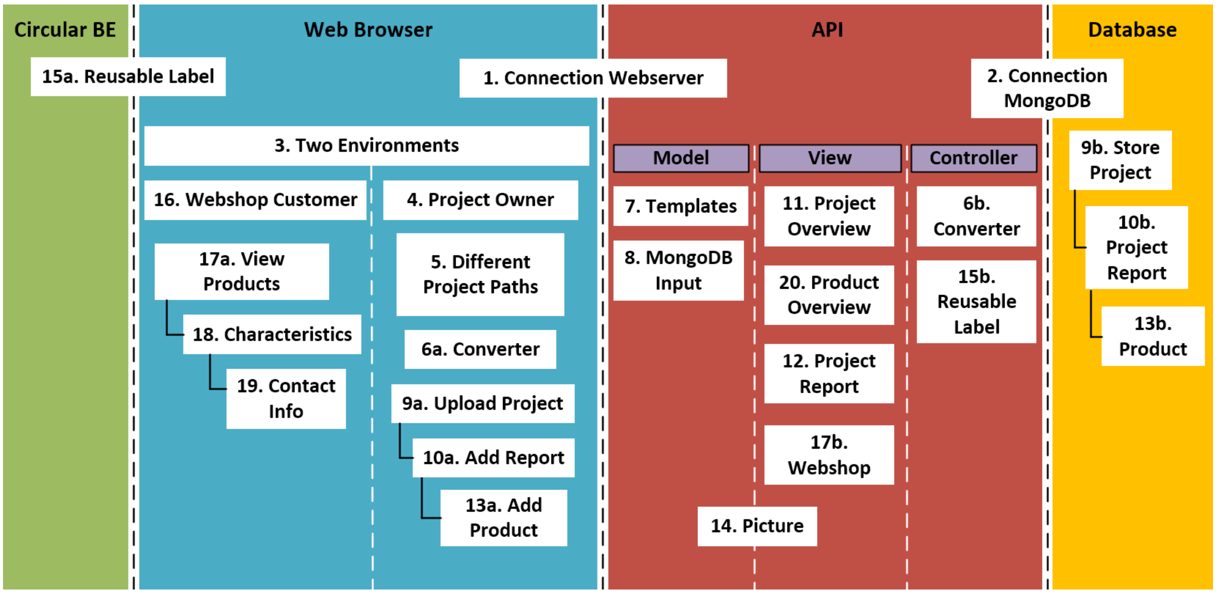


Figure 26: The requirements for the online application listed per domain

6.2 | Application Design

The second phase of the RAD is the design of the application. This phase aims to design drafted scenarios of the online application, which meets all the requirements that are defined in the previous phase. Now the requirements have been gathered and analysed, they can be verified and applied in the design. Creating the visualization is an important step that helps to remove possible flaws or mistakes in the design. The design will serve as the standard for the script will be developed in the next phase (sub-chapter 6.3).

To structure the application design, several important steps are made that are leading in this sub-chapter. The first step is the realization of the two environments. These two environments present the project owner on one hand and the webshop customer on the other hand. To enable a clear distinction, the project owner and the webshop customer environments are defined as follows:

Figure 27: Logo Project 2 Connect

The project owners will use the part of the online application, known as Project 2 Connect (Figure 27). This side defines the part of the application in which they can upload their projects, overview their projects, convert and add project reports, and finally add products. This environment aims to create a very simple way of uploading data and get easy access to the data that is retrieved from the project report. It therefore creates a Project 2 Connect. The project owner can easily see the building materials from their projects with their belonging characteristics.

Figure 28: Logo Re Use 2 Use

The second environment is the environment for the webshop customers. From this point in the research, this environment will be known as: Re Use 2 Use (Figure 28). The end-users of the online application, which fulfil the demand side, only need to be able to get access to the webshop. The distinction between the two environments will keep the online application attractive, in a way that the end-users only need to see the features and web pages that contain information for their needs.

The application design will be described based on an activity diagram and drafted scenarios. The activity diagram (Figure 36) is highlighted at the end of this sub-section. It shows the exact routing of web pages within the online application. The decisions for the development of these web pages and the choices the end-users have on these web pages are substantiated during the development of the application design in the next sub-chapters. The drafted scenarios are the visualization of the application design. With the activity diagram as a basis that outlines the structure and routing of the online application, will the scenarios visualize the required paths. These are both conducted to get an idea of the guidelines that must be considered during the development phase. An example of a drafted scenario is the start page of the online application depicted in Figure 29. The overall name for the online application is also: Re Use 2 Use (www.reuse2use.nl). In which Project 2 Connect will be a sidestep within this web domain. But both sides are important to create an online marketplace where the supply and demand side meet. The other part of the drafted scenarios are provided in Appendix VII. The data input and the redirecting of the pages are highlighted the activity diagram (Figure 36) as well.

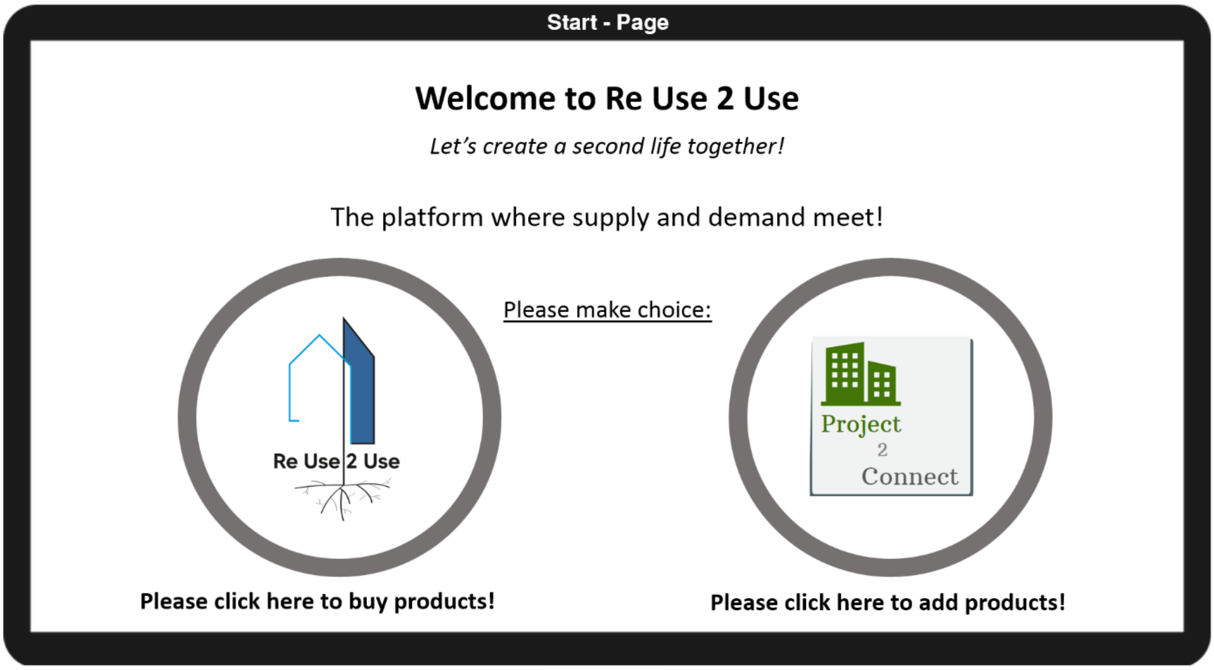


Figure 29: The start page of the online application (www.reuse2use.nl)

6.2.1 | Project 2 Connect

In the previous Figure 29, a decision can be made between the two environments. First, we will discuss the Project 2 Connect (project owner) environment. Considering the requirements for this environment, the first paths that need to be designed are the *add project* and *reusable label* paths. The *reusable label* path Figure 30 refers to the page where the assessment tool is located. This assessment tool will assess the building materials from the project based on the Building Circularity Indicators, which therefore indicates if the building materials have the potential to be reused or not.

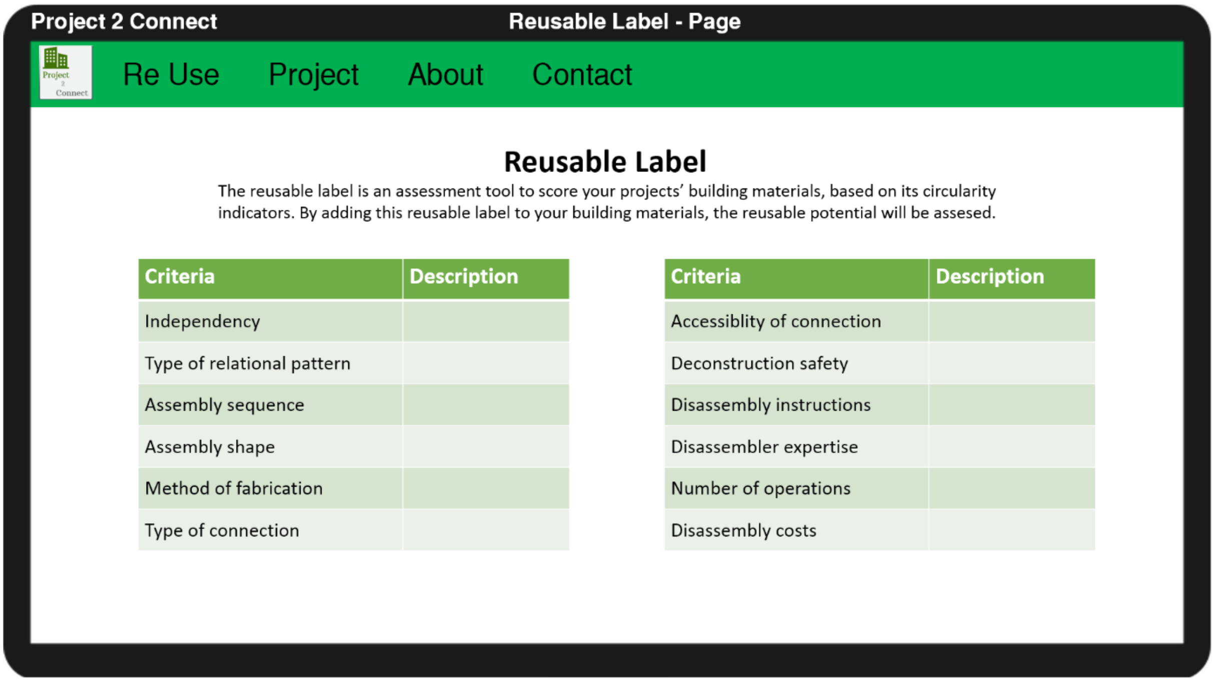


Figure 30: Reusable label page, for assessing the building materials for their reuse potentials

The *add project* path will refer the user to a page where three decisions can be made: *solibri user, private project user*, and *single product*. The *solibri user* path, as the name implies, is the path that is designed for the users who use Solibri for checking his or her BIM model. The *private project user* path is designed for the users that want to add a project but do not use Solibri for checking his BIM model. Other BMC software tools are not within the scope of this research and therefore do not have a particular path. Paths for other software tools will can be added as well. But for now, these project owners can use the *private* path or the *single product* path. This path refers to the web page to register the project immediately, before the need of converting the project report. If the *Solibri user* or *private project user* path is chosen, the webpage, where they will be referred to, explains stepwise how to upload and convert their project report. The *Solibri user* has several steps to follow, an example of the drafted scenario for this page is depicted in Figure 31. The convert tool must be developed in the online application which transforms the project report .xlsx file into a JSON file, which makes it possible for Javascript to read the file. The *private project user* path has the same layout, only this does not include the Solibri Role, but the Project Report Template. This Excel spreadsheet file, with the same layout as the ITO Report parameters, can be downloaded, adjusted with the non-graphical information from the project, and finally follow the same steps: upload, convert, and download as a JSON file.

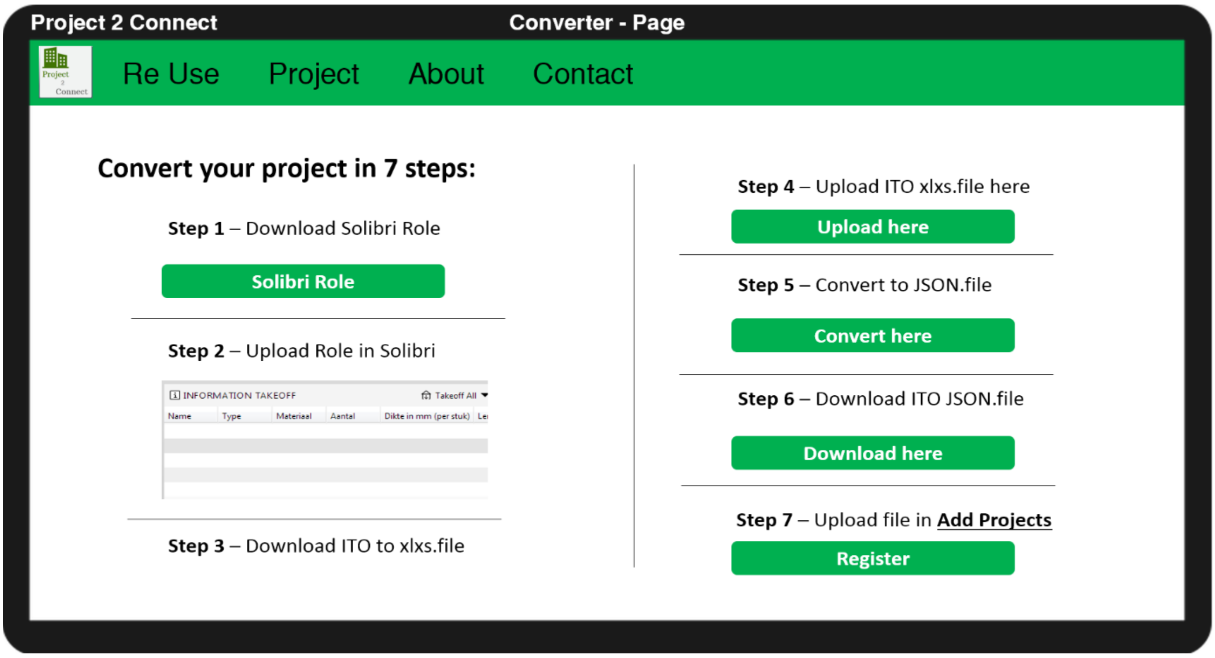


Figure 31: Solibri user path for upload / convert / download the .json Project Report

Once the ITO Report or the Project Report Template is transformed into the .json file, the next step will be to register the project in the online application. The *register* path is the page where the contact information of the project needs to be added to the online application. An example is shown in Figure 32, in which the company name, project address, etc. need to be applied to the online application. Afterward, this will be stored in the MongoDB database. Besides the contact information of the project or its owner, the converted report can be uploaded as well. The Project Report will be saved underneath the project in MongoDB. This, to retrieve the right data from the right project, once the project owners want to add products.

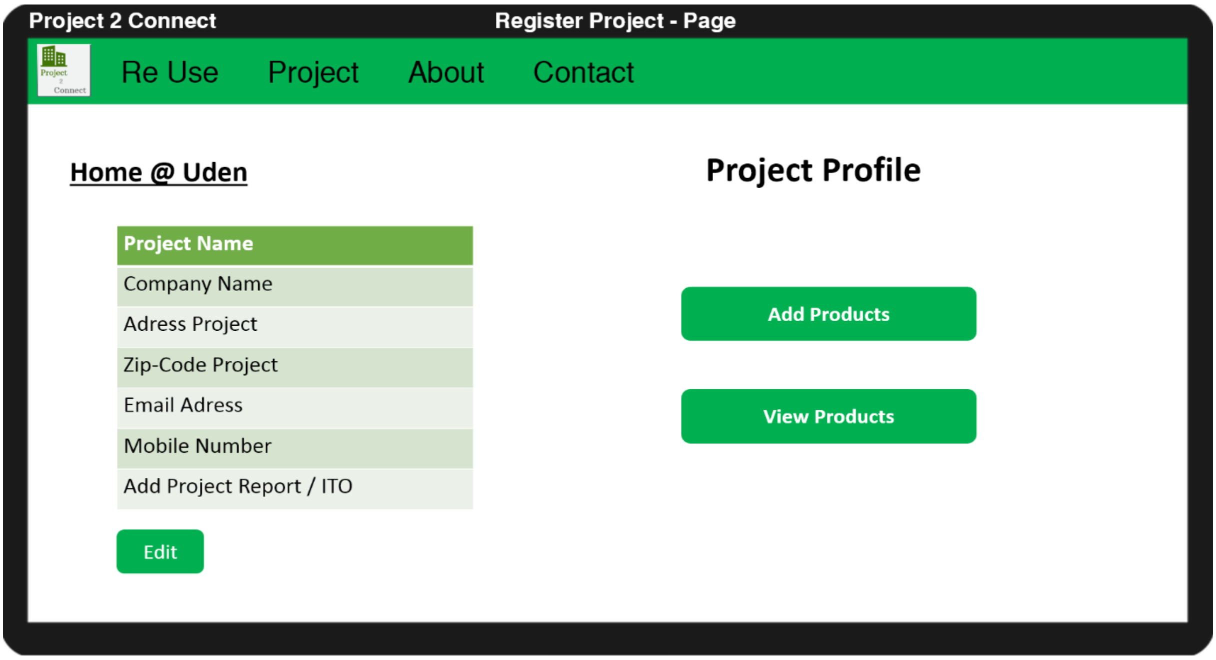


Figure 32: Register project and project profile path

The report will be connected to the database in which the data can be retrieved from once the products will be added to the project. After the project is registered, the user will be referred to a *project overview* path, in which every project that is added to the online application is located. The project profile of each project can be reviewed by clicking on the project. Once the user enters the project profile, two option are possible: the *view products* path, which refers to a page where all the products from that project that are added are listed; the *add product* path, which refers to the web page where the products can be added by the project owner.

The last page of the Project 2 Connect environment, which is the most important one, is the *add product* page (Figure 33). The *add product* page presents the non-graphical information, retrieved from, or the ITO report or the Project Report Template. The presented data are in line with the parameters from the BIM model data juxtaposition. These parameters and their values are presented in columns. With the non-graphical information on the left, the

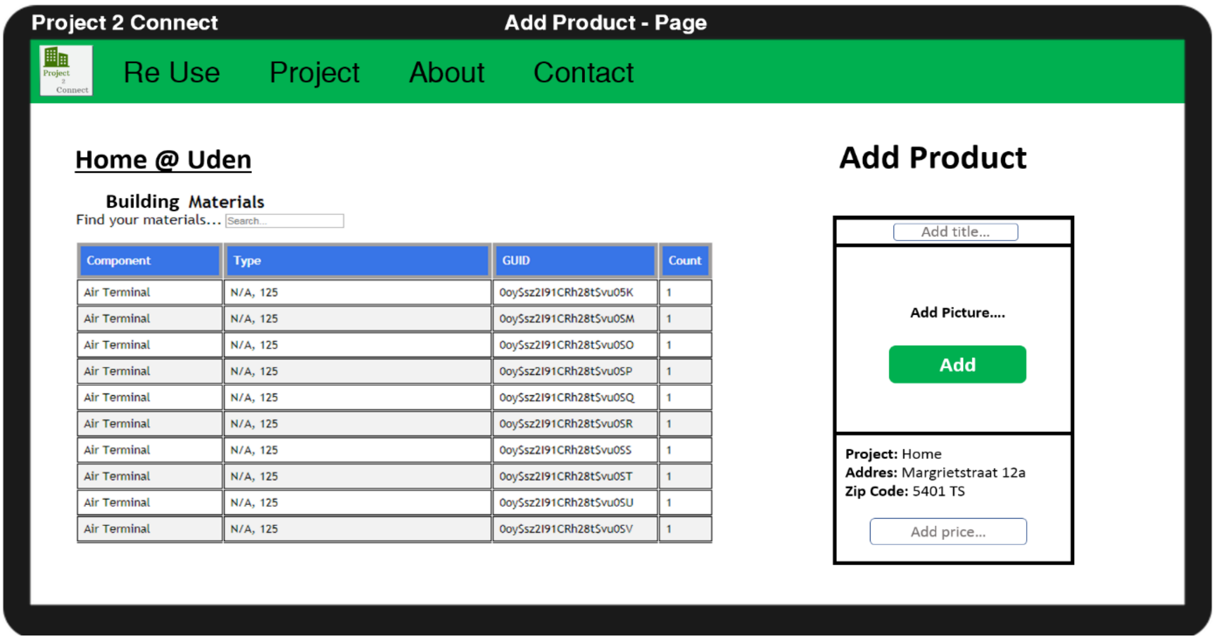


Figure 33: The add product path with the non-graphical information presented

data can automatically be adapted to the fields of the product, to connect the product to its characteristics. After the product is added, the product and the characteristics will be sent to the webshop. The *add product* page will be redirected to the *project overview* page.

6.2.2 | Re Use 2 Use

The Re Use 2 Use (webshop customer) environment requires fewer paths for the application design. After the decision is made for the Re Use 2 Use side of the online application, the *webshop* path will be presented. The webshop presents every single product that is added to the online application on the Project 2 Connect side. Within the webshop, the categories are listed in which the products can be classified. In the example of Figure 34, it was decided to search for the window frames. As can be noted, products in the shop are presented with their information, pictures of the product, and the price. The location of the project is highlighted in this format as well, because this is important information for the webshop customer. Since the scope of this research is to contribute to a cleaner planet, the location indicates the webshop customer if the product is nearby or not. If the product and its project is located on a far distance, it is not environmentally friendly to pick up a product when many kilometres by car or truck are required. Therefore, products on a short distance have the preference.

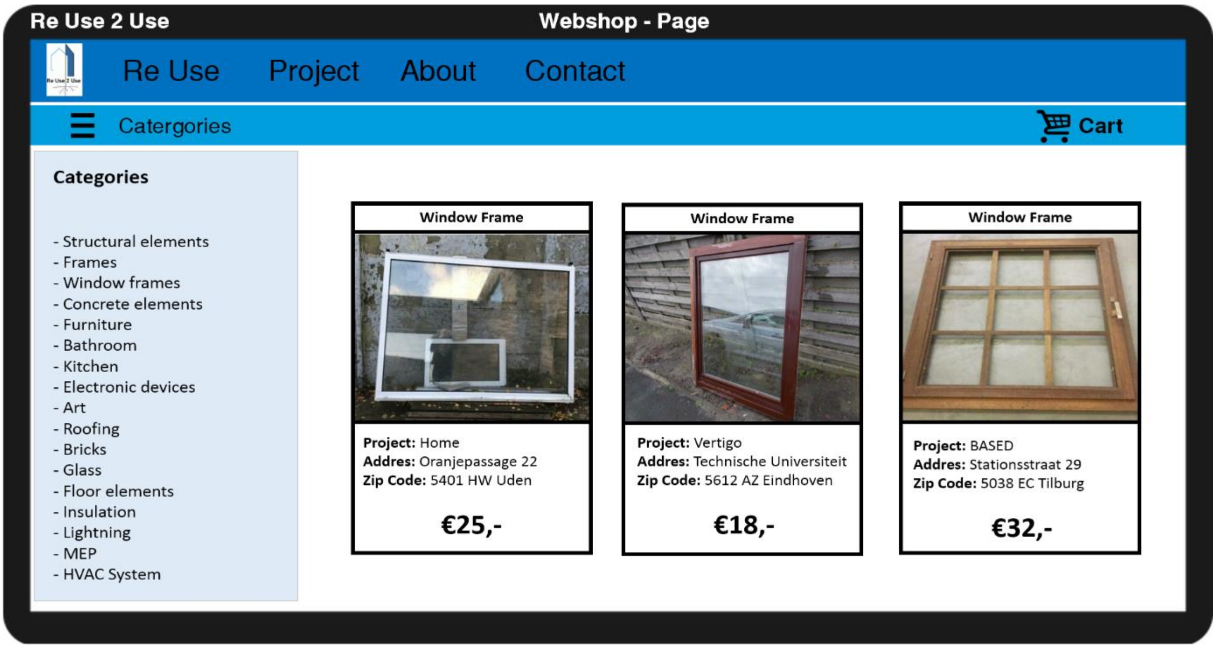


Figure 34: The webshop of Re Use 2 Use, with the example of the window frame

If the webshop customer wants to know more about the products that are presented in the webshop, this request can be fulfilled by clicking on the product. It will refer the customer to the *product* path. The *product* path (Figure 35) is the final path for the end-user in the webshop. In this path, the project information is highlighted, retrieved from MongoDB. The characteristics of the product, as they were added in the Project 2 Connect environment, are presented as well. Also retrieved from MongoDB. Finally, the contact information of the owner of the product is presented as well. This final step is necessary to make it able to create the actual transfer from the demand to the supply side. The webshop customer can contact the project owner and together they can make an agreement upon the price and number of products he or she wants to buy.

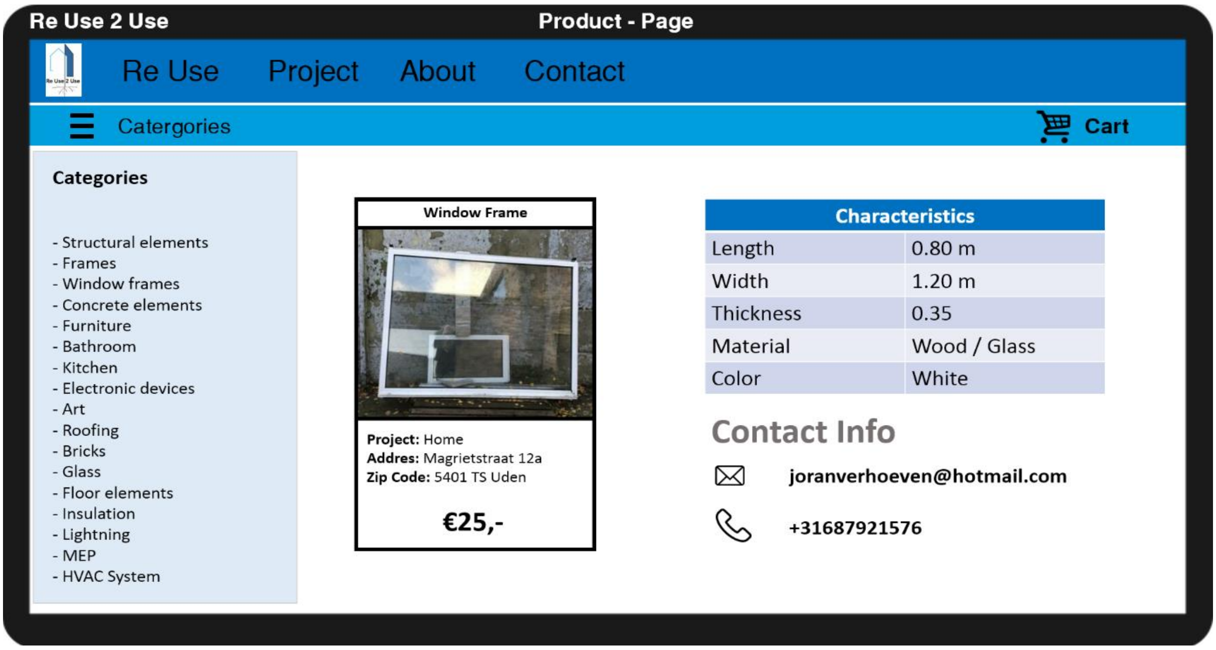


Figure 35: Final path in the webshop with all the characteristics of the product and its project information

6.2.3 | Navigation Bars

The final steps that must be adjusted to the online application are the navigation bars. Within each path, the end-user must always have the decision to return to the home pages. In the header of the webpage, at all times, the decision for the homepage of Re Use 2 Use (which refers to the *webshop*), the homepage of Project 2 Connect (which refers to the page to choose between the *add project* path or the *reusable label*), or the *about / contact* path (which gives background information and information about this research and the vision on of the online application) are located. Besides the header as a navigation bar, for each environment, an extra sidebar will be added, that refers to pages within that environment. These will be explained next.

6.2.3.1 | Project 2 Connect sidebar

The Project 2 Connect sidebar consists of three buttons. The first button is the *home* button, in which the end-user can refer back to the home page of the Project 2 Connect environment. The homepage is the page where the end-user can choose between *add project* and *reusable label* path. The second button is the *project* button, in which the end-users will be redirected to the *project overview* page. The last button is the *project search* button. This refers to the end-user to the *project search* page, in which the end-users can search between his own and other projects that are uploaded to the online application.

6.2.3.2 | Re Use 2 Use sidebar

The Re Use 2 Use sidebar consists of three main buttons. The first button is the *webshop* button, which refers the end-users at all times back to the home page of the webshop. The second button, similar to the *project search*, is the *product search*. On this page, the end-user can search for every product that is uploaded to the online application. The last button is a collection that refers to the *categories*. The *categories* are sub-divided by the IFC Entities of the objects from the BIM models. These IFC Entities are for example: window, wall, furniture, door, etc. When clicking on a category, the end-user will be redirected to the page of the webshop, which only shows the products with that IFC entity.

6.2.4 | Roadmap

The activity diagram is highlighting the final output of the application design sub-section. The drafted scenarios are discussed and visualized. The different web pages are explained, and which options the end-users have on each web page. This activity diagram is an overview of these webpages and the options the end-users have on each page. In Figure 36, the different environments can be distinguished, Project 2 Connect (green) and Re Use 2 Use (blue). The paths are defined with the arrows that point the direction to which web page the end-user will be redirected to. The diamond refers to a decision option, the forks to a point on which one incoming path will split up in multiple paths, and the joins the other way around, in which multiple incoming paths will evolve into one path. The data input for some web pages is depicted as well. The navigation bars are added as well, the header (purple) and the sidebars (orange). The home page at the start is coloured in red, since this page does not belong to one of the two environments. This design is fundamental to the application development phase (sub-chapter 6.3). Both the graphical part as well as the structure of the web pages is defined in this sub-section (6.2).

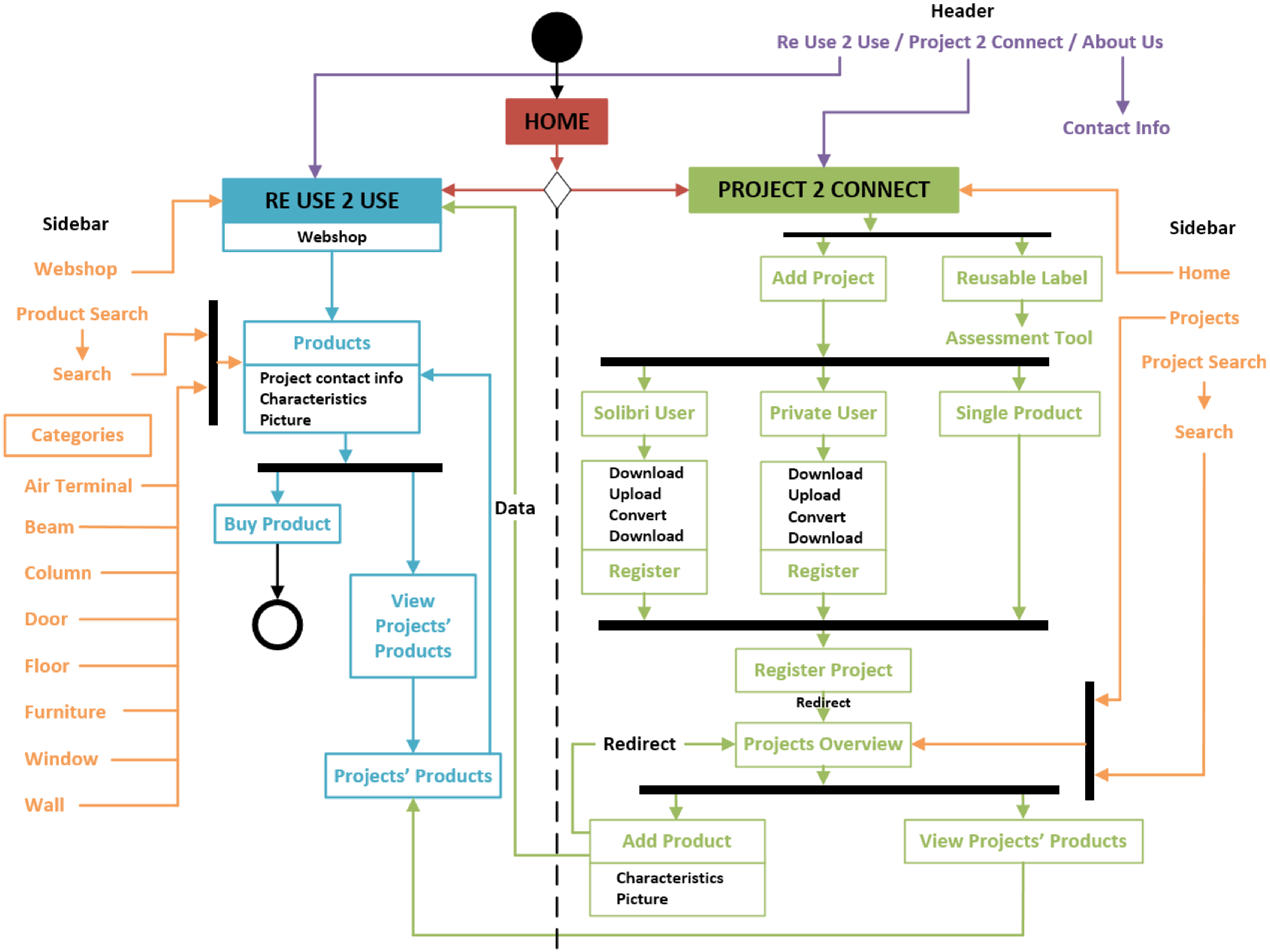


Figure 36: Activity diagram of the application design phase

6.3 | Application Development

In this sub-section, the development phase of the online application will be executed. The development will be based on the requirements that are set in the first phase (sub-chapter 6.1) of the RAD. The requirements are fundamental for the development of the online application. The second phase, as is a previously discussed, gives a clear visualization of what is looks like, especially on its front-end. The drafted scenarios define the tools and features that must be implemented on each web page. The activity diagram (Figure 36) shows the directions of each path within the online application. It shows how they are connected and therefore they must be developed in the back-end. In this phase, the application development will be described, starting from scratch up to launching the prototype. It then will be ready to be tested in the next Section 7. The stated requirements from sub-chapter (6.1) form the basis for the steps in the application development. Once they are met, this will be described referring to the number of the requirement as it is highlighted in Figure 25 (sub-chapter 6.1). The back-end will be discussed and depicted in this sub-chapter. Some parts of the back-end are highlighted in Appendix VIII as well.

6.3.1 | Web browser vs Server

Some parts of the online application are already briefly described and the connection between the web browser and the MVC pattern is already discussed in Section 4 (sub-chapter 4.4.3). This sub-chapter will go more in-depth this connection and explains the pattern of the back-end to clarify what is needed in the back, to achieve something in the front-end for the end-users. The first step is to describe how the web browser (Figure 37) is connected to the server. The user in the web browser is visiting a webpage, by entering a URL in the browser. This browser reaches out to a domain. The domain it searches for is an IP (Internet Protocol) address. The domain is the human-readable version, the IP address is the link to the server. The URL that is sent will lead to a server, the user in the web browser sends a request to that server, that belongs to that IP address. This the part where the application development steps in. The code on which the server runs with that particular IP address, must be developed to receive the request and send back a response. The code will be developed using Node.js. using Visual Studio Code (VSC) which is the code editor tool. VSC is a modern editor tool for developing web and cloud applications, which is invented by Microsoft for

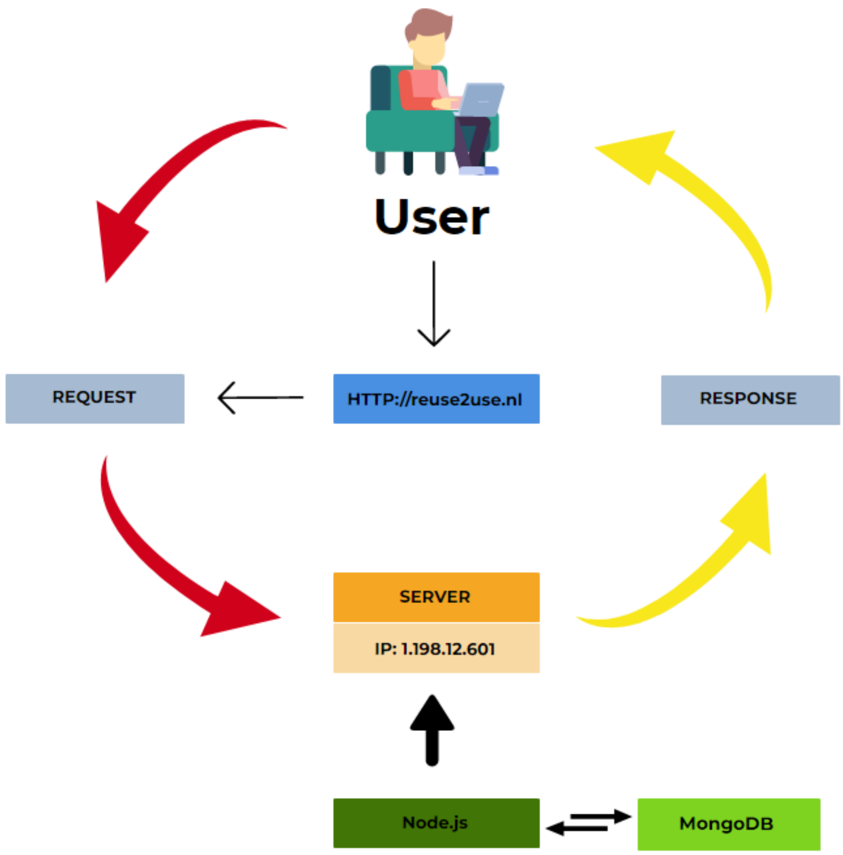


Figure 37: Example of connection user / server / Node.js / MongoDB

Windows. The code provides the user input validation as well as communicating with the database. If the request is handled, it sends back a response to the user in the web browser. This response can be HTML text, files, JSON data, etc. The HTTP defines the rules of the requests and responds. The Hypertext Transfer Protocol defines how a valid request looks like and how the data should be transferred from web browser to server, and from server to the web browser.

6.3.2 | Root File (Node.js server)

Starting with the development of the back-end of the online application, first, the root file must be created. This root file creates the connection between the Node.js code and the server in the cloud. A server needs to be created in Node.js, as well as core modules. The core modules help to create paths and features within Node.js that are not available in Node.js itself. This extends the possibilities with the Node.js code. With the *require* method (Figure 38), paths to other packages or modules can be created, after they are installed in Node.js. This method provides the opportunity to extend the possibilities with Node.js, features, and tools, which are originally not possible in Node.js. In Figure 38, examples of the core modules that are used for the development of the online application are depicted. As was briefly introduced in the Methodology (Section 4, sub-chapter 4.4.3.3), Express.js is used as a core module, which provides the EJS tool, to navigate more easy between files in the back-end. Those use the

```
var express = require('express');
var cookieParser = require('cookie-parser');
var bodyParser = require('body-parser');
var ejs = require('ejs');
var engine = require('ejs-mate');
var session = require('express-session');
var mongoose = require('mongoose');
var MongoStore = require('connect-mongo')(session);


app.use(express.static('public'));
app.engine('ejs', engine);
app.set('view engine', 'ejs');
app.use(cookieParser());

app.use(bodyParser.urlencoded({extended: true}));
app.use(bodyParser.json());
```

```
"author": "Joran Verhoeven",
  "dependencies": {
  "async": "^2.6.3",
  "bcrypt-nodejs": "0.0.3",
  "body-parser": "^1.19.0",
  "connect-flash": "^0.1.1",
  "connect-mongo": "^1.3.2",
  "cookie-parser": "^1.4.5",
  "ejs": "^2.7.4",
  "ejs-mate": "^2.3.0",
  "express": "^4.17.1",
  "express-session": "^1.17.1",
  "express-validator": "^3.2.1",
  "formidable": "^1.2.2",
  "moment": "^2.24.0",
  "mongoose": "^4.13.20",
  "nodemailer": "^3.1.8",
  "nodemailer-smtp-transport": "^2.7.4",
  "passport": "^0.3.2",
  "passport-facebook": "^2.1.1",
  "passport-local": "^1.0.0",
  "scripts-dev": "^1.1.0",
  "start": "^5.1.0",
  "underscore": "^1.10.2"
```

Figure 38: First input of the roof fil, app.js (core modules & packages)

same codes, such as headers, navigation bars, and footers. The EJS package is a template engine. Another core module, for example, is the mongoose and the MongoStore, which will result in the connection between the model in the MVC pattern and the external database of MongoDB. Other modules and their packages that are installed within the Node.js code for this online application are depicted on the right side of the figure. These are small features that resolve errors by creating a broader connection to the web browser. On the bottom left are the core modules and packages that are connected to the Node.js server. First, to be able to use them, the packages are installed and stored in Node.js. Next, the software input that is required from the modules is requested and finally, the modules are connected within Node.js.

The root file that connects the Node.js code with the server is named: *app.js*. This is the file that needs to run every time when the online application wants to connect with the server in the cloud. Using the *app.listen* method, makes it able to run the server on *PORT 3000* (URL = localhost:3000). A callback is added to the function to receive a message if the server runs successfully. In Figure 40: the green code is the request to be able to run the server; the yellow code shows the process of starting the server; the red code is the callback, which results in a successful action by sending the callback ‘our app is running on port 3000’, which means the Node.js server is connected to the web browser. This fulfils the first **requirement (1)**, as is depicted in Figure 26 (sub-chapter 6.1).

```
var app = express();
const PORT = process.env.PORT || 3000;

app.listen(PORT, () => {
    console.log(`Our app is running on port ${ PORT }`);
});

C:\Users\Joran\Desktop\Graduation Project\Graduation Project\Re_Use_2_Use>nodemon app.js
[nodemon] 2.0.3
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Our app is running on port 3000
```

Figure 39: Connection Node.js server with web server

The core modules are installed, and several are already connected to the Node.js server. The Node.js server is connected to the web browser, which provides a connection with the cloud. The next step is to connect an important module to the Node.js server, which is MongoDB. The Node.js server is the connection between the web browser and the external database (MongoDB). MongoDB is installed on the computer and the module is installed in Node.js. The database is required because data needs to be fetched and retrieved after the users upload their data. If the user refreshes a webpage, and no database is connected, all data gets lost. This can be resolved with the *app.use(session)* method (Figure 40). The first step is to connect the *mongoose* module, by connecting localhost (web browser) to the database (mongodb/reuse2use). This provides the opportunity for the web browser to store the data in the right database. The next step is to enable storing the data. This is done with the *app.use(session)* and the *MongoDB* module, based on the *mongoose* connection. This is an important step when the project owners want to upload their projects. Once this is stored,

```
mongoose.connect(process.env.MONGODB_URI || 'mongodb://localhost/reuse2use', {
    useNewUrlParser: true,
    useUnifiedTopology: true
});


app.use(session({
    secret: 'Thisismytestkey',
    resave: false,
    saveUninitialized: false,
    store: new MongoStore({mongooseConnection: mongoose.connection})
}));
```

Figure 40: Connection Node.js server with MongoDB

the user can add products to the project, which on their term can be stored as well. Finally, this data can be sent to the webshop to present the products and the project's contact information to the webshop customer. This connection fulfils the second **requirement (2)** that is determined in the application requirement phase, creating a connection between the API and MongoDB.

6.3.3 | Root Folder Structure

The Node.js server is now ready to be used. It already provides the connection between the web browser, the API, and MongoDB. Now it is time to classify the API, based on the MVC pattern that is defined in the Methodology (Section 4, sub-chapter 4.4.3.3)**.** The MVC pattern can be applied to the folder structure within the root folder, where the Node.js server is stored. The folders: models, views, and controllers are added, as well as the public folder, which contains uploads, files, images, CSS, and (java)scripts. A folder that was already added is the node_modules folder, which contains the packages and methods to run Node.js. The last two files that are in the root folder, as well as the *app.js* file, are the: *package.json* and *package-lock.json*.
The *package.json* file is depicted in Figure 38 on the right side, which shows the installed packages in Node.js and the version of the packages. The *package-lock.json* file is automatically generated once a module is installed in Node.js. P*ackage.json* defines the packages, the *package-lock.json* file describes the exact data tree that was generated, subsequent installs, updates, etc. (NPM Documentation, 2019). The tree structure from the root folder in which the complete online application will be built, is highlighted in Figure 41. This structure is the start for the development of the online application, based on the requirements, application design, and MVC pattern.

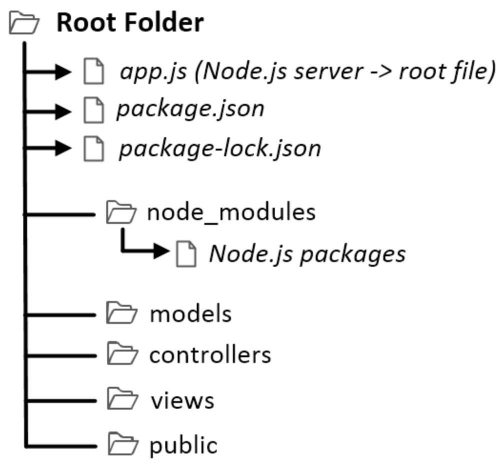


Figure 41: Tree structure of first plot (start MVC pattern)

6.3.4 | Navigation Bars

For the navigation bars, five bars can be distinguished: the *header*, *sidebar, sidebar reuse2use*, *sidebar project 2 connect, navbar.* The navigation bars run in the back-end, and as the name implies, they navigate through the API to respond and send data. The different navigation bars will be described one by one. The express.js packages are used to create this connection (Figure 42). This includes codes that refer to the navigation page in which the code is written. The views will not be an HTML file, but an EJS file, to use this package.

```
<% include ../navigation/sidebar-ru2u %>

<div id="page-content-wrapper">

<% include ../navigation/navbar %>

<div class="container-fluid main_body">
```

Figure 42: Express.js packages / navigation tools

Header

The header, as is defined by the Activity Diagram (Figure 36), has three options. In order to meet **requirement (3)**, two environments will be developed: Project 2 Connect (project owner) and Re Use 2 Use (webshop customer). In the header, the user can switch between the two environments. The header also has the *about us* button, which gives background information of the online application and contact information. The header will always be the same, no matter which environment is displayed.

Sidebars

Three sidebars can be distinguished. Because there are two environments, each environment gets its sidebar: *sidebar Re Use 2 Use* and *sidebar Project 2 Connect*. The Re Use 2 Use environment will be a blue environment, which has the options (Activity Diagram, Figure 36) to choose the buttons: *webshop, project search*, and *search* between different *categories* of the products. This sidebar needs to be different for the Project 2 Connect environment, since the buttons from Re Use 2 Use do not match the input of this environment. Therefore, this sidebar is coloured in green, to distinguish the environment. the sidebar has the buttons: *home, projects, and project search*. The third sidebar is the main sidebar, and this sidebar is used when the user did not choose for an environment yet. These pages belong to the main pages: *home* and *about us* page. The navigation page of the *sidebar Re Use 2 Use* is depicted in Figure 43.

```
<div id="sidebar-wrapper-ru2u">
    <div class="sidebar-brand">
        <a href="/home">Re Use 2 Use</a>
    </div>

    <ul class="sidebar-nav">
       <img src="/images/reuse2use.png" class="img-responsive img-rounded profile_img" alt="">
        <div class="profile_info">
            <span>Welcome</span>
            <h2>Joran Verhoeven</h2>
        </div>

        <li><a href="/reuse2use/webshop"><i class="fa fa-shopping-cart"></i>Webshop</a></li>
        <li><a href="/reuse2use/webshop"><i class="fa fa-search"></i>Product Search</a></li>
        <li><a href=""></i>Categories</a></li>
        <li><a href="/reuse2use/airterminal"><i class="fa fa-circle"></i>Air Terminal</a></li>
    </ul>
</div>
```

Figure 43: Example of the sidebar for Re Use 2 Use environment

Navbar

The navbar is the last navigation bar that is added, and this navigation bar creates a responsive menu. This means that the web pages are responsive to the width and height of the screen on which they are displayed. If the online application is opened on a web browser on a mobile phone, the page shrinks in the way that the user has the same features to choose from, only on a display with the same width and height of its mobile phone, laptop, computer, etc.

6.3.5 | MVC Pattern

The first codes of the API are developed as well as the root file which connected to the server. The MVC pattern is already added to the folder structure within the root folder. In this sub-chapter, the environment will be prepared to use the MVC pattern in the API. The first step is to add the *routes* to the *app.js* root file. The *app.js* file receives the request from the web browser, the *routes*, as is previously mentioned in the Methodology (Section 4, sub-chapter 4.4.3.3), send the controller in the right direction, Figure 44. Once the *routes* create the right direction, the controller can handle requests and decides what to do with the requests.

```
require('./controllers/project2connect')(app)
require('./controllers/reuse2use')(app)
```

Figure 44: Connection routes with controllers

The requested path will be fetched, and the controller is connected to that path, decides whether extra steps need to be followed, or the response can be rendered by the view and sent back to the web browser. The *controller* consists of three paths: *project 2 connect, re use 2 use, and users,* in which the *users* path controls the main pages such as the *home page* and the *about us* page, since this user is not yet defined. The model consists of one path: *project*. Which only needs to store projects and their belonging data. The non-graphical information can be stored and retrieved from MongoDB. The last component of the MVC pattern is the view. The view consists of three folders and two direct paths. The direct paths are already discussed (*home page and about us*). The three folders are: *navigation, project 2 connect, and re use 2 use.* If data is rendered by the controller, the view will send the presentation of that particular web page back to the controller, which on his turn sends it back to the web browser, using the Node.js server (*app.js*). In Figure 45, the tree structure of the MVC pattern folders is highlighted as they are known at this moment. This is the basis for completing the rest of the script based on the requirements, drafted scenarios, and the activity diagram.

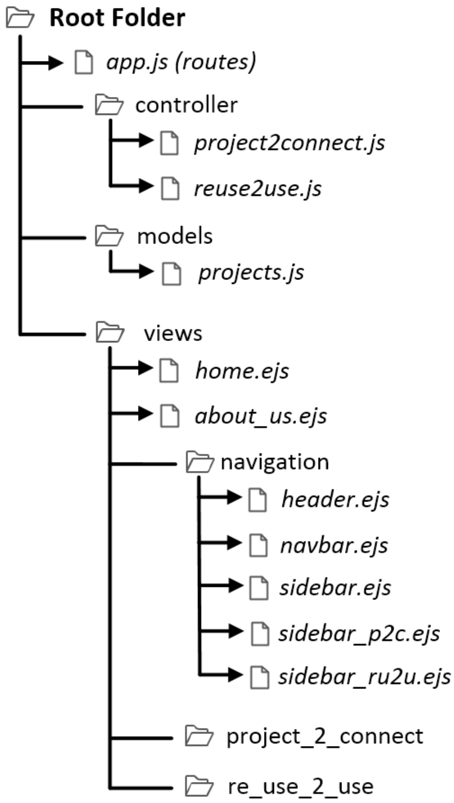


Figure 45: Tree structure second plot (input MVC pattern

6.3.6 | Project 2 Connect

If the user of the online application is the project owner, he/she chooses the Project 2 Connect environment **requirement (4)** and enters the homepage of this environment. Elaborating on the activity diagram (Figure 36), the project owner needs to get two decisions: *add project* and the *reusable label.* The first part of this sub-chapter concentrates on uploading the project by the owner, the second part will concentrate on the *reusable label.* Once the project owner

```
app.get('/project2connect/home', (req, res) => {
     Project.find({}, (err, result) => {
      res.render('project2connect/home', {title: 'Hompage || Project 2
       Connect, user: req.user, data: result});
     });
 });
```

Figure 46: App.get function, connection controllers with views

enters the homepage, the *routes* in *app.js* direct the controller to the path of *project 2 connect*. In this path, the following function (Figure 46) is required to create this path. What can be noted in the function is that the function is an *app.get* function. This means that the function ‘GET’ data, so retrieving data, sends a request for data. In a later stage, the *app.post* function is used as well. This function is able to ‘POST’, story and/or updating data. Ordering a product, will be able to use the data it retrieves and has the intention to execute activities with the data, rather than just retrieve data and not undertake activities.

A path is developed for every single webpage that is used in the online application, as well as the webpages for the Re Use 2 Use environment and the User pages. If a path is not programmed in the controller, the controller sends back an error and the request cannot be fulfilled. Figure 46 shows the script of the path of the homepage of the Project 2 Connect. Figure 47 shows an example of what this path looks like in the view. The homepage of Project 2 Connect provides two buttons where the users can choose from. The add project button and the reusable label button. These buttons refer to the next pages with the *a href* method. A few web pages only contain links to other pages, but some web pages contain data requests and responses. This example only contains links and does not need to retrieve or save data.

```
<div class="col-md-4" id="colDesc">
       <div class="info-icon"><i class="fa fa-folder-open"></i></div>
       <h3 class="text-center">My Projects</h3>
       <p class="text-center">Create My Project file. You can add your materials
       to your projects </p>

       <a href="/project2connect/home-project" class="btn btn-block btn btn-
       success link_btn">Add Project</a>
 </div>

<div class="col-md-4" id="colDesc">
        <div class="info-icon"><i class="fa fa-star"></i></div>
        <h3 class="text-center">Reusable Label</h3>
        <p class="text-center">Asses your products </br> based on </br> Building
        Circularity Indicators</p>

        <a href="/project2connect/reuseablelabel" class="btn btn-block btn btn-
        success link_btn">Check</a>
</div>
```

Figure 47: Example home-page Project 2 Connect environment

Converter

The *home-project* web page will meet **requirement (5)** by providing different paths where the project owner can choose from. The project owner has three options: *solibri user, private user,* and *single product*, as was already mentioned in the previous sub-chapter 6.2. The solibri user is the project owner who uses Solibri as a model view checker and can upload the Solibri Role and export the ITO report, as a result of the BIM model data juxtaposition of Section 5. The private user is the project owner who has a lot of data retrieved from a BIM model, but does not use Solibri as MVC. This user can use the project report template, which is also a result of Section 5**.** The single product path is for the project owner who just wants to add a single product, and therefore does not need to upload a complete project. Other paths might be added in the future for other software tools than Solibri. The webpage of

```
<div class="col-md-4" id="colDesc">
    <div class="info-icon"><i class="fa fa-download"></i></div>
    <h3 class="text-center">Step 1</h3>
    <p class="text-
    center">Upload the Re Use 2 Use role in Solibri and export the Information
    Takeoff Report</p>

    <div class="align-button">
        <a href="/files/Reuse2use_Template.zip" class="btn btn-success btn-lg align-
         button" type="button">Download Role</button></a>
    </div>
</div>
```

Figure 48: Example step 1 solibri_user path

the *solibri user* is divided over five steps. The first step is highlighted in Figure 48. The key for the *solibri users* and *private user* path is to convert the data from the ITO report or the Project Report from a .xlsx Excel spreadsheet file to a JSON file. We use JSON, which can be generated by a whole range of ways. One way to generate this, is to transform an .xlsx file from Solibri to JSON. Therefore, this interim step for both users is developed, to be able to work with the non-graphical BIM data when it is uploaded to the project. This should be done in the back and end and not be visible to an end-user. This is tried to be accomplished with the execution of the next steps: In the first step, the Solibri Role can be downloaded (1) from the online application. In the internal storage, the files folder in Node.js, a Reuse2use_template.zip file (**requirement 7**) is added. This zip file consists of the Solibri role. If the Solibri users download the zip file, the role can be opened (2) in Solibri, which gives the right environment to retrieve the non-graphical data from the BIM model as is defined by the BIM model data juxtaposition. Conducting an Information Take Off (3), an ITO report can be downloaded (4) from the SMC which contains the non-graphical BIM data based on the parameters from the analysis. This report can be uploaded (5) on and converted (6). This meets **requirement (6a)** in which a converter must be developed in Project 2 Connect environment. **Requirement (6b)** is depicted in Figure 49 and is a combination of several methods that together provide the converter tool. The steps are connected to the element ID. The figure of the Javascript creates the function to upload / convert / download the ITO report as a JSON file. The Javascript is the script behind the *solibri user* path in the view. Without this script, the webpage cannot proceed without errors or it just cannot get a result. The script retrieves step 2, after the file is uploaded, it uses it as an input for the *getElementById* method. The *var workbook* method makes it possible for Node.js to read the .xlsx spreadsheet and output the contents for a variety of formats. Step 3 converts the file from .xlsx to JSON with the same method. The script reads the Excel file, changes objects per row into an object per array. This is done to a JSON structure format. The *JSON.stingify* converts the Javascript object to a JSON string.

```
var selectedFile;

document
  .getElementById("upload-input")
  .addEventListener("change", function(event) {
    selectedFile = event.target.files[0];
  });

document
  .getElementById("uploadExcel")
  .addEventListener("click", function() {
    if (selectedFile) {
      console.log("hi");
      var fileReader = new FileReader();
      fileReader.onload = function(event) {
        var data = event.target.result;

        var workbook = XLSX.read(data, {
          type: "binary"
        });
        workbook.SheetNames.forEach(sheet => {
          let rowObject = XLSX.utils.sheet_to_row_object_array(
            workbook.Sheets[sheet]
          );
          let jsonObject = JSON.stringify(rowObject);
          document.getElementById("MyProject").innerHTML = jsonObject;
          console.log(jsonObject);

          var dlAnchorElem = document.getElementById('downloadAnchorElem');
          var dataStr = "data:text/json;charset=utf-
          8," + encodeURIComponent(jsonObject);
          dlAnchorElem.setAttribute("href", dataStr);
          dlAnchorElem.setAttribute("download", "MyProject.json");
        });
      };
      fileReader.readAsBinaryString(selectedFile);
    }
  });
```

Figure 49: Javascript of converter method

Finally, in step 4, the ITO report can be downloaded, and the ITO report is converted to a JSON file. With the *downloadAnchorElem* method, the function intends to download the element rather than display it in the web browser. That is needed to take the file to the next web page and the file will still be in possession of the project owner. Now, the file is ready to be added to the project.

But first, the project itself must be registered. These same steps are used for the *private user* path, it runs off the same script. The only difference is that the Solibri role cannot be downloaded in step 1, but it retrieves the Project Report Template (**requirement 7**) from the internal file storage. This has the same structure as the ITO Report, only this time it is an xlsx file, in which the data can manually be placed. The columns and rows are already provided, the project owner only needs to give a value to the fields. It may be noticed that this converter uses five steps instead of the seven steps from the drafted scenarios (Section 6, Figure 31). To make the webpage clearer, the first three steps for the *solibri user* could be combined..

Register Projects

This part of the online application requires several important steps. First, the webpage must be developed and connected to the controller, just like any other page. The next step is to connect the controller to the model, to store the data in the external database. The API is already connected to MongoDB, but the model needs to be programmed to store and fetch data. Once this is achieved, the final step is to create an upload button, in which the ITO report or the Project Report Template can be saved and stored underneath the project contact information. Step by step, this development will be explained.

The controller uses the same *app.get* method as in Figure 50, but now for the *register project* page. The *register-project.ejs* page in the view section requires several fields to retrieve the right data from the project owner. The project owner must fill in the fields to create a contact information list of the project. The fields are developed with the *POST* method, to store the data. The fields that are developed are the following ones:

- Project Name
- Country
- Adress
- Mobile
- City
- Email
- Project Report

Figure 49 clarifies the *POST* method as well as one field (project name) and the register button using the submit function.

```
<h4>Register Project</h4>

<form method="POST" action="/project2connect/my-project">

   <div class="form-group ">
      <label for="">Project Name</label>
      <span id="errorMsg1"></span>
      <input id="name" class="form-
      control" name="name" type="text" style="margin-bottom:20px;">
   </div>

   <div class="form-group">

      <input type="submit" id="register" value="Register" class="form-
      control btn btn-success" name="register" style="">
   </div>
</form>
```

Figure 50: Example register_project webpage, app.post function

This contact information is in a later stage connected to the product's profile, in which the webshop customer has all the information it needs to contact the project owner for a possible acquisition of the product. The country, as well as the address / city, are important to narrow down the range of search options for the products, since closer acquisitions reduce the emission of CO_2 since transport pollutions can be reduced.

The next step is to meet **requirement (8)** and connect the model of the API to MongoDB (Figure 51). The first function, as can be recognized from the function in *app.js*, *requires* the mongoose packages to create this connection. In this way, the root file and the model are connected and able to store and fetch data. The next part is to create the fields in MongoDB in the same order and value as is retrieved on the web page.

```
var mongoose = require('mongoose');

var projectSchema = mongoose.Schema({
    name: {type: String},
    address: {type: String},
    city: {type: String},
    country: {type: String},
    mobile: {type: String},
    email: {type: String},
    file: {type: String}
});

module.exports = mongoose.model('Project', projectSchema);
```

Figure 51: Input MongoDB and API model

Finally, the *mongoose.model* can be exported and stored in the external database of MongoDB.

One step that is important to not forget, to make this connection work, is the controller. The controller is the key in the API, which connects the model, view, and Node.js server. Therefore, the controller (Figure 52) needs to form the bridge between the model and the view. The first function *requires* the right path (models/project) and connects to the *module.exports* function. In the *my-project* webpage, the app *posts* the retrieved data in order of the fields that are defined in the model as well. A *new project* will be created each time, with the same value of fields. Finally, the project will be saved, using the model, and the project owner will be redirected to the *projects overview* page. With these steps, **requirements (9a & 9b)** are met. A new project is saved and stored in MongoDB. The project can now be used as data input for other web pages in the online application.

```
var Project = require('../models/project');
module.exports = (app) => {

    app.post('/project2connect/my-project', (req, res) => {

        var newProject = new Project();
        newProject.name = req.body.name;
        newProject.address = req.body.address;
        newProject.city = req.body.city;
        newProject.country = req.body.country;
        newProject.mobile = req.body.mobile;
        newProject.email = req.body.email;
        newProject.file = req.body.upload;

        newProject.save((err) => {
            if(err){
                console.log(err);
            }

            console.log(newProject);

            res.redirect('/project2connect/projects');
        })
    });
```

Figure 52: Connection controller with model

The last part of registering the project enables saving the ITO report and the Project report underneath the project that is almost ready to be registered. Not only does this tool need to upload the report, but it should also be able to search for the file on the computer of the end-users.

```html
<div class="form-group ">
  <label for="">Product Report</label><br>

  <button class="btn btn-lg" type="button" data-toggle="modal" data-
  target="#file">Add File</button>
  <span id="completed"></span>

  <div id="file" class="modal" tabindex="-1" role="dialog">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" id="close" class="close" data-
          dismiss="modal">&times;</button>
          <div class="modal-title">Upload Report</div>
        </div>

      <div class="modal-body">
        <div class="row">
          <div class="progress">
            <div class="progress-bar" role="progressbar"></div>
          </div>

          <button class="btn btn-lg upload-
          btn" type="button">Upload Report</button>
          <input type="file" class="form-control" name="upload" id="upload-
          input" style="display:none">
        </div>
      </div>
    </div>
  </div>
</div>
```

Figure 53: Add / upload / save, project report functions

Therefore, three steps must be achieved: *add file, upload report*, *save report*. In order to meet **requirement (10a)**, Figure 53 shows the steps that enables uploading the report within the view. Different than uploading the project report on the convert webpage, this report needs to be stored, and therefore a different approach must be used. In the figure is depicted how the three steps for adding / uploading / saving the report are programmed.

In the next Figure 54, the script that is required to execute these steps is highlighted. The project owner needs to know if the report is successfully uploaded before registering the project. Therefore, this script is programmed to not only upload the report but also shows a progress bar and sends after the report is successfully uploaded. It provides an upload within an upload (register project). Once the *add file* button is pressed, the end-user can search within the folders on its computer to search for the downloaded report.JSON file from the *convert* webpage. Once that file is picked, a progress bar shows the upload activity and finally sends the ‘file uploaded’ message when it is finished. The final step is to connect the script to the view using the controller. With the connection to the model, it stores the file underneath the project in MongoDB. The project owner can now press the register button; and the project and its report are stored in MongoDB, which fulfils **requirement (10b)**.

```javascript
$(document).ready(function(){

  $('#upload-input').on('change', function(){
     var uploadInput = $('#upload-input');

     if(uploadInput.val() != ''){
         var formData = new FormData();

         formData.append('upload', uploadInput[0].files[0]);

         $.ajax({
             url: '/upload',
             type: 'POST',
             data: formData,
             processData: false,
             contentType: false,
             success: function(data){
                 uploadInput.val('');
             },

             xhr: function(){
                 var xhr = new XMLHttpRequest();

                 xhr.upload.addEventListener('progress', function(e){
                     if(e.lengthComputable){
                         var uploadPercent = e.loaded / e.total;
                         uploadPercent = (uploadPercent * 100);

                         $('.progress-bar').text(uploadPercent+'%');
                         $('.progress-bar').width(uploadPercent+'%');

                         if(uploadPercent === 100){
                             $('.progress-bar').text('Done');
                     $('#completed').text('File Uploaded');
})
```

Figure 54: Javascript of uploading files in the API

Projects Overview

The end-users will now be redirected (Activity Diagram, Figure 36) to the *projects overview* page. This webpage is a collection of all the projects that are uploaded to the online application. This page only retrieves the project information that is stored in MongoDB. Is does only require the *app.get* method, since it sends a request and not a response. As is depicted in Figure 55, the function searches for the name of the project within the model. Every *name* that is retrieved gets its container and is connected to the ID of the project. This ID is created while storing the project. The ID makes the projects unique. This ID is the key to being able to use the same functions but create multiple outcomes within the same environment. The same holds for the *projects overview* page, in which the projects are defined by their name, get their container, and their unique link to go the project's profile. This unique environment is necessary if the application wants to add certain input to a particular project in a later stage.

```
<div class="bodyDiv">
     <h4 style="">Projects</h4>

     <% arrayAverage = function(arr) {
               return _.reduce(arr, (num1, num2) => {
                    return num1 + num2;
               }, 0) / (arr.length === 0 ? 1 : arr.length);
     } %>

     <% for(var i = 0; i < data.length; i++) {%>
               <% var total = arrayAverage(data[i].ratingNumber) %>

               <div class="col-xs-12 col-md-6 col-lg-4 ListDiv">
                   <a href="/project-profile/<%= data[i]._id %>">

                      <div class="ListItemInfo">
                          <h2 class=""><%= data[i].name %></h2>
```

Figure 55: Function for retrieving data from MongoDB to present it in the view

With the development of the *projects overview*, **requirement (11)** is achieved. This page gives a clear overview and makes it possible to search easily between the projects. The project owner can redirect him or herself to the *project's profile*. This webpage is unique for each project. The ID ensures that the link will never redirect to the wrong project, because of the 'passport' that is created. The project profile will retrieve the project information from MongoDB, in the same way it retrieves the data in the previous figure. Furthermore, within the *project's profile*, two choices can be made, as is depicted in the activity diagram (Figure 36): *add product* and *view project's products.* An example is also given in the scenario (Figure 32). Because no product is added yet to the project, we first need to develop the *add product* page. The *view project's products* will be defined in the Re Use 2 Use development part.

Add product

The *add product* page is a complex page, which eventually connects all the previous steps and is the bridge between the project owner and the webshop customer. This webpage must (1) retrieve the Project report from MongoDB from the particular project of which the products will be added to. (2a) It must present this data and (2b) be able to search within the retrieved non-graphical BIM data. (3) It must create fields to add the value of the characteristics to the product. (4) It must store this data underneath the project within MongoDB. And finally, (5) a picture must be added to the product as well to increase the acquisition possibilities of the product. This is a complex list of requirements and will be described step by step in this sub-chapter.

(1) Fetch Project Report from MongoDB

Behind the webpage *add-product* in the view, a script enables fetching the Project Report data from MongoDB. The script (Figure 56) retrieves the file (*document*) and does not have to respond to an action, only sends a request. Therefore, the *getJSON* method retrieves the .JSON file that is stored in MongoDB. Because the *add-product* webpage can only be visited after entering the *project profile* webpage, the webpage is already in the unique environment of the project, with its ID number. Therefore, the *getJSON* request can only retrieve the Project Report from that project. The next step is to fetch the data from each column of the Project Report. With the *myTable* function, the values of the columns are connected to the same value that are created in the templates. This is the powerful aspect of the template because now this data can be easily be retrieved from the report. If the project owners just uploaded their reports, the connection can be made, because Javascript cannot read those files when a connection cannot be made. The same parameters as were the result of the BIM Model Data Juxtaposition (Section 5) can be recognized. The script now retrieved the non-graphical BIM data and sends it back to the view, where the *add product* webpage must handle this request.

```
 $(document).ready(function () {

    $.getJSON("<%= data.file %>", function (data) {
        $.each(data, function (key, val) {

            $("#myTable").append(`<tr><td>${val.Name}</td><td>${val.Component}</t
d><td>${val.Count}</td><td>${val.IfcEntity}</td><td>${val.NlSfb}</td><td>${val.Ma
terial}</td><td>${val.Area}</td><td>${val.Height}</td><td>${val.Length}</td><td>$
{val.Width}</td><td>${val.Thickness}</td><td>${val.Volume}</td></tr>`);
            console.log(key, val);
        });
    });
});
```

Figure 56: Javascript of retrieving data from project report.json in the table

(2a) Present Retrieved Data

For presenting the data and afterward be able to add the values in the fields for adding the product, the *split-screen* method is used. This method splits the screens of the webpage, in which both sides can run on different scripts, style sheeting, responses, etc. On the right side of the webpage, the table with the Project Report data is presented. The view needs to request for this data, as is highlighted in Figure 57. It fetches the data, using the script, to be able to send the data back to the web browser. The values of the name of the columns are based upon the parameters retrieved from the BIM model analysis. The *id* of the table refers to the script, to fetch the data of that table. An example of the source of the script is depicted in Figure 58.

```
<script src="/javascript/table.js"></script>
```

Figure 57: Javascript source in Node.js

```
  <table class="table">
    <thead>
      <tr>
        <th>Name</th>
        <th>Component</th>
        <th>Count</th>
        <th>Ifc Entity</th>
        <th>Nl-Sfb</th>
        <th>Material</th>
        <th>Area m2</th>
        <th>Height</th>
        <th>Length</th>
        <th>Width</th>
        <th>Thickness</th>
        <th>Volume m3</th>
      </tr>
    </thead>
    <tbody id="myTable"></tbody>
  </table>
</div>
```

Figure 58: Data presentation of table in the view

(2b) Search option within table

The non-graphical BIM data is retrieved from the Project Report and the data is presented on the webpage, on the right side of the screen. As can be understood, this table can become enormous. Every single object from the BIM model is exported to the ITO or Project report, and therefore will be presented on the webpage. What can help to get easy access to the required product inside the table, is to add a search option to the table. This search option shows only the products that match the search hit. To develop this tool, an extra function is added to the script (Figure 59) of the *table*. The *jquery toggle* hides unmatching elements and shows matching elements of the *lowercase* values. In this way, the search hits that match the wanted products will be shown and the unmatching hits will be hidden. This creates an easy search tool for the end-user. With the presentation of the project report and the added search tool, **requirement (12)** is met, presenting the Project Report in the view.

```
$("#myInput").on("keyup", function () {

    var value = $(this).val().toLowerCase();
    $("#myTable tr").filter(function () {
        $(this).toggle($(this).text().toLowerCase().indexOf(value) > -1)
    });
});
```

Figure 59: Javascript of search function of the table

(3) Characteristics Fields

The right side of the *add product* webpage is defined, now it is time to focus on the left side, the actual addition of the products. As was depicted in the scenario (Figure 33), the table is one part and the other part is focussed on adding the characteristics to the products. Which (can in turn) can be sent as data input for the webshop. The values for the characteristics need to be added manually in the fields. This decision is made because the non-graphical BIM data can be very complex, or some fields have the same values, and this does not create an attractive webshop. By developing the same fields on the left side, based on the characteristics of the table on the right side, the value can be copied to the fields, while edit options remain to be available. The unnecessary non-graphical data can be filtered out, and the relevant data

```
<div class="form-group">
    <label for="">Project</label><br>
    <label><%= data.name %></label>
</div>

<form action="" method="post">
    <div class="form-group ">
        <label for="">Name</label>
        <input id="role" class="form-
        control" name="name" type="text" style="margin-bottom:20px;">
    </div>

    <div class="form-group">
        <input type="submit" id="new" value="Add Product" class="form-
        control btn btn-success" name="productInfo" style="">
     </div>
</form>
```

Figure 60: Presentation of characteristics field for adding product

can be applied. Even if some fields in the table do not have a value, the project owner can add the value him or herself. If the dimensions are measured afterward or the properties of the material status are too brief in the table, this can be added as well. An example of one of the fields for the characteristics of the webpage in the view is depicted in Figure 60. The other fields are added as well, similar to the values of the column names of the *table* on the right side. An extra field that is added is: *price*. In this field, the project owner can immediately set the price that he or she wants to get when selling the product to the webshop customer. The last step is to submit the product pressing the button: *add product*. With this step**, requirement (13a)** is met. To meet **requirement (13b)**, the values of the characteristics must be stored in MongoDB. This connection is not programmed yet.

(4) Store Product Underneath the Project

To store the characteristics of the product in MongoDB and save that data underneath the project to which it belongs, a new connection between the controller and the model must be programmed. In the model, an extension on the earlier created *projectschema* must be developed. This can be done using an array within the *projectschema*. The array *products* provide the opportunity to add extra product characteristics inside the *projectschema*. As is depicted in Figure 61, the list of parameters from the Project Report is added as well as three extra options. A product ID that is necessary to create an unique product for each project, the price as earlier mentioned to give an economic value to the product, and an image string is added as well, which will be defined in the next sub-section.

```
var projectSchema = mongoose.Schema({
    name: {type: String},
    address: {type: String},
    city: {type: String},
    country: {type: String},
    mobile: {type: String},
    email: {type: String},
    file: {type: String},
    products: [{
        productId: {type: String, default: ''},
        productName: {type: String, default: ''},
        productComponent: {type: String, default: ''},
        productCount: {type: String, default: ''},
        productIfcentity: {type: String, default: ''},
        productNlsfb: {type: String, default: ''},
        productMaterial: {type: String, default: ''},
        productArea: {type: String, default: ''},
        productHeight: {type: String, default: ''},
        productLength: {type: String, default: ''},
        productWidth: {type: String, default: ''},
        productThickness: {type: String, default: ''},
        productVolume: {type: String, default: ''},
        productPrice: {type: String, default: ''},
        productImage: {type: String, default: 'defaultPic.png'}
    }],
```

Figure 61: Connection products characteristics underneath the project is belongs to

Now the connection is made within the model, it is possible to store the characteristics from the product in MongoDB after it is submitted in the view. But the middleman cannot be forgotten, and therefore also the connection must be developed in the controller (Figure 62). Creating a new object is not necessary this time. Only the object *project* must be updated. And therefore, the function *Project.update* is used to keep the root object but add new data within the object. In this way, the product can be stored underneath the project. This provides the opportunity to connect the project information to the project's characteristics in the webshop.

The *Project.update function POSTS* the data to the model and therefore MongoDB. The function is a *callback* function because it is passed into another function because it uses the *update* method. By storing the product's characteristics into MongoDB, **requirement (13b)** is met as well. The last requirement that must be met, is the addition of a picture to the products.

```
app.post('/project2connect/add-product/:id', (req, res, next) => {
 async.parallel([
  function(callback){
   Project.update({'_id': req.params.id,'products.productId': {$ne: req.user._id}
   },
   {
   $push: {products: {productId: req.user._id, productName:req.body.name, product
Component:req.body.component, productCount:req.body.count, productIfcentity:req.b
ody.ifcentity, productNlsfb:req.body.nlsfb, productMaterial:req.body.material, pr
oductArea:req.body.area, productHeight:req.body.height, productLength:req.body.le
ngth, productWidth:req.body.width, productThickness:req.body.thickness, productVo
lume:req.body.volume, productPrice:req.body.price, productImage:req.body.upload}}
    }, (err, count) => {
         if(err){
            return next(err);
         }
         callback(err, count);
    });
```

Figure 62: Connection add product of the controller with the model

(5) Add a Picture to the Products

Adding a picture to the products flows through the same codes as adding the Project Report to the project object in MongoDB. The same function can be used. First, the end-user must press *add file*. The next step is to search for the image between his or her folders on the computer. Finally, the file can be uploaded. As can be noted on previous figures, is the *image upload* already added to this function. In the model, the *productImage* uses the .png format to save the image. In the controller, the *productImage* is an upload function, which is able to upload a file. This is different than the other parameters. **Requirement (14)** is met with programming this extra option within the *add product* webpage. After submitting the product, the characteristics will be stored in MongoDB and the end-users will be redirected to the *projects overview* web page.

Reusable Label

The last webpage that needs to be developed on the Project 2 Connect side is the *reusable label* web page. Once the project owner enters the Project 2 Connect environment, besides adding a project, he or she can also assess the building materials based on circularity indicators. The conducted circularity indicators, retrieved from the literature review (Section 2), give an indication for the project owner which of his products has the potential to be reused again, or maybe got too much damaged during the deconstruction process and therefore cannot be reused anymore. This circularity assessment tool is named as the *reusable label* scenario (Figure 30). On this page, a tool is developed which makes it easy to score the building materials. The webpage in the view is connected to a script: *reusablelabel.js*. This script (Figure 63) provides the opportunity

```
$(document).ready(function(){
    var clickedValue = 0;

    $('#1_star').hover(function(){
        $('#1_star').attr('src', '/images/star_on.png');
        $('#2_star').attr('src', '/images/star_off.png');
        $('#3_star').attr('src', '/images/star_off.png');
        $('#4_star').attr('src', '/images/star_off.png');
        $('#5_star').attr('src', '/images/star_off.png');

        $('#showTitle').html('Bad');
    });
```

Figure 63: Javascript of score table for reusable label

to give a star rating to the building material, based on the indicators. The circularity indicators are described in the webpage, and the rating is connected to this script. Based on a scale from 1 to 5 stars, in the end, a certain number of stars is given to the building material. At the bottom of the page, the ranges of the number of stars are defined. Based on the range that matches the number of stars that are counted by the user, he or she can decide if the building material has the potential to be reused. This meets the **requirements (15a)** and **(15b)**, since the reusable label is elaborated on the literature review and is defined by the script in the MVC pattern.

6.3.7 | Re Use 2 Use

The first side of the online application is developed. But there are still requirements left and the activity diagram (Figure 36) is also not completed. Therefore, these sections also need to be described to complete the application development. The complex part of the online application is finished after programming the Project 2 Connect side. The non-graphical BIM data is fetched, stored, edited, and adjusted. The project owner is finished with registering his project and adding the products he or she wants to sell to be reused. This side of the online application is focused on the customer of the webshop, and therefore has a more user-friendly approach based on the attractiveness of the webshop. The webshop customer (**requirement 16**) needs to get a clear overview of the products, hence the products must be presented with the necessary information such as the price and the picture. And the customer needs to be able to see the project contact information, the product's characteristics, and the complete list of products from one project.

Webshop

The most important aspect of the development of the webshop is how the products and their characteristics will be presented. The webshop, as is depicted in the scenario (Figure 34), is a collection of the products that are added to the project in the Project 2 Connect side. The data that is required is available, the most complex connections between the API, web browser, and MongoDB are already programmed. The webshop will be built upon these connections and the input of the added projects in MongoDB. The products need to be presented in a way that webshop customers know from other platforms. The products are placed next to each other and just some brief information of the products is presented. First, the name of the product must be presented on top. Underneath the name of the product is the category, to which the product belongs. This has a large size in the frame of the product. Below the image is the price, which also is an important aspect of the product. This is highlighted as well in a different colour. Finally, the city and country are presented to give an indication which reusable building materials can be bought close by, and which are available on a wider distance.

Now these development guidelines have been set, it is time to develop the *webshop* webpage. The required data from the guidelines must be fetched from MongoDB, where they may be available. Therefore, the back-end of the webshop must be developed on a function that can retrieve the available data. The *arrayAverage* function (Figure 64) returns the average of values that is provided. The arrays that have the value of the request of the functions are sent back to the *webshop*. The *reuse2use.js* file in the controller is also connected to *models/projects* of which it can fetch data from the view pages in the Re Use 2 Use environment. The *data[i]* array refers to the projects that are uploaded in MongoDB. Once this connection is made, the characteristics of the products inside this project can be fetched. Using the [i] array enable retrieving every single product from every single project. The required information is requested: productName, productComponent, productPrice, city, and country. With the *img src*, the image that is added to the product is retrieved as well. The last step is to connect and *a href* link to the product. Once the webshop customer wants to know more about the product, is can click on the product and he or she will be redirected to the *product's profile* webpage as is outlined in the activity diagram (Figure 36). Now these steps are programmed, and the products is presented in the webshop, **requirement (17a)** is met by creating a *webshop* in the view component as well as **requirement (17b)**, because the webshop customers can view the products.

```
<% arrayAverage = function(arr) {
   return _.reduce(arr, (num1, num2) => {
      return num1 + num2;
   }, 0) / (arr.length === 0 ? 1 : arr.length);
} %>

<% for(var i = 0; i < data.length; i++) {%>
   <% var total = arrayAverage(data[i].ratingNumber) %>

   <div class="col-xs-12 col-md-6 col-lg-4 productWebshopDiv">
      <a href="/reuse2use/product-info/<%= data[i]._id %>" style="text-
      decoration: none;">

        <div class="productWebshopItemInfo">
          <h1 class=""><%= data[i].products[i].productName %></h1>
          <h2 class=""><%= data[i].products[i].productComponent %></h2>

         <img src="/uploads/<%= data[i].products[i].productImage %>"
         class="productWebshopItemImg">
         <div class="productWebshopItemPrice">
            <h2>€<%= data[i].products[i].productPrice %></h2><br>
         </div>

         <h5 class="fa fa-map-
         marker "> <%= data[i].city %>, <%= data[i].country %></h5>
       </div>
      </a>
   </div>
<% } %>
```

Figure 64: Presentation of product frame for the webshop

Product's Profile

On this webpage, the complete object that is stored in MongoDB will be presented. This uses a similar approach in functions as the webshop. But this time the complete project contact information, as well as the product's characteristics, will be required. The product's profile aims to inform the webshop customers with all the details, information, and requirements of the product and the project which it belongs to. Therefore, the characteristics will be fetched from MongoDB to present the complete property set of the product. Also, the project contact information will be presented, which increases the chances for an acquisition between the webshop customer and the project owner. The project name, address, city, country, and mobile number and email of the project owner are presented, which makes is very easy to contact the owner of the product once the webshop customer is interested in purchasing the product. The right data will always be retrieved when entering the *product's profile* of the product that is chosen by the webshop customer because this webpage also is based on the ID of the product. Once the webshop customer clicks on a product, automatically the ID of the product will be retrieved and therefore only the characteristics and project information of this product will be fetched. The *product's profile* webpage, the **requirements (18)** and **(19)** are met. A final button will be added to this webpage that refers to the *project's products* overview webpage.

Project's Products Overview

This webpage is linked to the Re Use 2 Use environment as well as the Project 2 Connect environment, as is defined in the activity diagram (Figure 36). This webpage presents every product that is added to the project. This webpage is developed based on two aspects: the project owner has a clear overview of the products that are added to his or her project. The same holds true for the webshop customer: if he or she is interested in buying a product from the project, maybe there are more products available in this project which the customer can be interested in. The *project's products overview* page works with an *if-else* function. *If* data can be retrieved, the table will be filled with the values that are added to the characteristics of the product. The *else* function sends a response 'no products registered yet´ if no data is available in MongoDB and therefore cannot be retrieved. With the development of this webpage, the last **requirement (20)** is met.

6.3.8 | Cascading Style Sheets (CSS)

The last part of the application development, and also a very important aspect, is the CSS. The style sheeting of the webpages is highly recommended when designing an online application. In the previous sub-chapters, the back-end scripts, functions, and program languages are described and defined. As can be notated from the figures, the code is not a very attractive way of presenting the API to the end-users in the web browser. Therefore, the CSS (Figure 16, Methodology Section 4) renders the back-end codes retrieved from the controller and the view will send back a presentable response. The Graphical User Interface will be defined in this part of the application development. The .css files describe what the style is of an HTML file and how the elements within the webpage are displayed. Without style sheeting, the raw HTML pages are displayed in the web browser, as is depicted in the upper side of Figure 65. There are no frames, classes, buttons, sidebars, etc. It is just a web page that responds to the data that is available on the webpage in the back-end. Therefore, CSS is necessary to create a user-friendly environment based on the GUI. Every line, text container, frame, or picture must be style sheeted. What can be noted from previous figures, is that every line in the script is always referred to as a class. Sometimes, the class is placed within a class, to style sheet the frame and the text within differently. These classes refer to the CSS files in the API, which are placed in the public folder. With CSS, you can edit the background colour, font-size, font-weight, width, height, padding, border-style, buttons, text-align, margin, etc. There are many possibilities, whatever is the choice of the developer. On the bottom of Figure 65, the same page is depicted, but now after it is style sheeted by CSS. The difference is immediately recognizable, in which this part of the figure is a clean, structured, and user-friendly interface. In the API four .css files are developed to style sheet the complete API.

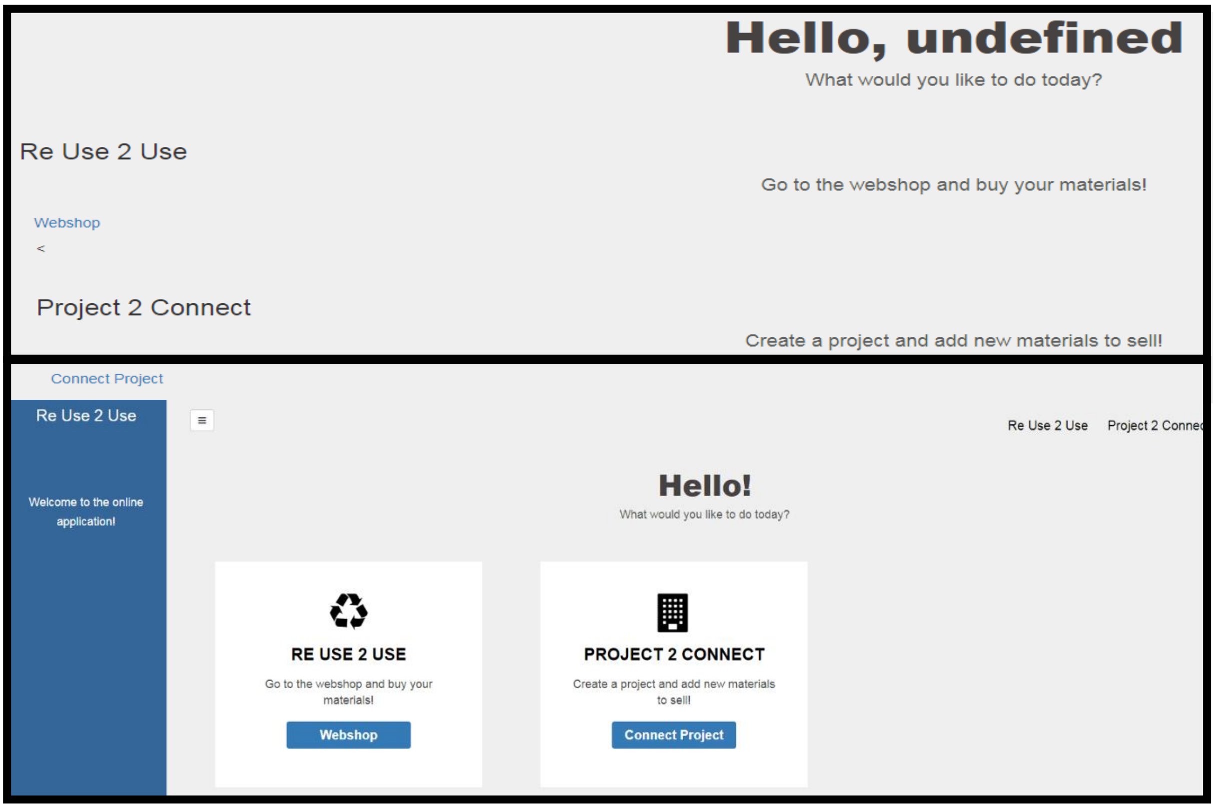


Figure 65: Difference between webpage without (top) and with (bottom) CSS

Navigation.css

The navigation.css file, as the name implies, stylesheets the navigation bars, header and sidebars. To keep a clean API, the style sheeting is divided over the four .css files. To keep the distinction between the two environments, both sides got their colour, to keep it clear for the end-user on which side he or she is located. Re Use 2 Use is style sheeted with blue and Project 2 Connect in green, same as their logos. The *header* is placed at the top of the page, which is available on every web page. The *sidebars* are placed on the left side of the screen, in which the input differs based on the environment. In Figure 66, an example is given of how one part of the sidebar of the Re Use 2 Use side is style sheeted. The sidebars contain links, as they are described in sub-chapter 6.3.4. Furthermore, the logo of the environment is added as well. The sidebars are developed with a responsive feature. This means that, if the online application is opened on a web browser of a mobile phone, the webpages shrink to keep a clear and clean web page. A toggle menu is programmed, in which the sidebar will be placed. If the end-user wants to search in the menu, it can press the toggle menu and his or her options will be available. To create this responsive webpage, the file is divided into two parts, based upon the total width of the screen. Below a width of 425 pixels, the menu is placed in the *toggle*. Above this width of

```
#sidebar-wrapper-reuse2use {
    z-index: 1000;
    position: fixed;
    left: 250px;
    width: 0;
    height: 100%;
    margin-left: -250px;
    overflow-y: auto;
    background: #333333;
    -webkit-transition: all 0.5s ease;
    -moz-transition: all 0.5s ease;
    -o-transition: all 0.5s ease;
    transition: all 0.5s ease;
    background-color: rgb(52, 102, 154);
}
```

Figure 66: Example of .css file of the navigation bars

pixels, the webpages are divided on a broader scale. The addition of the responsive menu and webpages increases the extensibility of the online application. It can not only be opened on web browsers on computers and laptops, but also tablets and mobile phones. This increases the possibilities to get a higher number of end-users for the online application.

Custom.css

The overall style sheet for the API is named as the custom.css file. In this file, especially the middleware of the webpages is described. The body of the webpages contains many different text-columns, text-containers, buttons, and all other sorts of input. Many fonts are used and their sizes, colours, weights, are defined in this file. Also, the alignment of text and columns are defined in this file. The classes of these columns and centres always refer to the custom.css file. Every webpage of the API is always style sheeted, mainly with this file.

Split_screen.css

This file is especially created for splitting the screen in the *add product* webpage in the Project 2 Connect environment. With the fields for adding the products on the left, and the table on the right side of the screen, the webpage needed to be split. In this way, it is possible to stylesheet the sides separately. Otherwise, the webpage will be seen as one part. With the split-screen method, this consists of two parts. The input of the table is a large data stream of which the sizes and weights should differ from the fields for adding the product.

Product.css

The last .css file is the style sheet of the products and the webshop. With retrieving the project information and product's characteristics, the webshop is still not ready for use. The information should be placed on many frames to present as many as possible products in the webshop. The header of the frames, the image, the price, and contact information, all need to be style sheeted differently. Programming a frame like this, will create an attractive webshop based on clean frames with the product's information. The attractiveness of the webshop is necessary to increase the number of customers. If the customer gets a good feeling when using the webshop and after his acquisition, he or she might return. Therefore, the style sheeting of the webshop and the product's profile are very important for the experience of the webshop customer.

6.3.9 | Complete Tree Structure API

To end the sub-chapter of the Application Development, the complete tree structure of the API will be highlighted. In the first part, some brief structures are already depicted. But now, the complete tree structure will be composed, including the Project 2 Connect side, Re Use 2 Use side, and the CSS files. Also, the *public* folder will be added. This folder contains the sub-folders: *javascript, css, files,* and *images.* The *javascript* folder contains the files with the scripts as they are discussed in the Project 2 Connect side. The *css* folder contains the files from the CSS sub-chapter. The *files* folder is the internal storage, as was defined in Methodology (Section 4) and it consists of the Solibri Role Zip file (ITO, classifications and ruleset) and the Project Report Template Excel spreadsheet file. The *images* folder contains the images that are used within the API and for the style sheeting of the webpages. These are the logos from both sides of the online application and the stars from the score table of the *reusable label*. With this last folder of the API, every single file of the API is now discussed, described, and defined. Therefore, the next Figure 67 gives an overview of the API of the root folder(s) and the files that are stored within these folders. The application development is now completed. In the next Section 7 (Proof of Concept & Validation), the prototype of the online application will be tested with the proof of concept and validated and evaluated, based on the quality and the rate of performance for the end user experience of the online application.

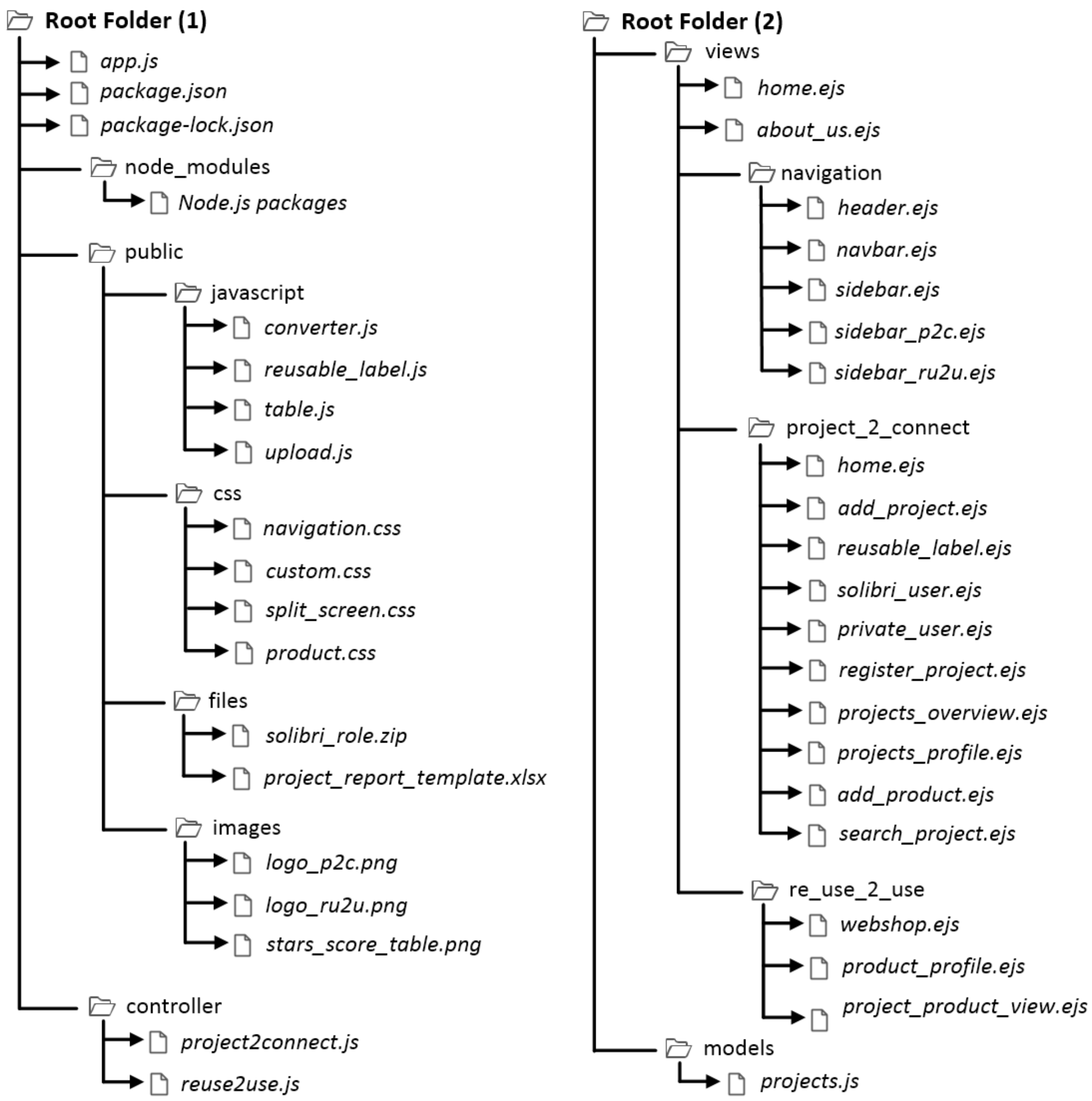
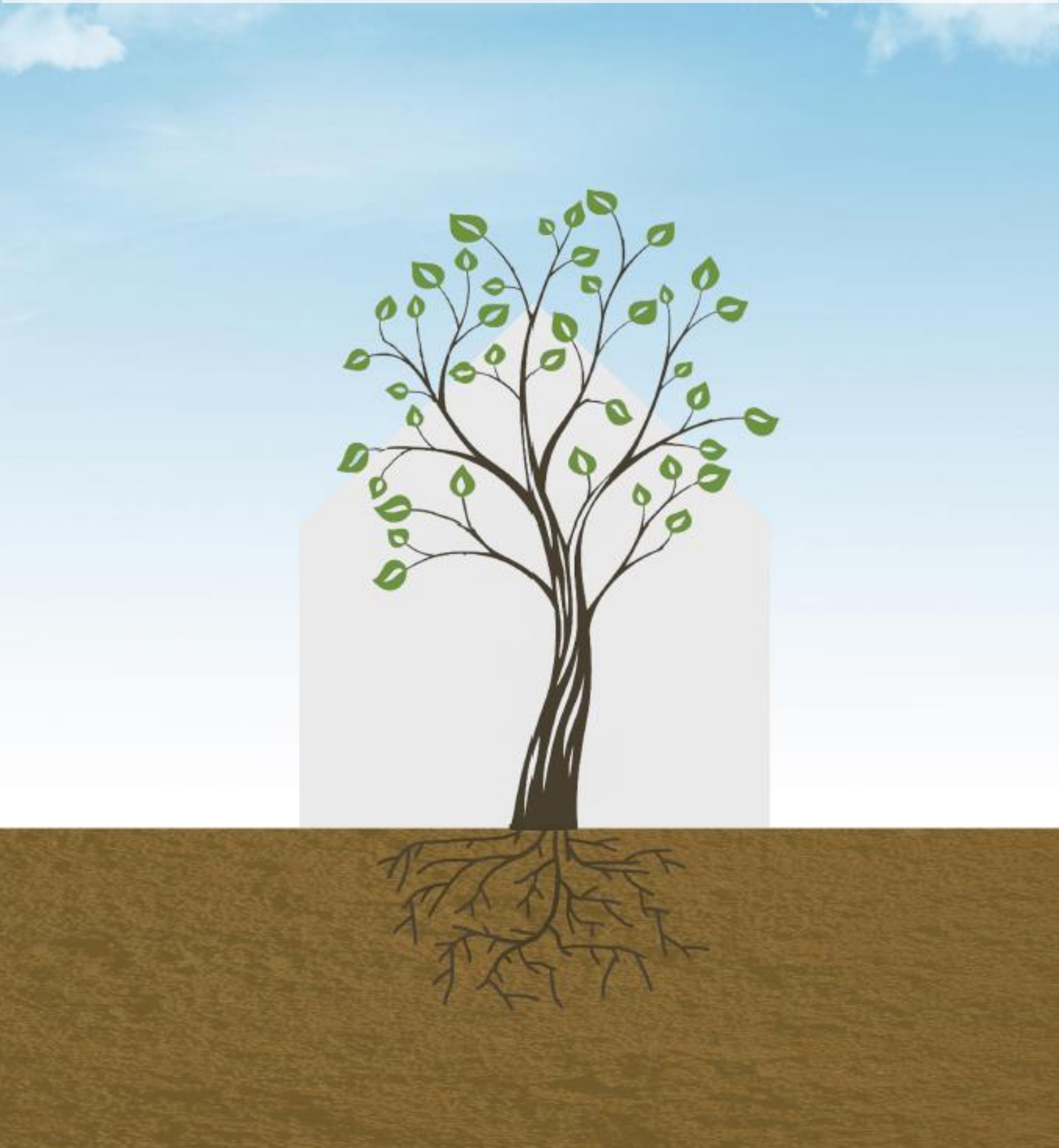


Figure 67: Completed tree structure in the root folder in the API

Section 7

Proof of Concept & Validation



7 | Proof of Concept & Validation

The first three phases of the Rapid Application Development are executed: (1) The requirements for the online application are analysed and determined, (2) the system was designed to create an ideation of the outcome, and (3) the online application was developed while programming the back-end. In this section, the fourth and final phase will be executed: (4) testing the online application that is developed. To execute this phase of the RAD, this section is divided into five sub-chapters. In the first sub-chapter (7.1), the online application will be tested, using the Alpha Testing method. Before launching the prototype on the market, the prototype will be tested by the developer. The performance of the online application will be tested after the first development phase. Once the gaps and/or bug and failures are highlighted, the second development phase starts, which is an optimization phase (sub-chapter 7.2). The results from the Alpha Testing will be optimized to increase the value of the online application. Once this is executed, the online application will be compared with the best-case scenario (sub-chapter 7.3). The best-case scenario is what the desired data output of the Re Use Index – ILS will look like if it is presented in the webshop. After this comparison, the Re Use Index – ILS is developed as an actual standard to be used in the AECO industry (sub-chapter 7.4). The proof of concept and the overall results of the research will be validated in the last sub-chapter (7.5).

7.1 | Alpha Testing

To ensure the success of the online application, it will be tested using the Alpha Testing method. After developing the back-end of the online application, it is time to test the front-end of the online application to filter out bugs and/or failures that are not discovered yet. Also, this phase aims to find the missing parts of the online application that need to be fulfilled to increase the user-experience of the end-users (Professional QA, 2020). Therefore, a real-user environment will be simulated using the two different end-users: the project owner and the webshop customer. This sub-chapter aims to improve and upgrade the quality and the functionality of the online application, but most importantly shows the test results in the front-end. Most design changes are already executed during the development phase, so this test phase is mainly focussed to show the delivered and effective software, from the perspective of the user experience. The front-end results, which are the Graphical User Interfaces, will be highlighted by Figures. The input of the non-graphical information originates from the parameters determined in the BIM model data juxtapostion (sub-chapter 5.3). This procedure aims to create an input for the project owner side, based on a BIM model retrieved from the database of BASED BIM Management & Consultancy. Otherwise, the online application could not be tested on a real BIM model, which is important for this research to concentrate on the stock of existing buildings. In sub-chapter (7.3), the difference between the input of this BIM model and the desired output (sub-chapter 5.2) will be described. In this sub-chapter, the value of a BIM model with a higher level of non-graphical BIM data will be defined with the best-case scenario. The improvements and necessary performance upgrades that are concluded in this sub-chapter will be considered and executed as being optimizations in the next sub-chapter (7.2).

7.1.1 | Real-Users Environment

The simulation of the real-user environment will be introduced by two fictional persons: the project owner and the webshop customer, willing to buy reusable building materials.

The first person is John Doe (Figure 68), which is the owner of an apartment complex. This building gets deconstructed to the best of his abilities, in order to keep the value of the building materials maintained in which they can be reused immediately. He has a BIM model available in IFC file format, which he checks on its value using the SMC software. During the life-cycle of the building, renovations, and maintenance, as well as possible transformations for the internal layout of furniture is always updated in the BIM model. The data from the model is identical to the building as it is nowadays. He likes to upload his project, to be able to sell the products that are assessed with the possibility to be reused, in the webshop of the online application.

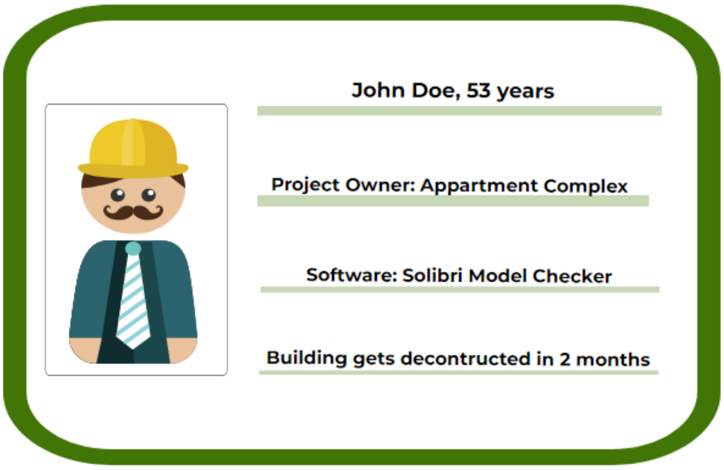


Figure 68: End-user John Doe, project owner

The second fictional person is Jane Roe (Figure 69), who is building a new house for her and her family. Her husband is a carpenter who can build their new house himself. Jane lives on a very sustainable footstep, and always tries to think about what is best for the environment, every contribution on a personal level is a step forward. She, therefore, is searching for second hand, reusable building materials, from projects which get deconstructed nearby the place where their new house will be located. Her husband can use these building materials to construct their new house. The 'waste' from another building, can get a new life in their new house. Her participation towards a circular economy means a lot for Jane and her family. She, therefore, keeps a close eye on the webshop of Re Use 2 Use to purchase the building materials which they need for the construction. Time is not an issue; the reusable building materials are a must.

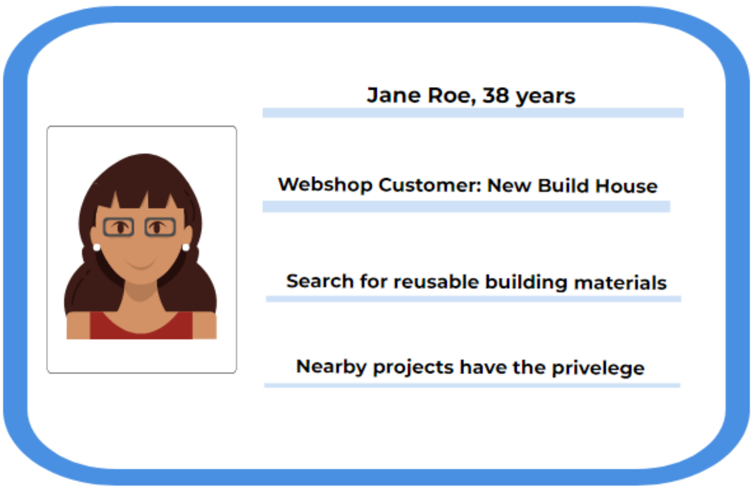


Figure 69: End-user Jane Doe, webshop customer

7.1.2 | Project 2 Connect

John wants to add his project to the online application. After entering the *home*-webpage, he chooses for the Project 2 Connect side instead of the Re Use 2 Use side. In Figure 70, the *home*-webpage of Project 2 Connect is highlighted. On this page, John has two decisions: to add his project to the database or to assess his products based on the circularity indicators of the reusable label. John chooses to first-rate his building materials to get a better insight of which have the potential to be reused.

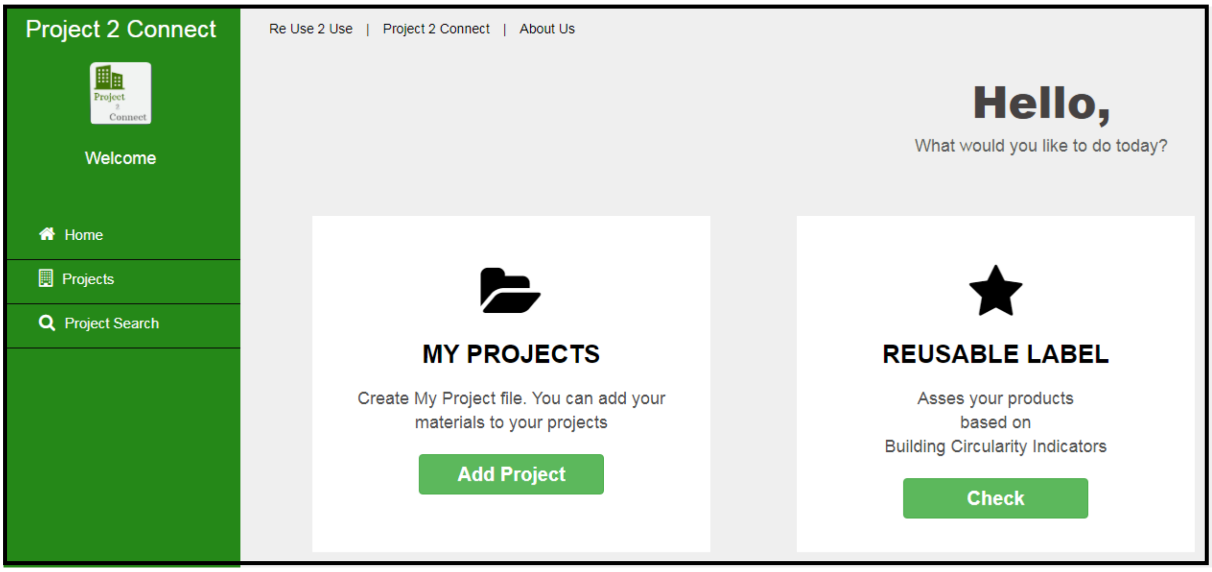


Figure 70: Homepage of the Project 2 Connect environment

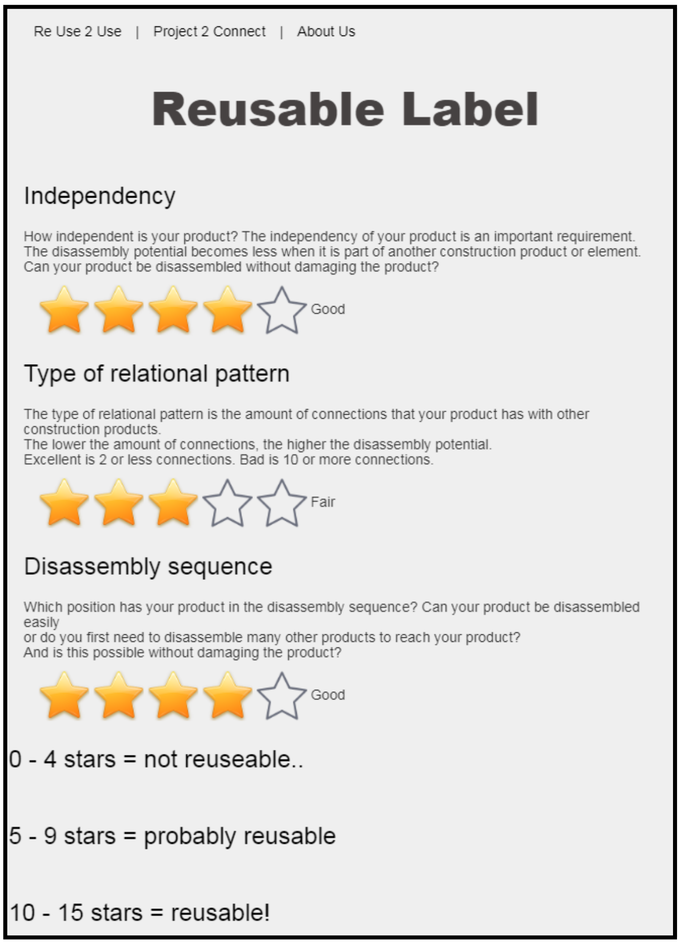


Figure 71: Reusable label page, with assessment tool

John has entered the *reusable_label*-webpage (Figure 71). On this page, circularity indicators are listed with a description. Based on these circularity indicators, the building material can be assessed. E.g. the first circularity indicator is the 'independence'. This indicator scores the independence of the product: is it independent from other materials or elements, or does it need to get damaged while disassembling. In this example, John is rating an assembly door frame, which can easily be disassembled when deconstructing the building. The assembly door frame is placed in the project, after constructing the internal walls, and can, therefore, be disconnected from the walls, which makes the building materials independent. It does not get the full score, since it must be damaged a bit, this is also stated in the second circularity indicator 'type of relational pattern'. The assembly door frame is connected with many screws to the wood that is placed between the door frame and the wall. To unscrew the door frame, it must be damaged since the door frame is finished with paint. The building materials score again very high in the 'disassembly sequence'. The door frame can at all times be disassembled from the project, it is not dangerous and does not affect any structural part of the building. Overall, the door frame scores 11 out of 15 stars. The rating below gives therefore the rating that the building materials are reusable. John can execute this assessment option with every single building material from his building, of which he doubts if it could be reused or not.

Once John conducted his list with potential reusable building materials from his project, he goes back to the *home*-webpage, using the home button in the sidebar. Back to the point where he started, he chooses now to add a project (Figure 70). When entering the *add_project*-webpage, John has three options which he can choose from: *solibri_user*, *private_user*, and *single_product* (Figure 72). John has a BIM model available in an IFC file format, which he views in Solibri. Therefore, John can use the Solibri path to convert his project information and upload this underneath his project, while registering it.

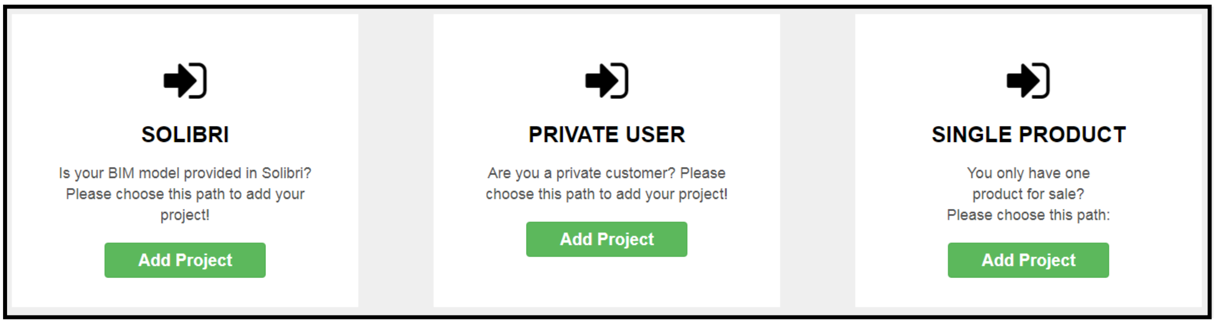


Figure 72: Three options for the project owners

Figure 73: Converted file and Solibri Role zip file

John enters the *solibri_user*-webpage (Figure 75) and has five steps he has to follow. In the first step he can press the button and the *Reuse2use_Template.zip* file will automatically be downloaded (Figure 73). This zip file contains the Solibri Role, as it is pre-defined by the BIM Model Data Juxtaposition (Section 5). John has to extract the .zip file on his computer, open the SMC and within Solibri he can upload the Solibri Role. This role contains a ruleset, classifications, and most importantly the ITO (Section 5, Figure 25). John only needs to go to the ITO tab in Solibri and the columns are automatically pre-sorted (Section 5, Figure 22). John presses 'take off all' to extract every single parameter from every single object in Solibri that fits the requested output of the predefined columns. Then, John has to press 'export report' and the Excel spreadsheet will be downloaded which contains the input of the ITO as John did retrieve from the BIM Model. This means that the non-graphical BIM data as is requested for the online application is now retrieved from Solibri, exported as .xlsx file and ready to be used in the online application. This is all possible within a minute of time, without any effort.

John is now ready to conduct the second step, uploading the report. When he presses the button, John can search within his folders to the ITO Report file on his computer. He can upload the ITO Report and then move to step 3. The only thing he needs to do is to press the button 'convert report' and the back-end of the online application (Section 6, Figure 49) will convert the file to .json. The script that is provided in the back-end accepts .xlsx files and is able to convert the file to an .json file, without loss of information. This only needs to happen, because

Javascript prefers to read .json files in order to work with that data. The structure of this type of file is cloud-based, which makes the potential for connections easier. The difference between both files is highlighter in Figure 74. The ITO Report .xlsx file is depicted in the upper

Name	Component	Count	IfcEntity	NlSfb	Material	Area	Height	Length	Width	Thickness	Volume
16_WA_beton ihw 400_gen	Wall	1	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	4,76	500	9520		400	1,9
16_WA_beton ihw 400_gen	Wall	1	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	4,96	500	9920		400	1,98
16_WA_beton ihw 400_gen	Wall	1	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	48,83	500	97660		400	19,53
16_WA_beton ihw 400_gen	Wall	2	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	2,24	500	4470		400	0,89
16_WA_beton ihw 400_gen	Wall	2	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	5,25	500	10500		400	2,1


```
[{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":1,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12"
{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":1,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12",
{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":1,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12",
{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":2,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12",
{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":2,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12",
{"Name":"16_WA_beton ihw 400_gen","Component":"Wall","Count":3,"IfcEntity":"IfcWallStandardCase","NlSfb":"16.12",
```


Figure 74: Excel Report (Top) JSON Report (Bottom)

part of the figure whereas the ITO Report .json file is depicted on the bottom of the figure. The data input is exactly the same, different is the file structure.

John can now move to step 4, which is downloading the file (Figure 73). The ITO report is exported, uploaded, converted, and downloaded. These steps also can be fulfilled within a minute. The project is now ready to be registered, so John moves to step 5, in which he will be redirected to the *register_project*-webpage. A sidestep will be explained, before entering the *register_project*-webpage. If John was not the owner of a BIM Model, but was a private client for the online application, he was still able to upload his data. He then chooses the *private_user* path in Figure 72. This path shows the same steps as for the Solibri user, the only difference is that the user cannot download the Solibri Role, but he can Download a Project Report Template. This template is depicted in Figure 76. The template consists of the same columns, which again are pre-defined. The private user does probably not have a BIM model at his disposal and maybe does only have a single house which he is deconstructing and wants to add the building materials to the online application. With the creation of this path, private users can use the online application as well. They download the template and fill in

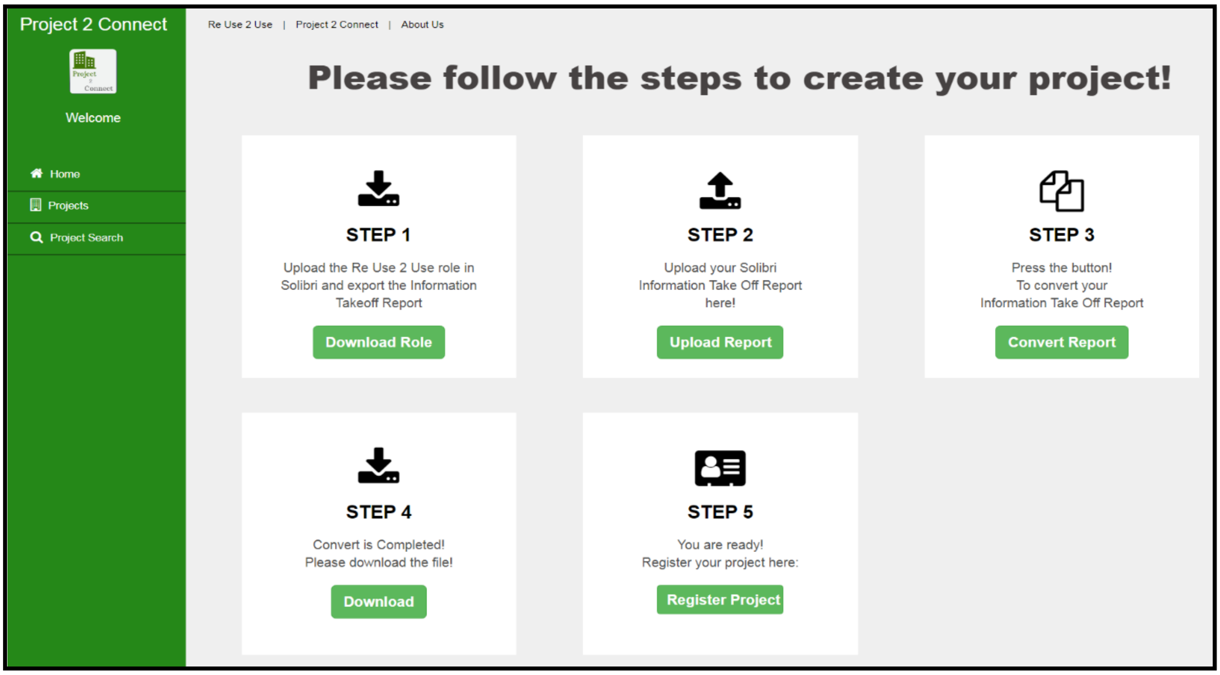


Figure 75: Solbri webpage, 5 steps to follow before registering the project

their data in the columns as they are asked to do. Once this is done, they follow the same steps as John did as Solibri user. Upload the Project Report, convert, download (.json file), and go to the *register_project*-webpage. The last path is the single product path. If the user only wants to add a single product, he or she does not have to upload a report, but only needs to register the project and add a single product to the project. If a user has an own project or multiple products he or she wants to add, it is recommended to follow one of the two first paths, to add the products to the project very easily in a later stage.

Name	Component	Count	IfcEntity	NlSfb	Material	Area	Height	Length	Width	Thickness	Volume
Name of product	*Classification*	*How many?*	*Not Required*	*Not Required*	*Which material*	*In m2*	*In mm*	*In mm*	*In mm*	*In mm*	*In m3*
Structural Column	Column	1	-	-	Steel	2	2000	300	300	-	0,8
External Window	Window	5	-	-	Wood. Red Paint	1	300	300	-	45	0.15

Figure 76: Project Report Template

On the *register_project*-webpage, John can fill in the contact information of him and his project (Figure 77). This contact information is important to get in contact with the webshop customer. The potential buyer needs to be able to reach the project owner to come to an acquisition. The address and city are important to create a range for the potential buyer, to know if the products can be bought nearby or not. John already uploaded his ITO Report .json file, since the project report status is 'file uploaded'. Once John presses 'register', the project will be stored in the back-end. The controller sends the data to the model, which stores the data in MongoDB (Figure 78). By storing this data in the external database of MongoDB, the online application can save and fetch more data, which is connected to this project, once John enters other webpages or refreshes the webpages. If this connection it is not developed, the data gets lost and the end-users are not able to upload data, because it cannot be stored in the back-end of the online application. What also can be noted in Figure 78, is that automatically an ID is created for the project of John. This ID is important to create unique webpages, which only belong to this project, as was previously mentioned in Section 6.

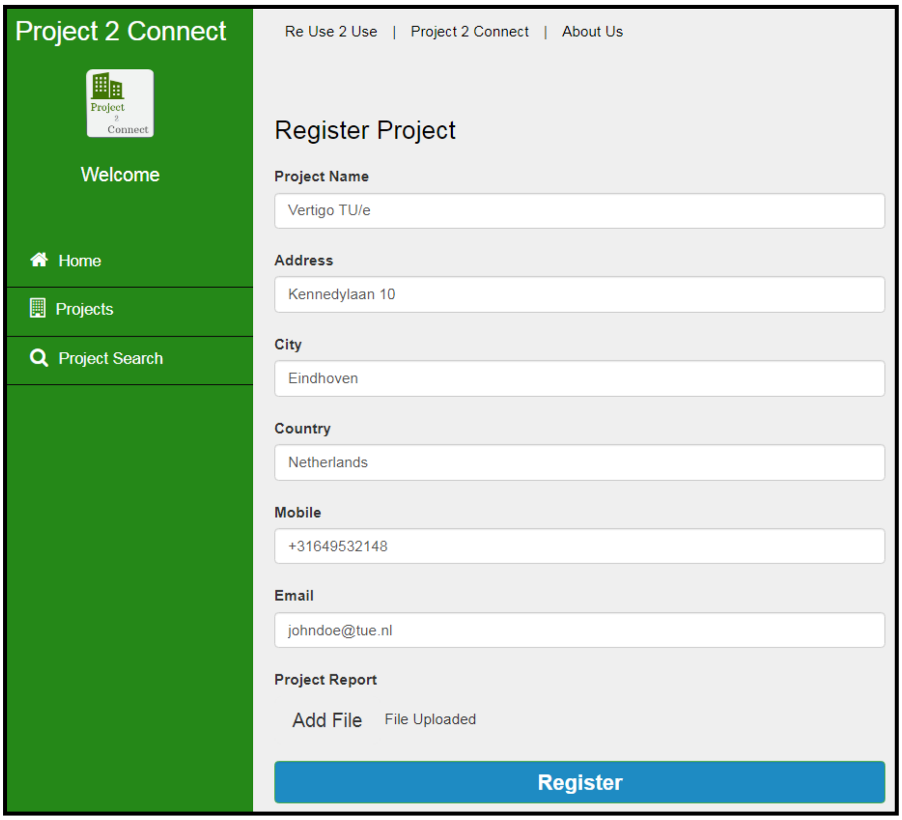


Figure 77: Register project webpage, adding project contact information

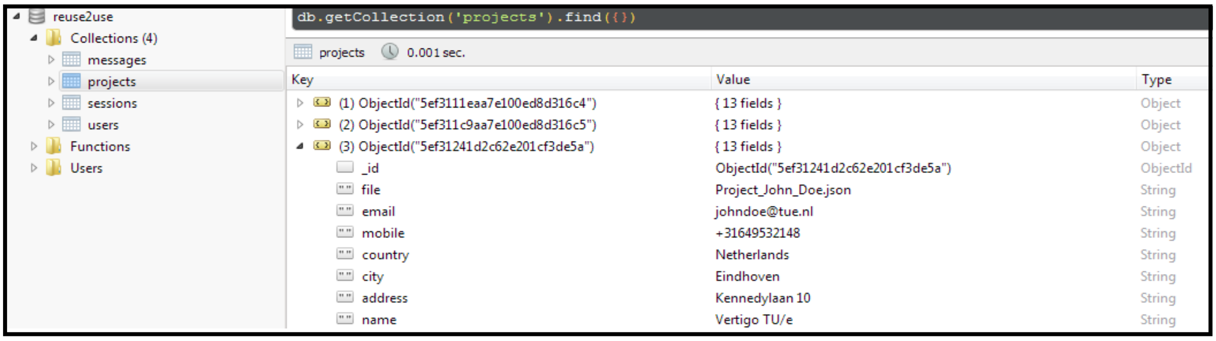


Figure 78: Project registered and stored in MongoDB

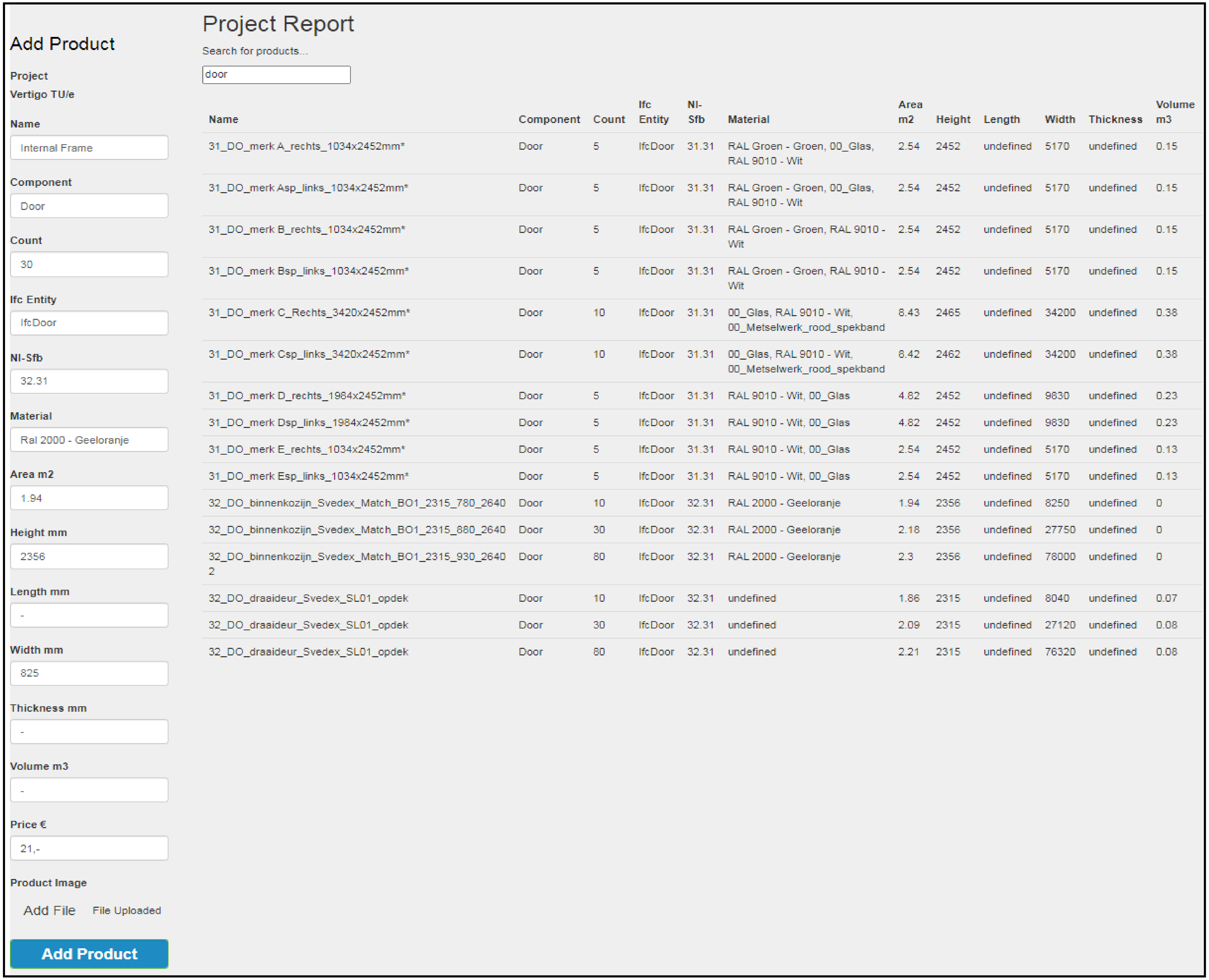
After registering the project, John will see the *project_overview*-webpage (Figure 79). On this webpage, every project will be presented. Not only the project of John, but also other projects. And if John has multiple projects, they will be presented as well. In this way, the projects can easily be managed, and products can be added in a later stage.

Figure 79: Three examples of project in the project overview

The *project_profile*-webpage shows the project contact information and gives the options: add product or view project products (Figure 80). The overview of the products can always be viewed after adding the product, but John first wants to add his products to his project.

Figure 80: Project profile webpage

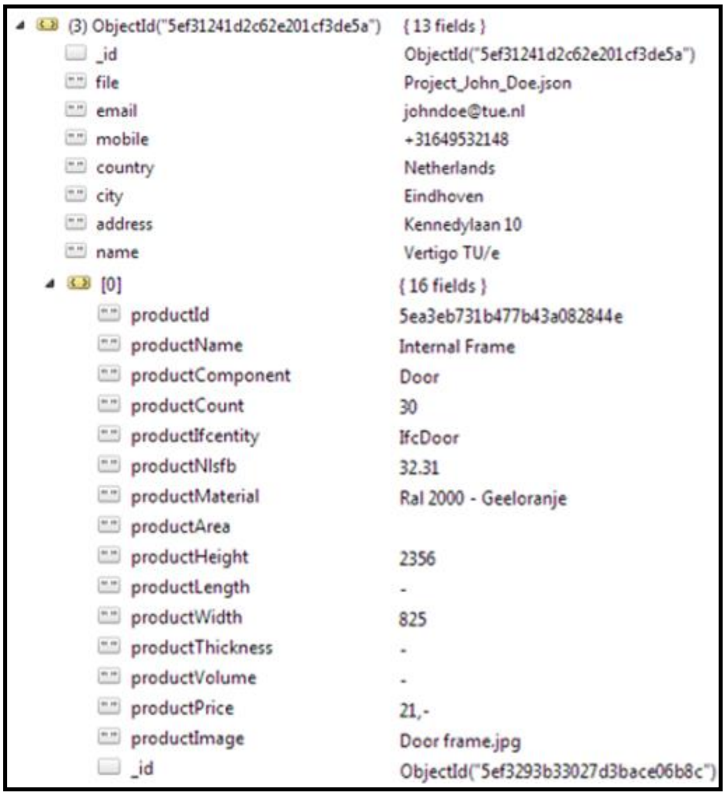
John enters the *add_product*-webpage (Figure 81). The webpage is split into two parts, on the left part, John can add his products with all the available characteristics. He just needs to give a value to the fields. On the right side, John can see that the data from his ITO report is presented, which gives him direct access to view all his products from his project. In the table, the same columns and parameters as they were exported from SMC are presented. If this were an example of the private user, the non-graphical information will be presented in the same way. Once any report is uploaded (.json or .xlsx), the data would be presented in this table underneath the products which it belongs to. The table is an active table, which refers to the search tool that was developed in the back-end (Section 6, Figure 59). The search tool makes it possible for John to search for any product within his project report. As can be seen in the example, John searched for ‘door’, since he was sure that the internal door frame had a high potential to be reused, because of the reusable label assessment tool. The hits in the table are all connected to the search hit in the search bar. Every product in the project, which contains a certain parameter with the word ‘door’ will be presented. This of course is possible for any other word, letter, or number. Other search functionalities and filters could be developed as well. On the left side, the fields are almost corresponding to the columns of the table, only the fields ‘price’ and ‘product image’ are added as well.



Name	Component	Count	Ifc Entity	Nl-Sfb	Material	Area m2	Height	Length	Width	Thickness	Volume m3
31_DO_merk A_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, 00_Glas, RAL 9010 - Wit	2.54	2452	undefined	5170	undefined	0.15
31_DO_merk Asp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, 00_Glas, RAL 9010 - Wit	2.54	2452	undefined	5170	undefined	0.15
31_DO_merk B_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, RAL 9010 - Wit	2.54	2452	undefined	5170	undefined	0.15
31_DO_merk Bsp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, RAL 9010 - Wit	2.54	2452	undefined	5170	undefined	0.15
31_DO_merk C_Rechts_3420x2452mm*	Door	10	IfcDoor	31.31	00_Glas, RAL 9010 - Wit, 00_Metselwerk_rood_spekband	8.43	2465	undefined	34200	undefined	0.38
31_DO_merk Csp_links_3420x2452mm*	Door	10	IfcDoor	31.31	00_Glas, RAL 9010 - Wit, 00_Metselwerk_rood_spekband	8.42	2462	undefined	34200	undefined	0.38
31_DO_merk D_rechts_1984x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	4.82	2452	undefined	9830	undefined	0.23
31_DO_merk Dsp_links_1984x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	4.82	2452	undefined	9830	undefined	0.23
31_DO_merk E_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	2.54	2452	undefined	5170	undefined	0.13
31_DO_merk Esp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	2.54	2452	undefined	5170	undefined	0.13
32_DO_binnenkozijn_Svedex_Match_BO1_2315_780_2640	Door	10	IfcDoor	32.31	RAL 2000 - Geeloranje	1.94	2356	undefined	8250	undefined	0
32_DO_binnenkozijn_Svedex_Match_BO1_2315_880_2640	Door	30	IfcDoor	32.31	RAL 2000 - Geeloranje	2.18	2356	undefined	27750	undefined	0
32_DO_binnenkozijn_Svedex_Match_BO1_2315_930_2640 2	Door	80	IfcDoor	32.31	RAL 2000 - Geeloranje	2.3	2356	undefined	78000	undefined	0
32_DO_draaideur_Svedex_SL01_opdek	Door	10	IfcDoor	32.31	undefined	1.86	2315	undefined	8040	undefined	0.07
32_DO_draaideur_Svedex_SL01_opdek	Door	30	IfcDoor	32.31	undefined	2.09	2315	undefined	27120	undefined	0.08
32_DO_draaideur_Svedex_SL01_opdek	Door	80	IfcDoor	32.31	undefined	2.21	2315	undefined	76320	undefined	0.08

Figure 81: Add product webpage, the table with the retrieved data from the project report (right) and the fields for the final input for the webshop (left)

The decision to create the fields on the left side, which requires a manual input, is because the non-graphical BIM data is not always very brief or relevant. E.g., the name of the internal door frame contains many values which do not increase the value of the products, once it is added to the webshop. If every single field of a product would be directly uploaded from the table to the webshop, the webshop gets very unattractive, and the webshop customer receives a lot of data which only makes it unclear what product they are buying. Also, fields that contain mistakes, e.g. the fields of width from the last products in the table are often wrong. The project owner is still able to resolve mistakes from the BIM model. John can fill in the fields on the left side using the input of the table. Most fields can immediately be copied, some fields need some adjustments. The final step is to give a price to the product(s) and to upload an image. This tool that is added in the back-end immediately creates a higher value of the product, since the buyer always wants to see the product which he or she is willing to buy. John can upload any image that is located in the folders on his computer. Once John is satisfied with the input of the product, he can press the button. The back-end stores this data by the controller and the model in MongoDB underneath the project which it belongs to, in this example Vertigo TU/e. As is highlighted in Figure 82, the characteristics are stored in the project. The tree structure is visual, in the first column, the project is stored with the information and project report. In the second column, the product is stored. The characteristics are presented, the price, and the file of the image. With this structure in MongoDB, it is possible to store the products underneath the project. This is important for the webshop, because, in this way, the product profile can always present the project contact information as well, since they are connected. Every product gets an ID which makes them unique. But this ID is always connected to the project. If the webshop customer wants to see a *products_profile* of a product, the webpage can only refer to that product, but will send forward the same project contact information, as long as it is stored underneath that project.



(3) ObjectId("5ef31241d2c62e201cf3de5a")	{ 13 fields }
_id	ObjectId("5ef31241d2c62e201cf3de5a")
file	Project_John_Doe.json
email	johndoe@tue.nl
mobile	+31649532148
country	Netherlands
city	Eindhoven
address	Kennedylaan 10
name	Vertigo TU/e
[0]	{ 16 fields }
productId	5ea3eb731b477b43a082844e
productName	Internal Frame
productComponent	Door
productCount	30
productIfcentity	IfcDoor
productNlsfb	32.31
productMaterial	Ral 2000 - Geeloranje
productArea	
productHeight	2356
productLength	-
productWidth	825
productThickness	-
productVolume	-
productPrice	21,-
productImage	Door frame.jpg
_id	ObjectId("5ef3293b33027d3bace06b8c")

Figure 82: Product stored in project MongoDB

John has now the option to add more products or projects to the online application. But for now, John is ready with uploading his project and products. He can always come back and add more products on another day, the data is stored and therefore never gets lost.

7.1.3 | Re Use 2 Use

Jane enters the website www.reuse2use.nl, which she has heard from friends that you can buy reusable building materials on that platform. Her husband asked her to search for some internal door frames. Once he knows the dimensions, he can start constructing the internal walls and leave the openings for the internal door frames. The first webpage Jane sees is the *home*-webpage. She has the option to go to the webshop or to add a project. Jane is not a project owner, but a webshop customer, she therefore chooses the webshop. Once entering the *webshop*-webpage, she immediately enters the Re Use 2 Use environment (Figure 83). The webshop presents just the most important aspects of the products. The webshop customer only needs to see the: name, category, image, price, location, and reusable score. This keeps the webshop clean of not certainly relevant information on first sight. In the webshop, there are already three products added from three different projects. If there were more products added, Jane could choose a category in the sidebar which would direct here to that particular category, or she can search for a product with the 'product search' tool. But for now, she already sees a product that gets her attention. The internal frame that is presented, located in Eindhoven. She lives in Uden, so the transport will not be environmentally-unfriendly, because it is nearby. Jane clicks on the product to receive more information about the product and the project owner.

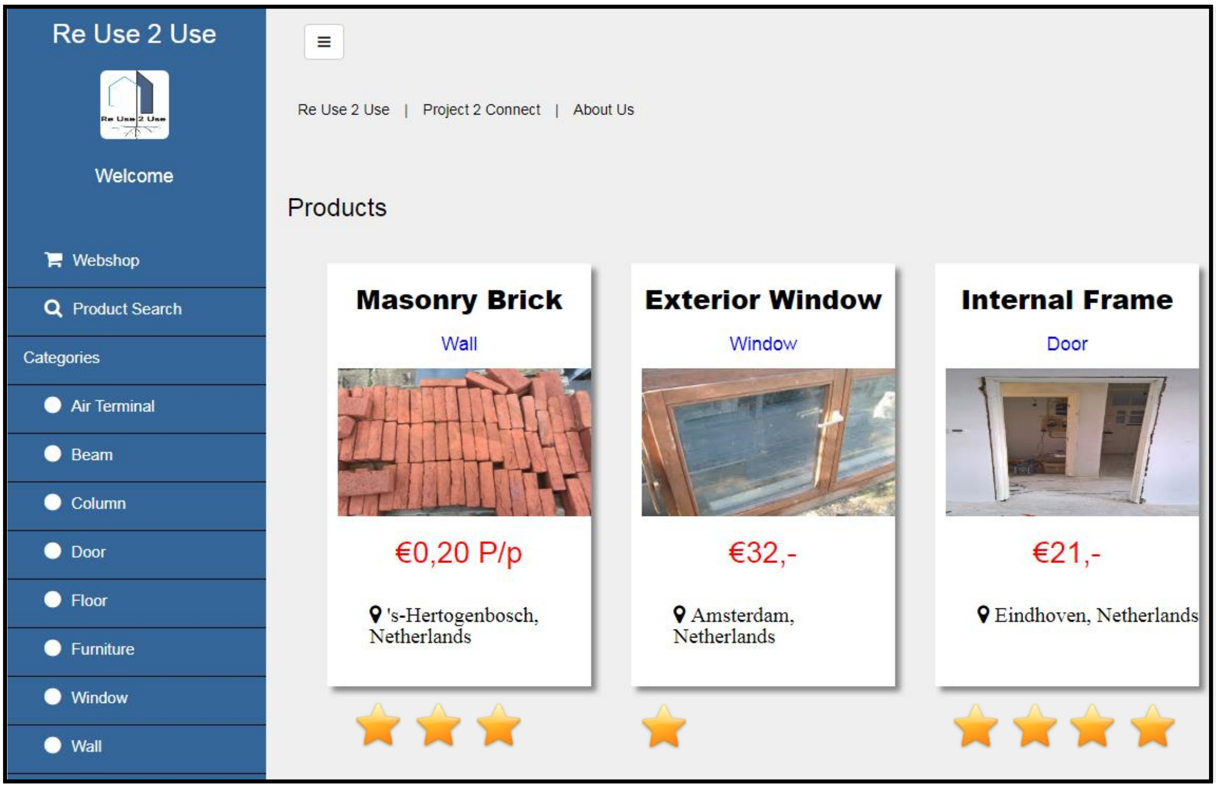


Figure 83: Webshop Re Use 2 Use

Jane is watching the product's profile on the *product_profile*-webpage (Figure 84). In one overview, she can see the product's characteristics as well as the project information. This is the most important part of the online application. This webpage shows the connection between the MVC pattern in the back-end, the external database of MongoDB, and the web browser. The project contact information and the product characteristics are stored by the MVC pattern in MongoDB. On another page, on another time, this data is retrieved from MongoDB by the MVC pattern. This data is sent to the View, which renders the data and sends back the presentation. Therefore, on this webpage, the product characteristics are presented, as well as the project contact information. Jane wants to have six internal door frames, and is going to contact John, to get to the acquisition of the reusable building materials. Jane can also see the other products that are added to the project by proceeding to the *project_product_view*-webpage by pressing the button (Appendix IX). With this option, she can see if there might be more products she is interested in since she is already planning to buy building materials from the project.

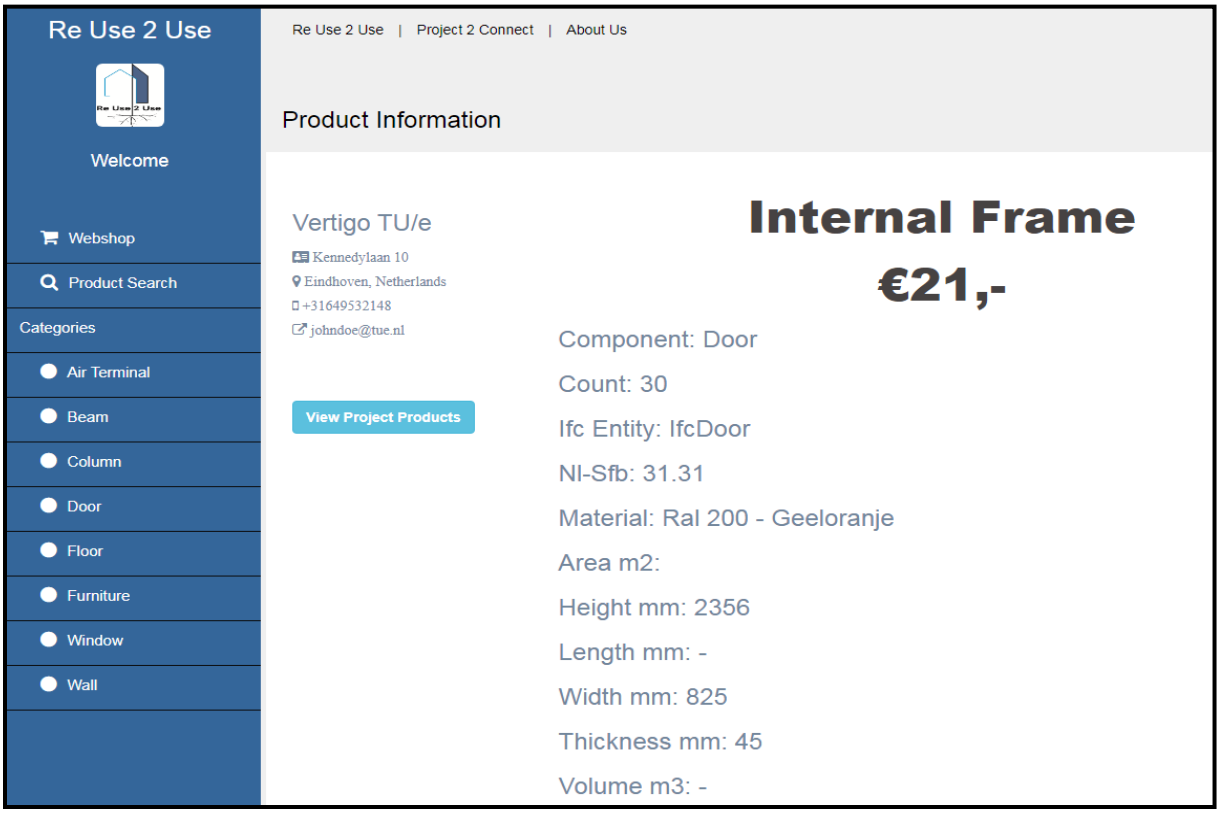


Figure 84: Product's profile of internal frame

7.2 | Optimization

After conducting the Alpha Testing phase with the real-user environment, some additions can be made. The most important bugs, software failures, and errors were already adjusted in the application development phase (Section 6.3). During the development of the coding in the back-end, the software did already crash many times by the use of wrong functions and unfinished scripts. These errors were already resolved in the back-end and changed when needed. The optimizations, therefore, are mainly focussed on the user experience resulting from the test with the real-user environment. The optimizations are divided into two topics: user account and user experience.

7.2.1 | User Account

Since the online application is working with project information and personal contact information, the addition of a user account would provide more privacy for the users. Over the last years where everything went from paperwork to digital environments as well as personal information on social media, etc., did the privacy aspect become an important topic in the society. Therefore, if the online application wants to be able to be launched on the market, the privacy aspect must be implemented in the online application. By creating a user account, the projects that are uploaded by the project owner and the belonging data can only be viewed by the project owner of that particular project. The back-end programming of the user account works in the same way as with adding the project to the online application. First, in the root file *app.js*, a connection must be made between the Node.js server and the routes. The routes connect the Node.js server with the controllers, which on their terms are connected to the model. In the Model folder, a new file is created: *users.js* (Figure 85). By this script, the user full name, email, and password can be stored in MongoDB. Storing this data makes it possible for the online application to recognize the user account information after signing in. The next time, the user only needs to login and the online application provides their environment with their belonging projects and products.

```
var mongoose = require('mongoose');
var bcrypt = require('bcrypt-nodejs');

var userSchema = mongoose.Schema({
    fullname: {type: String, required: true},
    email: {type: String, required: true},
    password: {type: String},
});

userSchema.methods.encryptPassword = (password) => {
    return bcrypt.hashSync(password, bcrypt.genSaltSync(10), null);
}

userSchema.methods.validPassword = function(password) {
    return bcrypt.compareSync(password, this.password);
};

module.exports = mongoose.model('User', userSchema);
```

Figure 85: Javascript of user account

To create this extra environment in the online application, an extra controller is needed: *users*. And extra views were added in the folder: *users.* A new homepage must be created, since the 'old' homepage is not the first page anymore, which the end-user will see when entering the online application. The first page is the *index*-webpage. The second and third webpages are the *sign_in*-webpage and the *login*-webpage. First, the end-user needs to sign up before getting access to the online application. Next time, the end-user does not need to sign up anymore, in that case, the end-user just needs to login, since MongoDB will recognize the login code. An extra feature can be to send an email if the end-user has forgotten his or her user account login combination. The files that were adjusted (app.js) and the new files and new folders that are part of the tree structure of the back-end of the online application are highlighted in Figure 86 (left side). In Figure 87 (right side) is depicted how the *sign_up*-webpage is presented in the web browsers once the end-users of the online application want to sign up.

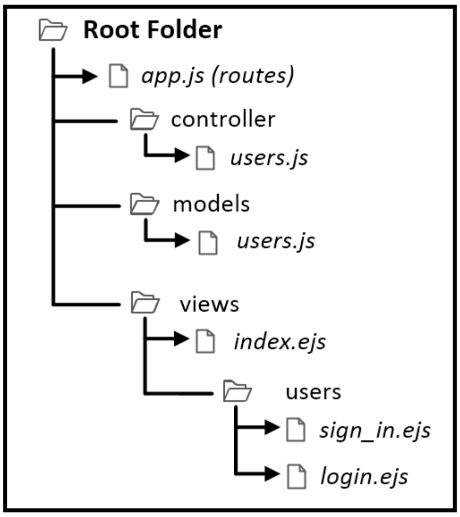


Figure 86: Extension of optimizations for the root folder

Figure 87: Sign up webpage (optimization)

7.2.2 | User experience

During the test phase, it was stated that the user experience could be improved as well. The first optimization is to add an information webpage to the *home*-webpage of the Project 2 Connect environment. This will be the *get_started*-webpage (Figure 88). On this information page, the project owner gets a clear explanation of how this side of the online application works. To make the online application easily accessible for the end-users and to ensure that the user experience will always be positive, this information page with the needed guidelines was missing, and therefore added as optimization.

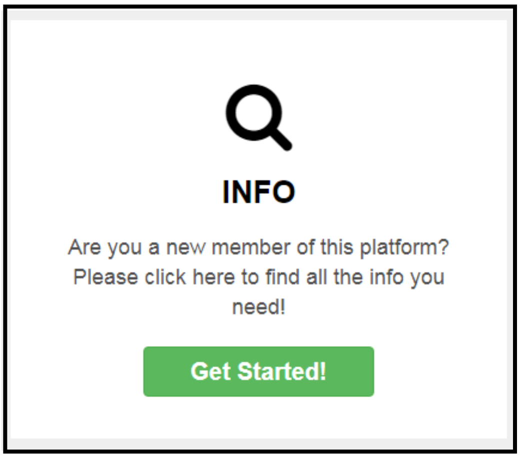


Figure 88: Get started info page (optimization)

The second optimization increases the chances for a better acquisition between the webshop customer and the project owner. The contact options for the webshop customer stopped at calling or emailing the project owner. A new webpage was developed, the *message*-webpage (Figure 89), in which the webshop customer can easily send a message to the owner of the products. He can rate the products as well. This webpage is added in the views folder, in the reuse2use folder. Now the login is created, the webshop customer can send a message and an email will directly be connected to the login of the webshop customer. Both parties get immediately in contact with each other. This research aimed to stimulate the reusable building materials. Besides the online application that is developed, will this optimization stimulate that even more. The figures are the most important aspects of the optimizations, the other webpages, extra back-end scripts, and the adjustments of the optimizations to the activity diagram are provided in Appendix IX.

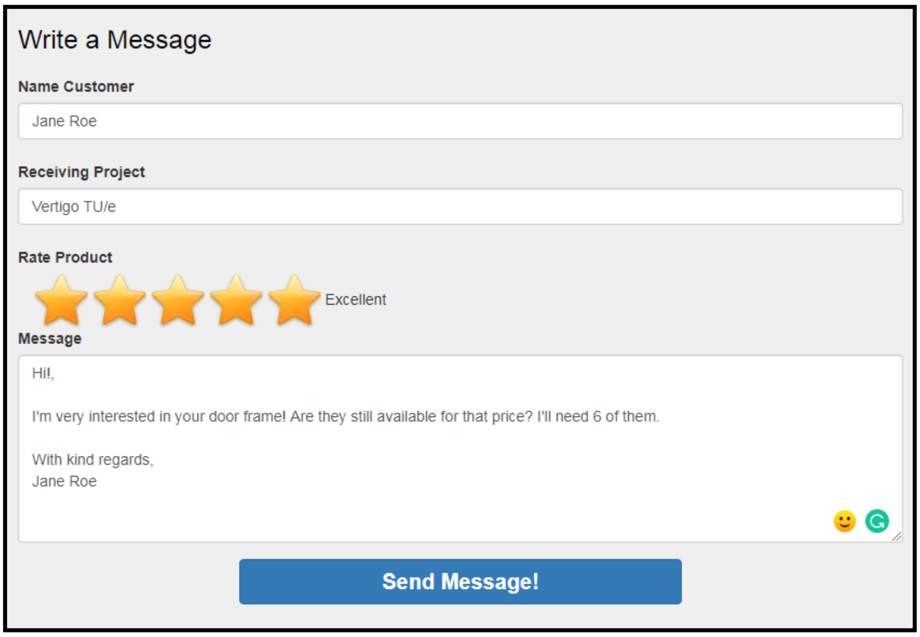


Figure 89: Possibility to send a message (optimization)

7.3 | Best-Case Scenario

Concluded can be that the result of the products in the webshop in sub-chapter (7.1.3) is very limited, based on its properties and characteristics. This was already expected since the BIM model data juxtaposition (sub-chapter 5.3) resulted in very few matching and semi-matching parameters in the BIM models and were relevant to retrieve. The results from sub-chapter (7.1.3) substantiate this brief outcome of characteristics' value, that will stimulate the reuse of building materials. Besides the name, dimensions, and a very brief description of the material properties, no more non-graphical information was included in the BIM models that would contribute to the stimulation of reusable building materials. To overcome this problem of this very brief circular-based non-graphical information input, the exchange requirements of the Re Use Index – ILS were proposed in the sub-chapter (5.2). If this guide was implemented in a project, considering these exchange requirements, the output of this same research would be completely different. To verify this statement, an evaluation is executed based on a comparison that is made between the result from the sub-chapter (7.1.3) and the desired output of non-graphical information if the Re Use Index – ILS was implemented. This will be the best-case scenario, in which every detail is described, what is desired to know if the complete composition and reuse potentials are required as input in the non-graphical information from a BIM model.

To be able to evaluate what the difference is between the retrieved data and what the output of the desired value of the best-case scenario would be if the parameters from the Re Use Index – ILS could be extracted, an example is created. This example is outlined in the same layout as it is developed in the webshop from sub-chapter (7.1.3). The example that is created fulfils all the exchange requirements that are requested in the Re Use Index – ILS (sub-chapter 5.2). The object that is used for this example is an exterior window frame that can be deconstructed from a building very soon. It has the potential to be reused and, therefore, to be applied to the online application to be out for sale.

Figure 90 shows the result of the exterior window frame in the webshop as if it was added to the online application. The non-graphical information that meets the exchange requirements of the Re Use Index – ILS that were available in the BIM model and could, therefore, be retrieved. The difference in value of the characteristics of the two compared products is evident. The non-graphical information is divided over the three exchange requirements topics: object properties (blue), object circularity indicators (green), and the object dismantle factors (yellow). A complete list of circularity-based non-graphical information is applied to the product, in which the complete (expected) life cycle of the building material is described, before and after the usage of the project of which it is part now. By being able to extract this data, the value of the online application and its webshop will obviously increase. The more we know about building materials and their complete life cycle, the more possibilities we have to support and stimulate their reuse. This does not require that much of a change, just the right guidelines during the design phase of a project and a software tool that connects the project owner and the future customer, as it was developed in this research.

Re Use 2 Use

Welcome

- Webshop
- Product Search

Categories

- Air Terminal
- Beam
- Column
- Door
- Floor
- Furniture
- Window
- Wall

Re Use 2 Use | Project 2 Connect | About Us

Product Information

Vertigo TU/e
Kennedylaan 10
Eindhoven, Netherlands
+31649532148
johndoe@tue.nl

Exterior Window Frame

Count: 1
Nl-SfB: 31.02

€21,-

Exchange Requirements	Description
Production Date	05-04-2001
Manufacturer	Bico BV., Ligusterlaan 142a, 9635 OL Amsterdam. Phone nr. 0906 – 2001832
Warranties	The validity date has expired
Components	External frame + Window
Material Types	Meranti (wood) + Hr ++ (glass)
Finishings	Paint (frame) + Coating (glass)
Finishing Color Codes	Ral 9001 (frame)
Accessories	Tilt and turn window; aluminum handle; steel joints
Height	930 mm
Length	-
Width	930 mm
Thickness	45 mm
Area m²	0,86 m²
Volume m³	0,39 m³

Exchange Requirements	Description
Used in Previous Projects	Used in 1 previous project, total time used is 17 years and 5 months
Expected Durability	From the release date: around 60 years
Depreciation	Expectation is 30% depreciation every 10 year
Average Status	Status is on average, well maintained in the previous project.
Composition of Building Material	Tropical Timber (frame) + Quartz, chalk, sand (glass)
Replaced Components	The window was replaced 10 years ago with a reused window
Destination after use	Components can both be reused, also separate from each other.

Exchange Requirements	Description
Accessible Connections	The connections are easily accessible, the window can be cut out of the framework and the frame itself can be decomposed without to much effort
Material of Connections	The connection of the frame is just pressed together within the construction of the frame and is not glued or put together screws. The window is connected to the frame with acrylic construction kit.
Type of Connections	The frame can be decomposed using a hammer. The window can be decomposed using a utility knife to cut the window loose.
Human-Scale Components	Components can be replaced manually, and this can be done in the building itself. The frame and the window are based on human-scale strength and just standard mechanical equipment is needed.
Interchangeability	The exterior window frame is a universal building material. Replacing components will not be a problem, the dimensions are standardized as well.

Figure 90: Example of best-case scenario in the webshop of Re Use 2 Use

7.4 | Re Use Index – ILS Guide

It is obvious that the implementation of the Re Use Index – ILS would be a valuable contribution towards the stimulation of the reuse potential of building materials. It is clear that, by being able to retrieve this information, the potential of reusable building materials is going to a higher level, after they get deconstructed from buildings. Even if reuse would not be possible anymore, the non-graphical information tells exactly how the building material can be decomposed. Which materials are included, which types of connection, which parts can be replaced, which parts can be recycled, etc.

The proof of concept will not be completed if the Re Use Index – ILS will not be developed as a real guide. To enable retrieving the desired non-graphical information from future projects, the Re Use Index – ILS should be implemented in the project process. This ensures that this complete list (Figure 90) will be available for every object from the BIM model. It enables us to conduct a complete project life cycle plan from the building. This would be a great opportunity for future projects to increase their circularity-based data input of non-graphical information.

But, the scope of this research was to concentrate on existing buildings and to stimulate their potentially reusable building materials. For future projects, the Re Use Index – ILS would be very beneficial, especially with its transition towards a circular built environment. But even for existing buildings, this would still be a great opportunity to implement the Re Use Index – ILS. If a building is almost at its end of life cycle, the project owner can still consider the exchange requirements and can increase the value of the objects in his or her BIM model by implementing these exchange requirements. The most important aspect of the introduction of the BIM methods in the AECO industry was the 'I' in BIM. By adding the information, this will always be valuable, even if this is at the end of the life cycle of the building. The best part about reusable building materials is that they will be reused as well as their data input. The data will be stored and will be part of the object until it reaches its end of life. Even if some exchange requirements are not known and traceable anymore, a lot of exchange requirements can still be fulfilled. Especially the values of the object properties as well as the object dismantle indicators can be captured at any time. The non-graphical information of the object will not get lost, so when an object is reused in future project, many exchange requirements are already attached to the objects. The information only needs to be changed or adjusted with the information of that project. Therefore, it does not matter at what moment of time (in the life cycle of the building material) the information will be added to the object, it will always be valuable.

This guide is developed to increase the circular-based non-graphical information in BIM models that will stimulate the reuse potentials of building materials. The complete life cycle of every building material will be described, based on the desired exchange requirements of the Re Use Index – ILS (Figure 91). It is a customized guide which is inspired on the guidelines of the basis-ILS and can, therefore, easily be implemented in the AECO industry because of its universal agreements and way of structuring data.

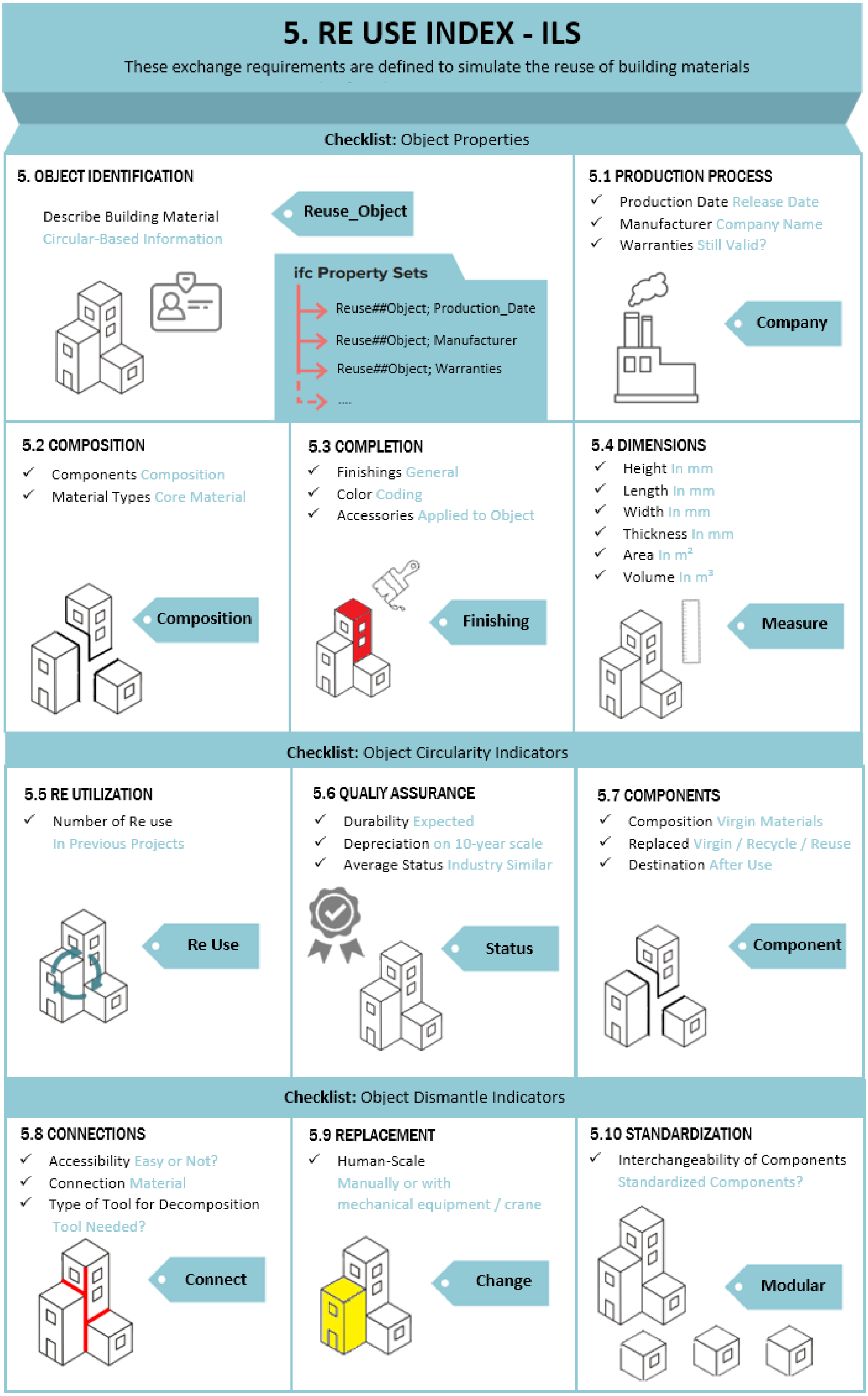


Figure 91: Re Use Index - ILS guide

7.5 | Validation

The research is conducted: the desired connection for the stimulation of reusable building materials is defined; the online application is developed; the optimizations are implemented; and the desired exchange requirements towards a circular built environment are determined. It is now time to validate the proof of concept, as it is constructed through the entire process of this research. This validation will be focused on the final connection that is developed between the non-graphical information and the cloud-based software (online application), as well as the contribution towards the circular built environment.

7.5.1 | Online Application

The development of the online application solved is in line with the scope of this research to focus on a connection between information input of the AECO industry and the cloud-based software. As is stated in sub-chapter (3.6), the AECO industry will have to extend their BIM methods with cloud-based software. Their software tools, standards, file formats are shifting towards cloud-based software, but a connection is still not developed. The most generic, scalable, and highest adoptable technology is the JSON exchange data format, which is widely adopted with the usage of Javascript programming language for web development. With these cloud-based technologies, data can easily be exchange, stored and will always be accessible.

This research tried to solve the connection with the cloud-based software issue by developing an online application using Javascript and with JSON file structure input. A possible way for retrieving non-graphical information from BIM models was proposed and developed, in which the connection with .xlsx files was created. Another option that was developed, was to export and convert data from the widely adopted software tool Solibri, and to be able to present the data in the online application. Therefore, the online application can have a contribution for the AECO industry, since it tackled the problem how to export data from BIM models into cloud-based software exchange files. The most important step that had to be implemented to overcome this problem, was to convert the BIM model data into a .json file. If this was not created, the Javascript software was not able to read the file with BIM data. Sending the data to the web browser is possible, but to be able to store the data in a database, without this converting it to .json is inevitable. This, therefore, is a point of discussion for the development of a connection between the software used in the AECO industry and the cloud-based software. Cloud-based file formats must be implemented as an export opportunity (.json is proposed), to be able to use the data from the BIM models on cloud-based platforms. If this is made possible, the data from the BIM models can much more easily be adopted and the first steps towards the cloud-based BIM platforms can be created.

The online application itself did prove the concept, as was proposed in the Methodology (Section 4, sub-chapters 4.1; 4.2). The shifting towards a cloud-based BIM platforms did increase the potential to connect the project owner with the webshop customer. A digital marketplace was invented, which fulfils the economic part of the circular economy. The reuse of building materials will fulfil the circular part of the circular economy. If we validate the potential of the online application to be launched on the market, the cloud-based software did increase this opportunity. The strength of cloud-based software is that data is easily exchangeable, data can easily be stored, and the information is always accessible. Especially this last part of the online application is the most important aspect considering the usability of the online application. Another important part of the usability is the user experience. The requirement was to create an attractive and user-friendly interface, which therefore was the focus to implement a Graphical User Interface. Since the application is only tested by the Alpha Method (sub-chapter 7.1), and only the simulated end-users tested the application, it can therefore not be concluded if the GUI of the online application meets the expectations of potential end-users based on their experience. It, therefore, is recommended to conduct a Beta Test phase, in which external feedback can be retrieved from the users that test the app.

Another important part of the online application that must be validated is the development. The online application as it is developed, is a prototype. A positive result is achieved, but it must be noted that it only works within the conditions that are mentioned in sub-chapter (6.3). Before a possible launch on the market, many optimizations are needed. The optimizations that are considered in sub-chapter (7.2) are only focused on the user experience. The most important optimizations that need to be adjusted to the back-end of the online application will now be discussed. (1) The model is hardcoded, which means that the parameters that will be retrieved from the .xlsx project report files, have to be identical to the parameters in the model. If they do not match, the non-graphical information cannot be retrieved, and the online application will read them as 'undefined'. This does also mean that no more parameters can be added to the .xlsx project reports, since the model is not able to read them. (2) The converter webpage is still a very complex web page, on which the end-user still needs to execute many steps. An optimization needs to take place in which this all is executed in the back-end. The end-user only needs to upload his or her project and it will immediately be saved as a .json file. (3) The products can be added to the online application manually and only one by one. This will take a lot of time if an end-user wants to add a complete project to the webshop. Therefore, it is important to extend this function into an easier way of adding products to the online application. The online application is a prototype, which is in its early stage. Many optimizations need to be added to the back-end before it can proceed to the next phase in its development process.

The extensibility of the online application is also shown. This first draft is a prototype version of how the application works and it shows that it is possible to store multiple projects in the database. If the application will be launched, the server must be able to handle an enormous number of data input once many end-users will access the online application. The decision to choose for a NoSQL database (sub-chapter 4.4.3.2) is, therefore, an excellent opportunity to extend the servers. Where SQL reaches a limit, the NoSQL database can easily be expanded horizontally as well as vertically. A point of discussion is of course if the back-end of the online application can handle huge numbers of data input as well. Since this is not tested on a large scale, this cannot yet be concluded. This enhancement needs to be part of an optimization phase after this research.

Another important aspect, which is not implemented in the online application on a large scale, and is mandatory if the application will be released, is the privacy aspect. The optimization of the sign-up and login features enable the project owners to work within their environment and no other project owner can access their data. But the responses and requests that are handled by the online application are done with HTTP and not HTTPS. The Hypertext Transfer Protocol handles the request, by sending it the server, the server sends back the response, and the end-users get the web page they asked for (sub-chapter 6.3.1). The HTTPS does the same thing, only includes the 'S' in the protocol, which stands for 'Secure'. The HTTPS handles the same requests and responds but secures this connection. It encrypts the data input in the web browser, such as passwords, email addressed, bank accounts, etc. Validating the developed online application will recommend to secure this connection between the server and the web browser by using HTTPS.

The online application has the potential to be launched on the market and to be used in a real-life environment. It did prove that the concept works, that the project owner can reach the webshop customer by uploading data and that the webshop customer can reach the project owner, because the project and product information are presented in a webshop. The online application is a prototype version, in which the Alpha Testing is already executed, and the optimization is implemented. The full application can run without errors or failures and has the potential to be scaled. The final success of the online application can only be concluded if more tests are conducted, especially based on external user-experience and uploading high numbers of data.

7.5.2 | Circular built environment

The most important part of the validation of the final results, is the contribution these results have for the AECO industry, and most importantly, for the transition towards a circular built environment. This research aimed to stimulate the reuse of building materials, since it is the feedback loop within the circular built environment with the highest potentials and has the least negative impact on the environment (sub-chapter 2.3). This aims to reduce the waste production of the construction sector by closing the cycle loops, without the need for input of new components or materials, which are extracted from natural resources. The reuse possibilities should especially be focused on the existing buildings since the circular concepts are mainly focused on future projects (sub-chapters 2.6; 2.7; 2.8). Existing building materials have the higher potential to be reused, since these buildings will be deconstructed sooner. But information input is required to be able to stimulate the reuse potentials of these building materials.

The BIM methods did increase this information input in the AECO industry. Many standards (sub-chapter 3.3) were developed to enable a higher outcome of information in the BIM model, to increase the maturity from 3D models into BIM models. This provided insight in the information sources of a BIM model, in which the LOI (sub-chapter 3.3.2) was most valuable to elaborate on in this research. It is desired to increase the LOI in the collaborative processes within the AECO industry, to retrieve a higher output of non-graphical information, and to get insights in the full project's life cycle (sub-chapter 3.8). But, to be able to increase the LOI, the collaboration aspect of a project team is very important. A BIM project collaboration can only achieve a high LOI if it sets the right requirements and agreements based upon the desired information output. The standards that are developed try to overcome this challenge because the AECO industry is still struggling with how to work with universal and structured agreements. A possible solution to tackle this challenge is the basis-ILS on which was elaborated on during the execution of this research.

The validated results from the analyses that were conducted in Section 5 concluded that the difference between the desired non-graphical information in a circular built environment and the matching data in the contemporary BIM models, is very big. Almost none of the desired exchange requirements (sub-chapter 5.2) could be retrieved from the BIM models (sub-chapter 5.3. Validating this outcome, it must be noted that it is positive that not much of the desired data was matching the data from the BIM models. The desired exchange requirements were not part of the process during the development of those projects. You need to set requirements to obtain certain information. And this was not the case for the used BIM models. Therefore, the Re Use Index – ILS was developed, to set circular-based requirements and provide guidelines for projects in which the stakeholders want to retrieve the desired non-graphical information from the BIM model. But, besides the fact that this information was not available, the potentially reusable building materials are still be stimulated with the use of the online application. It realized an easier connection between the project owner and the webshop customer, even if the non-graphical information input from BIM models is not circular-based. And therefore, the main goal of this research is still accomplished.

But the Re Use Index – ILS needs to get a follow-up study that will make sure that is the circular-based exchange requirements are complete, need adjustments, or need to be changed or deleted. The exchange requirements are based on several resources, but it is never investigated if this is requested within circular-based project teams as well. If a study is conducted to verify the completeness and strength of the Re Use Index – ILS, this guide can be a contribution to the AECO industry in its transition towards the circular built environment. Because it is inspired by the basis-ILS, which is based on universal guidelines and agreements, set by the AECO industry itself, the Re Use Index – ILS can easily be implemented. The final contribution of this research for the market, the AECO industry, and to support the circular built environment, especially by concentrating on the stimulation of reusable building materials, is, therefore, the Re Use Index – ILS guide. This guide has great potentials in achieving the vision of the Dutch government to have a full circular economy with only reuse cycle loops by 2050.

Section 8

Conclusion & Discussion

8 | Conclusion & Discussion

This section shall finalize the thesis. The final conclusion is drafted (sub-chapter 8.1), based on formulating the answer on the main question that was stated for this research. In the next sub-chapter (8.2) the reflection on the research sub-questions are described as well as the scientific and societal relevance of this research. The thesis report will end with a critical discussion as well as defining recommendations for future research which are described in the last sub-chapter (8.3).

8.1 | Conclusion

The answer on the main question is formulated and will be the final conclusion of this research:

"How can the connection between non-graphical BIM data of existing buildings and cloud-based software be realized, in order to stimulate the reuse of building materials to support the circular built environment?

The answer to this question resulted in two parts: desired non-graphical circular-based information output and the development of an online application that realized the connection.

Circular-based non-graphical information is necessary if building material life cycle information is desired to be extracted from BIM models. To enable stakeholders from projects to retrieve the desired information, circular-based exchange requirements must be listed, that will provide guidelines for the stakeholders to implement the requested circular-based information. If these exchange requirements are not implemented in the project process, the desired information cannot be obtained in a later phase of the project. The BIM model data juxtaposition did conclude that almost none of the desired circular-based exchange requirements is implemented in contemporary BIM models. This research, therefore, developed the Re Use Index – ILS, which is inspired by the basis-ILS, to provide guidelines for stakeholders in projects, based on circular-based requirements. These requirements can be implemented in future projects, but they can also be added to BIM models of existing buildings that do not contain these requirements but do have potentially reusable building materials available. The Re Use Index – ILS aims to increase the value of circular-based non-graphical information in existing buildings, to increase the stimulation of the reuse of its building materials. The stimulation of reusable building materials is in line with the vision of the Dutch government, to have a full circular economy, with only using reuse processes by 2050. The Re Use Index – ILS can be implemented as a guide in the AECO industry, which will contribute to its shifting towards a circular built environment.

The second part of this research that will stimulate the reuse of building materials, is based on the shifting towards cloud-based BIM platforms in the AECO industry. This aims to connect the non-graphical information from the BIM models to the cloud-based software technologies. The value of cloud-based software as well as the unique selling points and opportunities that were concluded from the literature review, resulted in an online application that is developed. This online application is a digital marketplace, developed with cloud-based software, for the supply and demand of reusable building materials. The acquisition of building materials fulfils the economic part, and the stimulation of the reuse of it, the circular part of the circular economy. Although the retrieved non-graphical information from the BIM models was not circular-based as expected, the online application did show that the software that is used in the AECO industry can be connected to cloud-based software, and that this connection does increase the reuse potentials of the building materials. In this research, it is concluded that the JSON exchange data format in combination with Javascript for web development, has the highest value and potential for realizing the connection between the software used in the construction sector and the cloud-based software. The online application can convert .xlsx files into .json, present the data in the web browser, and sell the objects from the BIM models as products in a webshop. The online application is a real-time prototype, which has positive test results. With the optimization of several hardcoded scripts in the back-end, it has realistic potentials to be used in the AECO industry.

8.2 | Research Questions & Relevance

In this sub-section, the research sub-questions, that are formulated in the first section of this research will be reflected. This sub-section also describes the relevance of this research based on scientific and societal contributions.

8.2.1 | Reflection on sub-questions

What is a circular built environment and how does this affect the AECO industry?

The circular built environment, different to the linear construction sector, tries to close their feedback cycle loops, instead of the 'take-make-waste' disposal that defines the linear construction sector. Especially the four feedback cycle loops: maintain, reuse, remanufacture, and recycle are fundamental in the circular built environment. The construction sector tries to implement the circularity approach during its complete life cycle as well as the primary and secondary processes. In every phase, changes can be implemented that create a more circular construction project. These changes can be: using building materials with a high life expectancy, using recycled materials, rather maintain than replace, and finally the most important one, reduce the waste production by recycling waste or reuse the building materials in future projects. But the circularity approach that affects the AECO industry are the initiatives for future projects, specially developed for a full circular built environment. These initiatives are mainly implemented in the design phase and change the entire traditional output from a building in being a completely deconstructable building. The highest values of the building materials can be captured if they can easily be dismantled from the building, in which they are immediately ready to be reused. The whole scope around the implementation of the circular built environment for the AECO industry is to think about what can be changed in the traditional processes that could be more sustainable. Buildings must be seen as material banks and constructing future projects must be done by reusing building materials that are extracted from existing buildings.

How is the building data managed within a BIM model and which standards increase the collaboration and interoperability potentials?

The way the data of a building is managed within the AECO industry is depending on the maturity level of BIM that is used for a project. The LOD (level of development) and LOI (level of information) indicate how much graphical and non-graphical information is included in the model. If this is done, it provides opportunities to do cost calculations, extract quantities and even retrieve a full life cycle report. The quality level of the graphical information and the non-graphical information must be pre-defined during a collaboration project to retrieve certain levels of data. These levels are also defined in several standards that are developed for the AECO industry to use as fundamental agreements and requirements in the project. The ISO and the CEN/TC committees provide many standards that can help project teams in managing their building data. But especially the openBIM standards increase the collaboration and interoperability potentials in the AECO industry. Neutral exchange file formats (e.g. IFC) made it possible to exchange data between stakeholders. With BIM-based model checking software, quality checks, clash detections, and data reviewing became possible. This increased the collaboration quality as well, since communicating is possible within the BMC software tools. Besides the standards, are guidelines proposed in the AECO industry which enable creating a more structured collaboration process as well as the transition towards a circular built environment. These are the guidelines of the basis-ILS. Customizations on these guidelines are possible, in which a project can ask for specific exchange requirements, based on their project goals.

Which matching building materials characteristics can be retrieved from the BIM models, which will increase the reusability potentials of the building materials?

Based on the BIM model data juxtaposition that is conducted in this research, very few of the desired circular-based building materials characteristics could be retrieved from the contemporary BIM models. It can be concluded that if no proper exchange requirements of that purpose are set, the desired information cannot be obtained from the BIM model. The result of this analysis was a brief list of parameters that were mainly focused on the dimensions of the objects and a few general characteristics, none were circular-based. But since that was probably not required when these projects started, this can be an obvious result. It can be determined that if this circular-based non-graphical information is desired to be obtained from a BIM model, it is very important to pre-define the exchange requirements. The expected circular-based outcome is only possible if the participants of the project do include these requirements to the objects in the BIM model. To be able to retrieve this data in projects, the Re Use Index – ILS was developed, to provide guidelines for project teams, based on circular-based exchange requirements. This to be implemented in the BIM collaboration processes, that can increase the reusability potentials of the building materials. The input for this guide was based on: object properties, object circularity indicators, and object dismantle indicators, which all contain a list of specific circular-based exchange requirements.

What requirements must be met, when creating the possible transfer between the non-graphical BIM data and the cloud-based software?

The transfer from the non-graphical information of the BIM models to the cloud-based software. It was an issue that had to be solved. The AECO industry has a variety of software tools that are used to design, check models, calculated, communicate, etc. is enormous. And these software tools all need to be able to exchange and retrieve data from the other software tools. This problem is tackled in the BIM process using neutral file formats like IFC. With the transition towards the cloud-based software. The JSON exchange file format is determined as the highest potential in realizing this connection. JSON is a key-value style lightweight data exchange format that has higher parsing efficiency than XML and due to the inadequacies of XML, JSON has been widely used in web applications, specifically in Javascript web services (Afsari, et al., 2017) A key problem is that the .json file export option is not (yet) implemented in widely adopted software tools like Solibri, which is used in this research as BIM-based model checker. The only export file that was available to retrieve the non-graphical data was a .xlsx (Excel) file. The Javascript software that was used to develop the back-end of the application was not built to parse this file input but was able to read the neutral format .json file. Therefore, a converter is developed which transformed the .xlsx file with the project data into the .json file. The data input was maintained, only the file structure changed. A script was developed in the back-end that was able to transform the files. In the online application, a webpage was developed, on which the project report .xlsx file can be uploaded, converted, and finally be downloaded as .json file. The project data was still preserved. It therefore became possible to transfer the non-graphical BIM data to the cloud-based software.

What pattern can be designed that connects the cloud-based software with a user-friendly front-end?

A pattern was designed that was able to connect the back-end with the front-end, the API with the database, the Node.js server with the web browser, and was able to create user-friendly front-end as well. This was a complex pattern in which different methods and frameworks were put together into one system design. The MVC framework was fundamental in the system design. This framework is very often used for web development processes and can connect the web browser with an external database. The *routes* receive the HTTP request and send it to the *controller* by directing the *controller* into the right path. If data must be stored or fetched from the external database, the *controller* sends this request to the *model* which is connected to the external database. For this research, the NoSQL MongoDB software tool was used, since it was best applicable. The *model* stores or fetches data from MongoDB and sends it back to the *controller*. This data must be rendered in which the *controller* sends the data to the *view*. This part of the MVC pattern is responsible for the user-friendly interface of the front-end. The style sheeting of the online application is also developed and added to the *view.* The *view* sends back this presentation to the *controller*, which on his turn sends the response back to the end-user in the web browser. This MVC pattern was developed in Node.js which runs on Javascript but has extra features available. The root file in Node.js connects the back-end with the web browser in the front-end. With this pattern, it was possible to connect the different parts from the research, and therefore create one complete system design.

How to design a system that involves different stakeholders and end-users?

The different stakeholders involved in this project could be distinguished as the project owner and the webshop customer. For both sides, the key group was determined as: contractors, investors, real estate owners, clients, housing corporations, freelancers, and private house owners. The project owner fulfills the supply side of the digital marketplace (online application) because he or she owns a BIM model of a project that soon will be deconstructed, and the project owner, therefore, wants to add the potentially reusable building materials to the online application. The other stakeholder is the webshop customer, who fulfills the demand side. These customers want to buy reusable building materials to use them in future projects. Both sides have to be connected but need a different environment in the online application. One online application was developed, on which both sides were brought together. This was decided, since the input of the project owners, is the output for the webshop customers. With the connection to one database, the same data could be used, but for different purposes. To create a clear distinction in the online application for both sides, a different environment was developed. The project owners used the Project 2 Connect side, and the webshop customers used the Re Use 2 Use side.

Which circular-based approaches for existing buildings, conducted from the literature review, can be applied to the online application?

The conducted literature review about the circular built environment did provide many insights for this research. In the Methodology, several decisions were made that finally resulted in the development of an online application. That could be the possible connection that was asked in the main question of this research. This was based on unique selling points over several circular built environment initiatives. The fundamental aspect of the online application was based on the USPs on the 'building as materials bank' approach and the distinction between the Madaster platform with the development of the digital marketplace. A reusable label was developed in the online application, that enables the project owner to assess the reuse potential of the building materials that could be extracted from the building. This is done based on Building Circularity Indicators. These BCIs score the building and its materials based on the value that can or cannot be captured once the building materials would be disassembled. Other circular-based literature resources are included in the guidelines of the Re Use Index – ILS. This was not applied to the online application but does have a high value in the final output of this research.

Can a proof of concept be developed that meets the requirements of the stakeholders and end-users, and/or does it need optimizations after it is validated?

The concept that was proposed in the methodology has been developed and tested. With the positive test result, it can be stated that the proof of concept is achieved. For the system design, a list of requirements was determined, based on the needs of the stakeholders (project owners and the webshop customers). When developing a product or application, it is advised to consider the needs of the end-users, to lower the threshold when the application will be launched on the market. If the requirements of the end-users are met, the usability of the application will probably receive more support. The prototype was developed which can run in the web browser without failures or bugs. It can upload, store, and retrieve data from and to the database as well as presenting this same data as products in a webshop. The prototype was also tested using the alpha testing method, based on a real-user environment. During the execution of this test phase, a few points were defined that need to be optimized, based on the needs of the end-users. In the validation of the online application, more optimizations were discussed based on the development of the back-end. Within the conditions of this research, the online application has a positive result. But the prototype in an early stage. Before a possible launch on the market, it is advised to implement the stated optimizations to increase the value and opportunities of the online applications.

8.2.2 | Scientific Relevance

The scientific relevance of this research is to create awareness of what possibilities the JSON exchange data format has in combination with Javascript for the AECO industry. In the shifting towards cloud-based software technologies, has especially the JSON file format the high potential to create the connection between the software tools that are used nowadays, and towards the desired cloud-based BIM platforms. JSON has a high potential to easily exchange BIM data, because of its lightweight data exchange and high parsing possibilities. The JSON file format is generic, scalable, and highly adoptable by a variety of software tools. The development of the online application did prove that it is possible to export BIM data into cloud-based software and be able to work with this data in a web development environment using the Javascript program language. By retrieving non-graphical information from the BIM models, being able to present it in a web browser, and therefore, having this data always accessible, the value of a cloud-based BIM platform is substantiated. The opportunities that JSON has in this shifting can, therefore, be relevant for future researchers to elaborate on. The development of the system is described step by step, which makes it possible to adopt certain parts of the script and extend it for other research purposes.

8.2.3 | Societal Relevance

The societal relevance of this research is the contribution that this research has for the transition towards a circular built environment and therefore can help in the reduction of the waste generation. The stimulation of the reuse of building materials is in line with the vision of the Dutch government, to have a fully circular economy by 2050. Which makes this research very topical, since the government only wants to use reuse processes. This research provides two solutions for the AECO industry to stimulate the reuse of building materials. The Re Use Index – ILS can be implemented in their processes which will increase the circular-based non-graphical information in the BIM models. The desired exchange requirements will provide a complete list of the requirements that are needed in order to know if a building material has reuse potentials. Once this non-graphical information is attached to the objects, it can be retrieved from the BIM models and added to the online application. This enables the project owners to sell their reusable building materials in which they are easily connected to the future customers of the building materials. By increasing the circular-based non-graphical information of the objects in the BIM models and the ease of selling them, the reuse of building materials will be stimulated. Which is the main objective of this research.

8.3 | Critical Discussion & Recommendations

The conclusions from the thesis are drafted and the results are defined. The conducted research has points of discussion that will critically be described as well as improvements for the conducted research. These improvements will be substantiated as recommendations for future research. The point of discussion and recommendations are divided over three topics: general, online application, and the Re Use Index – ILS.

8.3.1 | General

The first point of discussion is the lack of circular-based non-graphical information that could not be obtained from the BIM model data juxtaposition. It was hoped to retrieve more matching data, especially with circular-based characteristics attached to the object. Unfortunately, this was not possible, and almost none of the listed exchange requirements of the Re Use Index – ILS were implemented as well. But this can be discussed as being a positive point. Since the information was not provided in the BIM models, no extra information, which was probably not pre-defined as exchange requirements during the design of the BIM models, was implemented. This is actually positive, because the exchange requirements and agreements of a project, define what information must be included in the BIM models and what information must not. To be sure that circular-based information can be extracted if this same analysis is conducted in future research, it is recommended to focus on the exchange requirements of that particular project. If circular-based information was desired to be retrieved from the BIM model, it probably is included

As an output of the BIM model data juxtaposition and during the development of the online application, Solibri was considered as an important path. The real focus was the connection between the .xlsx file to the .json file. Solibri was provided by the supervising company, and therefore an easy software tool to apply to this research. The research aimed to connect the BIM model data to the cloud-based software. Since Solibri is a commonly used software tool in the AECO industry, it was an obvious decision to investigate the possibilities with this software tool and the cloud-based software as well. But the final input was still an .xlsx file, which is universal and not a unique export option file from Solibri. Therefore, it must be noted that Solibri did not was the complete focus of this research, but it was used based on its popularity and availability. The most important part is the connection with the .json file, which should be elaborated on in future research. An opportunity for future research would be, if Solibri is considered to be used as well, to investigate if a direct export to a .json file would be possible.

8.3.2 | Online Application

Based on the development of the back-end of the online application, several optimizations should take place to get the application from being a prototype, to a more mature level. The most important optimizations are: (1) The model is hardcoded. The data that will be uploaded to the online application, must be identical to the codes in the back-end of the model. If this data is not identical, the online application cannot read the data. This means that the project owner is not able to make changes to the ITO Report as well as the Project Report Template, without creating failures in the back-end. (2) The convert option must completely be executed in the back-end. The end-user needs to execute multiple steps before his or her report is converted and uploaded. This all should be done in the back-end. (3) The only possibility to add the products now is one by one. This should be changed as well. The project owner should be able to add his or her entire project as single products to the webshop in once. These are the three most important optimizations in the development of the back-end which should be adjusted. If the online application will be able to be launched on the market, the adjustments should definitely take place.

The online application is a prototype version, which has a positive test result. Within the right conditions, that are used in this research, the online application can run without failure. But only the alpha testing method was executed: during the development based on the findings of the developer, and in the test phase based on the simulated real-user environment. If the online application will be taken to a more mature level, it is recommended to execute a beta testing, based on the experience of real end-users. Their findings are critical in a possible successful launch on the market.

Since the online application is a prototype, it should also be tested in a secured environment if the online application can handle a possible launch on the market. The online application should be able to handle large amounts of data input. Because many projects will be uploaded. And if many projects are uploaded, the security aspect is even more important as well. Because the project owners need to be able to work in a secure environment. It, therefore, is recommended to execute these two tests, to be sure it can handle large amounts of requests and can work in a secured environment as well.

The online application has a lot of potential to be launched on the market and to actually have value for the AECO industry in the stimulation of reusable building materials. But it is stated that first many tests and optimizations need to take place. If this is done, a marketing campaign should be launched to create awareness for the online application, especially for projects that will be deconstructed soon. This will enable project owners to already upload their reusable building materials even before the building is deconstructed. The potential buyers on their term, have the chance to already see which reusable building materials become available soon. They therefore can design a project, based on the availability of reusable building materials nearby.

8.3.3 | Re Use Index - ILS

The Re Use Index – ILS guide is inspired by the guidelines of the basis-ILS. The exchange requirements used as input for the Re Use Index – ILS are conducted from literature resources. The layout of the guide is identical to the basis-ILS which will provide an easier implementation in the AECO industry since the basis-ILS is developed by and for the AECO industry. But it is never investigated if the Re Use Index – ILS does have this same effect on the AECO industry and that it meets the requirements of the construction sector. Therefore, further research is needed on several points: (1) It must be investigated of the exchange requirements that are listed, are complete, based on the circular-based information that will stimulate the reuse of the building materials. If research will be conducted it is wise to let experts discuss if the exchange requirements are complete, need to change, or maybe have to be excluded. (2) It must be proven that the Re Use Index – ILS guide has the same effect as the basis-ILS on the market. This to be sure that the exchange requirements are clearly described, and the layout of the guide does not contain mis interpretable visualizations. (3) Some exchange requirements ask for a larger description of information input. It must be determined if it is useful to ask for this information, or if they might not be specific enough. It, therefore, is wise to test the information input among modelers, and to determine if they are willing to add larger amounts of information, or if more specific exchange requirements are required. It is clear that besides the online application, the Re Use Index – ILS is a prototype version as well. This guide also needs to be tested on different topics, before it can be launched on the market.

If the online application is combined with the Re Use Index – ILS, as was done in the best-case scenario (sub-chapter 7.3), a research must be conducted if every single exchange requirement of the Re Use Index – ILS should be extracted from a BIM model to be presented in the webshop. Because not every exchange requirement will increase the attractiveness of the product. The exchange requirements must be attached to the objects in the BIM model, but they might not all be valuable for the online application. It, therefore, is recommended to test this among the end-users, which non-graphical information they desire to get presented and which not.

In the first part of the literature review (Section 2), the Madaster platform was discussed. The platform does collect circular-based information of the building, this is done with material passports. A point of discussion is that some requirements of the material passport are similar to the requirements of the Re Use Index – ILS. This is because they both try to retrieve information about the building materials, based on its core components and virgin materials. But the materials passports as well as the Madaster platform are focussed on new building projects. The distinction with the output of the exchange requirements of the Re Use Index – ILS as well as this research aims to focus on existing buildings and therefore, they both have a different information input.

If the Re Use Index – ILS will be implemented in the AECO industry, agreements must be made in which part of the process the three topics: object properties, object circularity indicators, and object dismantle indicators, will be attached to the objects and by whom. A recommendation would be to include an information-manager in the project. This manager will be responsible for meeting these requirements, as well as other project life cycle information, if this is desired. The information-manager will verify if the exchange requirements of the Re Use Index – ILS are met, as well as other exchange requirements. It is important to have that type of control during the full execution of the project. This to ensure that the desired LOI is accomplished in the project.

A final recommendation that is advised if future research will be conducted on the Re Use Index – ILS, is to compare the exchange requirements with the entities as they are determined in the latest official version of the IFC 4.1. The BIM model data juxtaposition did conclude that almost none of the exchange requirements cloud be retrieved from the contemporary BIM models. After the entities in IFC are investigated and if comparisons are found, it can be noted that it already should be possible to include these types of circular-based information in the contemporary BIM models. A point of discussion then will be, why this information was not attached to the objects in the first place. The development of the specifications in IFC and its shifting towards a circular built environment was not taken into consideration in this research but can, therefore, be a recommendation as a topic for future research.

< Page intentionally left blank >

9 | References

Abd, A.M., Khamees, A.S. (2017). As built case studies for BIM as conflicts detection and documentation tool. *Cogent Engineering Journal, 4.* P. 1-11.

Afsari, K., Eastman, C.M., Castro-Lacouture, D. (2017). JavaScript Object Notation (JSON) data serialization of IFC schema in web-based BIM data exchange. *Automation in Construction, 77.* P. 24-51

Aguiar, A., Vonk, R., Kamp, F. (2019). BIM and Circular Design. *IOP Conference Series: Earth and Environmental Science, 225.* P. 1-8.

Akanbi, L.A., Oyedele, L.O., Akinade, O.O., Ajayi, A.O., Davila Delgado, M., Bilal, M., Bello, S.A. (2018). Salvaging building materials in a circular economy: A BIM-based whole-life performance estimator. *Resources, Conservation and Recycling, 128.* P. 175-186.

Arayici, Y., Fernando, T., Munoz, V., Bassanino, M. (2018). Interoperability specification development for integrated BIM use in performance based design. *Automation in Construction, 85*. P. 167-181

Ayanoglu, E., Aytas, Y. (2016). Mastering RabbitMQ. *Packt Publising Limited, 1*. P. 286

Azhar, S., Asce, A. (2011). Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AECO industry. *Leadership and Management in Engineering.* P. 241-252

BAMB. (2019). Reversible Building Design. Retrieved from: https://www.bamb2020.eu/topics/reversible-building-design/

BAMB. (2020). Buildings As Material Banks. *Enabling a circular building industry*. Retrieved from: https://www.bamb2020.eu/

Baldwin, M. (2019). The BIM-Manager: A Practical Guide for BIM Project Management. *Mensch und Machine Deutschland GmbH*

Beurskens, P. R., & Bakx, M. J. M. (2015). Built-to-rebuild*. Eindhoven University of Technology.*

Bew, M., Richards, M. (2008). BIM maturity levels model.

Bicarri, C.D., Abualdenien, J., Borrmann, A., Corallo, A. (2019). A BIM-Based Framework to Visually Evaluate Circularity and Life Cycle Costs of buildings. *Earth and Environmental Science, 290*. P. 1-9

BIM & ICT. (2016). Common Data Environment (CDE). *BIM portal*. Retrieved from: https://www.bimportal.be/nl/lexicon/cde/

BIM Loket. (2017). BIM basis-ILS. *Een basis om op te bouwen*. Retrieved from: https://www.bimloket.nl/p/112/bimbasisils

Brand, S. (1994). How Buildings Learn; What happens after they're built. *Penguin Publishing Group.*

BSI. (2019). Introductie van de nieuwe internationale norm voor BIM, ISO 19650. *Het integreren van digitale innovatie en het versnellen van wereldwijde toepassing.* P. 1-7

Braungart, M., McDonough, M.W., Bollinger, A. (2007). Cradle-to-cradle design: Creating Healty Emissions – a strategy for eco-effective product and system design. *Journal of Cleaner Production, 15*. P. 1337-1348

Bui, N., Merschbrock, C., Munkvold, B.E. (2016). A Review of Building Information Modelling for Construction in Developing Countries. *Procedia Engineering, 164.* P. 487-494

buildingSMART International. (2019). Model View Definitions (MVD). Retrieved from: https://www.buildingsmart.org/standards/bsi-standards/model-view-definitions-mvd/#:~:text=An%20IFC%20View%20Definition%2C%20or,IDM%20(also%20ISO%2029481).

buildingSMART International. (2020). openBIM. Retrieved from: https://www.buildingsmart.org/about/openbim/

buildingSMART International. (2020). Technical Roadmap buildingSMART. *Getting ready for the future*. P. 1-33

CAD & Company. (2019). BIM Levels. *BIM Maturity Model.* Retrieved from: https://www.cadcompany.nl/kennisbank/bim/bim-levels/

CEN/TC 442. (2019). Building Information Modelling (BIM). *Business Plan*. P.1-19

Churcher, D., Davidson, S., Kemp, A. (2019). Information Management according to BS EN ISO 19650. *Guidance Part 1: Concepts*. P. 1-42

Clemen, C. (2017). Collaboration – Standards for BIM information exchange and process management. *Fig working week, BIM for survivors*. P. 1-27

Cobouw, Maas, M. (2017). Bouwpartijen nemen regie in BIM-afspraken. Retrieved from: https://www.cobouw.nl/bouwbreed/artikel/2017/05/bouwpartijen-nemen-regie-in-bim-afspraken-101248055

Deller, K., Price, K., Webster, M., Kahley, E., Hosey, Lance., Bennink, Dave. (2019). Design for Disassembly in the construction sector. *A guide to closed-loop design and building*. P. 1-69

Dimyadi, J., Hjelseth, E., Lassen, A.K. (2016). Development of BIM-based model checking solutions – ongoing research and practitioners' demand. *Computer science.* P. 1-10

Duurzaam Gebouwd. (2018). Uniforme ILS constructieve betonwanden beschikbaar. *Het intergrale platform*. Retrieved from: https://www.duurzaamgebouwd.nl/artikel/20180711-uniforme-ils-constructieve-betonwanden-beschikbaar

Evolve Consultancy. (2018). LOD = LOD + LOI. Retrieved from: https://evolve-consultancy.com/lod-lod-loi/

Ellen MacArthur Foundation. (2012). Circular Design. *Gestaltung der Kreislaufwirtschaft.* Retrieved from: http://www.circular-design.eu/automatisch/

Ellen MacArthur Foundation. (2015). Towards a circular economy: Business rationale for an accelerated transition. P. 1-20

Ellen MacArthur Foundation. (2015). Circularity Indicators. *An approach to measuring circularity*, *project overview.* P. 1-12

Ellen MacArthur, Zumwinkel K., Stuchtey M.R. (2015b). Growth within: A circular economy vision for a competitive Europe. *Ellen MacArthur Foundation.* P. 1-98.

EMF (Ellen MacArthur Foundation). (2012). Towards the circular economy vol. 1: An economic and business rationale for an accelerated transition. Retrieved from: www.ellenmacarthurfoundation.org

Ellen MacArthur Foundation. (2016). Circularity In The Construction sector: Case Studies. *A compilation of case studies from the CE100.* P. 1-71

Fabian, B., Baumann, A., Lackner, J. (2015). Topological analysis of cloud service connectivity. *Computers & Industrial Engineering, 88.* P. 151-165

Fifield, B., Medkova, K., (2016). Circular Design – Design for Circular Economy. P. 1-16

Gafaar, A. (2017). SQL vs NoSQL. *Computer Science Department, 3*. P. 1-4

Gao, X., Pishdad-Bozorgi, P. (2019). BIM-enabled facilities operation and maintenance: A review. *Advanced Engineering Informatics, 39*. P. 227-247.

Geyer. R., Jackson. T. (2004). Supply Loops and Their Constraints. *California Management Review.*

Gigante-Barrera, Á., Dindar, S., Kaewunruen, S., Ruikar, D. (2017). LOD BIM Element specification for Railway Turnout Systems Risk Mitigation using the Information Delivery Manual. *Materials Science and Engineering, 245*. P. 1-11

Graham, P. (2002). Building Ecology: First Principles For A Sustainable Construction sector.

Grierson, D. (2009). Towards a Sustainable Construction sector. *University of Strathclyde Glasgow*. P. 70-77

Grytting, I., Svalestuen, F., Lohne, J., Sommerseth, H., Augdal, S., Laedre, O. (2017). Use of LoD Decision Plan in BIM-projects. *Procedia Engineering, 196*. P. 407-414

Hanfield. D. (Producer), Hancock, J.L. (Director). (2016). *The Founder* [FilmNation Entertainement]. United States. Quote: Ray Kroc (Michael Keaton).

Helendi, A. (2018). What is Event-Driven Programming And Why Is It So Popular? *Cloud Zone*. Retrieved from: https://dzone.com/articles/what-is-event-driven-programming-and-why-is-it-so

Hjelseth, E. (2015a). BIM-based Model Checking (BMC). *Building Information Modeling.* P. 33-61

Hjelseth, E. (2015b). Foundations for BIM-Based model checking systems. *Department of Mathematical Sciences and Technology*. PhD Thesis

Het Groene Brein. (2019). Kenniskaarten. *Hoe circuleren materialen in een circulaire economie?* Retrieved from: https://kenniskaarten.hetgroenebrein.nl/kenniskaart-circulaire-economie/circulatie-materialen-circulaire-economie/

Hijazi, A.A., Omar, H.A. (2017). Level of Detail Specification Standards and File-Format Challenges In Infrastructure Projects for BIM Level Three. *Building Information Modelling (BIM) in Design, Construction and Operations II.* P. 44-62

Hossain, M.A., Yeoh, J.K.W. (2018). BIM for Existing Buildings: Potential Opportunities and Barriers. *IOP Conference Series: Materials Science and Engineering, 371.* P. 1-9.

Iacovidou, E., Purnell, P. (2016). Mining the physical infrastructure: Opportunities, barriers and interventions in promoting structural components reuse. *Science of the Total Environment, 557.* P. 791–807.

ISO. (1946). International Organization for Standardization. Retrieved from:https://www.iso.org

Juan, D., Zheng, Qin. (2014). Cloud and Open BIM-Based Building Information Interoperability Research. *Journal of Service Science and Management, 07.* P. 47-56

King, A.M., Burgess, S.C., Ijomah, W., McMahon, C.A. (2006). Reducing waste: Repair, recondition, remanufacture of recycle? *Sustainable development, 14.* P. 257-267

Landman A. (2017, July 14). Cobouw. *Voor circulaire economie moet bouwsector op de schop.* https://www.cobouw.nl/bouwbreed/nieuws/2017/07/voor-circulaire-economie-23057202.1581591603

Lüdeke-Freun, F., Gold, S., Bocken, N.M.P. (2018). A Review and Typology of Circular Economy Business Model Patterns. *Journal of Industrial Ecology, 23.* P. 36-61

Ma, L., Sacks, R. (2016). A Cloud-Based BIM Platform for Information Collaboration. *International symposium on automation and robotics in construction*. P. 1-8

Madaster. (2020). Waste is material without an identity. Retrieved from: https://www.madaster.com

Martin, J. (1992. Rapid Application Development. *Prentice-Hall, Englewood Cliffs.*

Masoud, F.A., Halabi, D.H. (2006). Frameworks in Model View Controller Implementation. *ASP.NET and JSP.*

McArthur, J.J. (2015). A Building Information Management (BIM) Framework and Supporting Case Study for Existing Building Operations, Maintenance and Sustainability. *Procedia Engineering, 188.* P. 1104-1111

McLellan, H. (2019). Sustainable design project. *Design for Disassembly.* Retrieved from: http://studiof-waste.weebly.com/design-for-disassembly.html

Ministry of Environment. (2019). Circular economy in the construction – Experiences from Finland. https://ec.europa.eu/transparency/regexpert/index.cfm?do=groupDetail.groupMeetingDoc&docid=35658

MongoDB. (2020). MongoDB. *MongoDB Atlas Inc.* Retrieved from: https://mongodb.com

Montague, R. (2016). Building Information Modelling: What information is in the model? *NBS.* Retrieved from: https://www.thenbs.com/knowledge/building-information-modelling-what-information-is-in-the-model

Moses, I. (2019). #BIMTalk: From Level of Development (LOD) to Level of Information Need. *BIM Africa*. Retrieved from: https://bimafrica.org/bimtalk/from-lod-to-level-of-information-need/

Murray A., Skene K., Haynes K., (2015). The Circular Economy: An Interdisciplinary Exploration of the Concept and Application in a Global Context. *Journal of Business Ethics, 140.* P. 369-380

Nawari, N. (2012). BIM-Model Checking in Building Design. *University of Florida*. P. 1-9

NBS, Mordue, S. (2015). BIM levels of information. Retrieved from: https://www.thenbs.com/knowledge/bim-levels-of-information

Ng, W.Y., Chau, C.K. (2015). New Life of the Building Materials- Recycle, Reuse and Recovery. *Energy Procedia, 75*. P. 2884-2891

Node.js. (2020). OpenJS Foundation. Retrieved from: https://nodejs.org/en/

NPM Documentation. (2019). npm-package-lock.json. *A manifestation of the manifest*. Retrieved from: https://docs.npmjs.com/configuring-npm/package-lock-json.html#:~:text=Description,regardless%20of%20intermediate%20dependency%20updates.

Opoku, A. (2015). The Role of Culture in a Sustainable Construction sector. *Sustainable Operations Management.* P. 37-52

Piscicelli, L., Ludden, G. (2016). The potential of Design for Behaviour Change to foster the transition to a circular economy. *Conference paper.* P. 1-16

Pop, D.P., Altar, A. (2014). Designing an MVC Model for Rapid Web Application Development.

Professional QA. (2020). Alpha Testing. *What is Alpha Testing?* Retrieved from: https://www.professionalqa.com/alpha-testing
Procedia Engineering, 69. P. 1172-1179

RAD, Worldpress. (2017). The James Martin RAD method. *The first blog spot*. Retrieved from: https://rapidapplicationdevelopment.wordpress.com/2017/09/10/first-blog-post/

Reenskaug, T. (1970). Model-View-Controller Framework. *Smaltalk-79, Xerox Alto Research Center.*

Respini-Irwin, C. CADalyst. (2012). Open BIM aims to overcome collaboration challenges of modern AECO worksflows. *Vol. 29*. P. 1

Rijksoverheid. (2019). Nederland Circular in 2050. *Ministerie van Infrastructuur en Waterstaat.* Retrieved from: https://www.rijksoverheid.nl/onderwerpen/circulaire-economie/nederland-circulair-in-2050

Rios, F.C., Chong, W., Grau, D. (2015). Design for Disassembly and Deconstruction - Challenges and Opportunities. *Procedia Engineering, 118.* P. 1296-1304

Rios, F.C., Grau, D. (2020). Circular Economy in the Construction sector: Designing, Deconstructing, and Leasing Reusable Products. *Encyclopedia of Renewable and Sustainable Materials*. P. 338-343

Scarpellini S., Portillo-Tarragona P., Aranda-Usón A., Llena-Macarulla F. (2019). Definition and measurement of the circular economy's regional impact. *Journal of Environmental Planning and Management, 62.* P. 2211-2237.

Shan, Y.P. (1989). An Event-Driven Model-View-Controller Framework for Smalltalk. *Department of Computer Science.* P. 347-352

Shen, S. (2019). SQL or NoSQL? *Towards Data Science.* Retrieved from: https://towardsdatascience.com/sql-or-nosql-d3f3e3970635

Smith, P. (2014). BIM Implementation – Global Strategies. *Procedia Engineering, 85.* P. 482- 492

Socotec. (2016). Assure your project in BIM mode. socotec.com/en/assure-your-projects-in-bim-mode

Soust-Verdaguer, B., Llatas, C., Garcia-Martinez, A. (2017). Critical review of bim-based LCA method to buildings. *Energy and Buildings, 136*. P. 1-37

Statistics Netherlands; CBS. (2019). Construction sector leading in waste and recycling. Retrieved from: https://www.cbs.nl/en-gb/news/2019/45/construction-sector-leading-in-waste-and-recycling

Statistics Netherlands; CBS. (2019). Meer grond voor woningen en bedrijven.Retrieved from: https://www.cbs.nl/nl-nl/nieuws/2019/35/meer-grond-voor-woningen-en-bedrijven

Statistics Netherlands; CBS. (2019). Biologische veestapel groeit. Retrieved from: https://www.cbs.nl/nl-nl/nieuws/2019/19/biologische-veestapel-gegroeid

Studios. (2019). Product Design Studios. *6 belangrijke principes van een circulaire economie.* Retrieved from: https://productdesignstudios.nl/belangrijke-principes-van-een-circulaire-economie/

Sun, Z., Xie, J., Zhang, Y., Cao, Y. (2019). As-Built BIM for a Fifteenth-Century Chinese Brick Structure at Various LoDs. *ISPRS International Journal of Geo-Information, 8.* P. 1-15.

System Innovation. (2019). Event-Driven Architecture. *SI*. Retrieved from: https://systemsinnovation.io/event-driven-architecture/

Tekla. (2019). BIM maturity levels. Retrieved from: https://campus.tekla.com/bim-maturity-levels

The American Institute of Architects. (2013). Guide, Instructions and Commentary to the 2013 AIA Digital Practice Documents. *AIA Document E203, G201, G202.* P. 1-62

Thelen, D., van Acoleyen, M., Huurman, W., Thomaes (Arcadis), T., van Brunschot, C., Edgerton, B., & Kubbinga, B. (2018). Scaling the circular built environment. *Pathways for business and government*. P. 1-36.

Theorin, A., Bengtsson, K., Provost, J., Lieder, M., Johnsson, C., Lundholm, T., Lennarson, B. (2015). An Event-Driven Manufacturing Information System Architecture. *IFAC-PapersOnline, 48.* P. 547-554

Trevor, Z., Roland, G. (2017). Circular Economy Rebound. *Journal of Industrial Ecology, 21.* P. 593-602.

Verberne, J. J. H. (2016). Building circularity indicators**.** *Eindhoven, University of Technology.*

Visma. (2015). How APIs are revolutionising the way we buy software. *Are you benefiting from the fast changing market?* P. 1-7

Vliet, M. (2018). Disassembling the steps towards Building Circularity. *Redeveloping the building disassembly assessment method in the building circularity indicator.* Eindhoven, University of Technology.

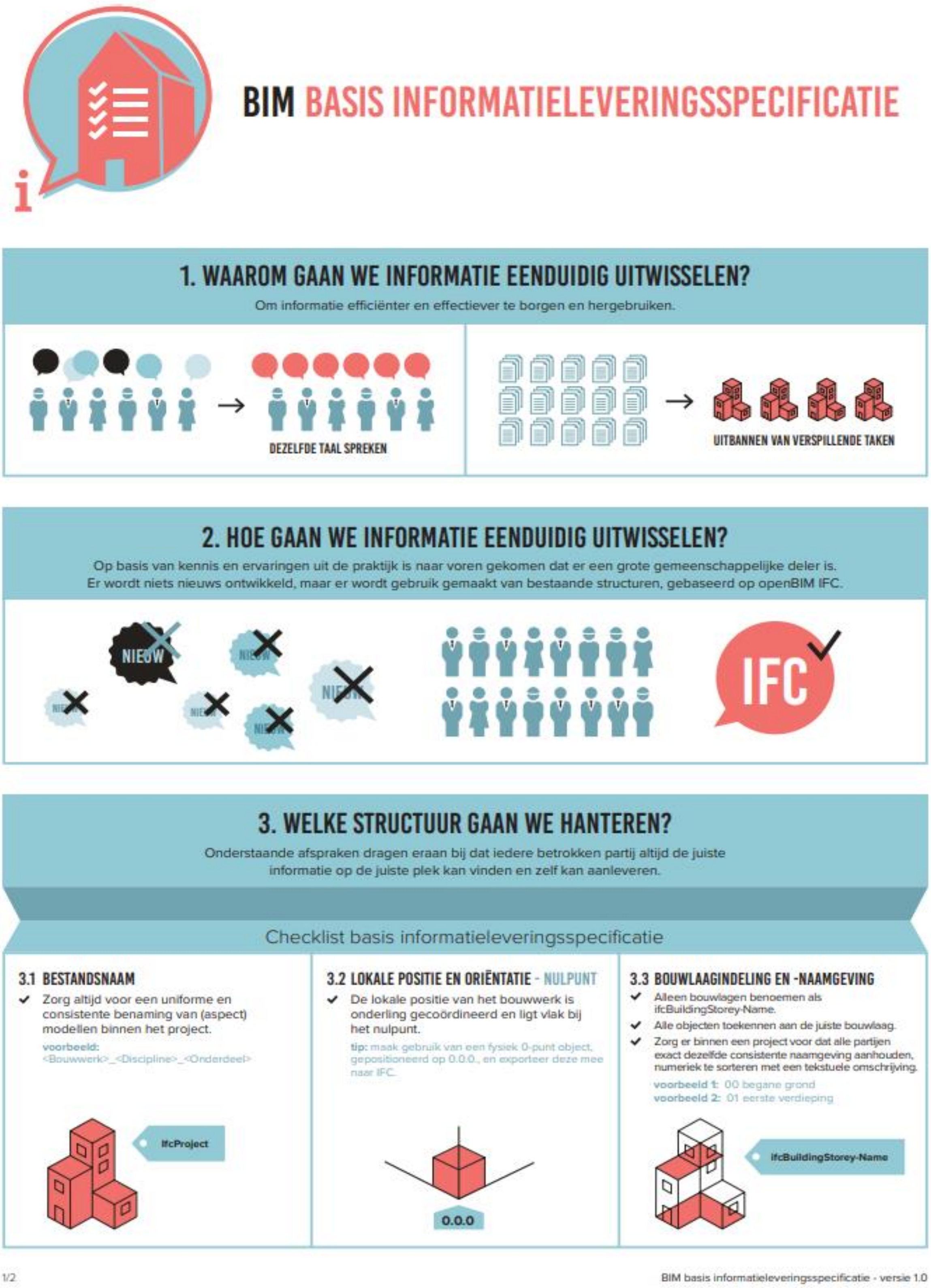
Walter R.S. (2016, March 23) The circular economy. *Nature, International journal of science.* Retrieved from: https://www-nature-com.dianus.libr.tue.nl/news/the-circular-economy-1.19594

Zimmann, R., O'Brien, H., Hargrave, J., Morrell, M. (2016). The circular economy in the built environment. ARUP. P. 1-94.

Zhang, J., Lin, J., Hu, Z., Yu, F. (2012). Research on IDM-based BIM process information exchange technology. *Conference paper*. P. 1-9

10 | Appendices

I | BIM basis-ILS (buildingSMART Benelux – BIM Loket)



BIM BASIS INFORMATIELEVERINGSSPECIFICATIE

1. WAAROM GAAN WE INFORMATIE EENDUIDIG UITWISSELEN?

Om informatie efficiënter en effectiever te borgen en hergebruiken.

DEZELFDE TAAL SPREKEN

UITBANNEN VAN VERSPILLENDE TAKEN

2. HOE GAAN WE INFORMATIE EENDUIDIG UITWISSELEN?

Op basis van kennis en ervaringen uit de praktijk is naar voren gekomen dat er een grote gemeenschappelijke deler is. Er wordt niets nieuws ontwikkeld, maar er wordt gebruik gemaakt van bestaande structuren, gebaseerd op openBIM IFC.

NIEUW

IFC

3. WELKE STRUCTUUR GAAN WE HANTEREN?

Onderstaande afspraken dragen eraan bij dat iedere betrokken partij altijd de juiste informatie op de juiste plek kan vinden en zelf kan aanleveren.

Checklist basis informatieleveringsspecificatie

3.1 BESTANDSNAAM

- ✔ Zorg altijd voor een uniforme en consistente benaming van (aspect) modellen binnen het project.

voorbeeld: <Bouwwerk>_<Discipline>_<Onderdeel>

IfcProject

3.2 LOKALE POSITIE EN ORIËNTATIE - NULPUNT

- ✔ De lokale positie van het bouwwerk is onderling gecoördineerd en ligt vlak bij het nulpunt.

tip: maak gebruik van een fysiek 0-punt object, gepositioneerd op 0.0.0., en exporteer deze mee naar IFC.

0.0.0

3.3 BOUWLAAGINDELING EN -NAAMGEVING

- ✔ Alleen bouwlagen benoemen als ifcBuildingStorey-Name.
- ✔ Alle objecten toekennen aan de juiste bouwlaag.
- ✔ Zorg er binnen een project voor dat alle partijen exact dezelfde consistente naamgeving aanhouden, numeriek te sorteren met een tekstuele omschrijving.

voorbeeld 1: 00 begane grond
voorbeeld 2: 01 eerste verdieping

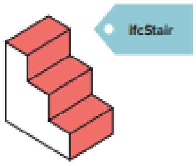
IfcBuildingStorey-Name

1/2 BIM basis informatieleveringsspecificatie - versie 1.0

3.4 CORRECT GEBRUIK VAN ENTITEITEN

- ✔ Gebruik het meest geëigende type BIM-entiteit, zowel in de bronapplicatie als de IFC-entiteit.

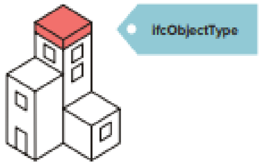
voorbeeld: vloer = ifcSlab, wand = ifcWall, balk = ifcBeam, kolom = ifcColumn, trap = ifcStair, deur = ifcDoor etc.



3.5 STRUCTUUR EN NAAMGEVING

- ✔ Objecten consistent structureren en aanduiden.
- ✔ In basis altijd TYPE (ifcType, ifcObjectType of ifcObjectTypeOverride) van elementen correct invullen.
- ✔ Waar van toepassing ook Name (ifcName of NameOverride) correct invullen.

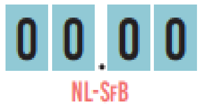
voorbeeld: dakisolatie, type: glaswol



3.6 INFORMATIEINDELING CLASSIFICATIE NL-SFB

- ✔ Voorzie objecten in basis van een viercijferige NL-SfB variant-elementencode.

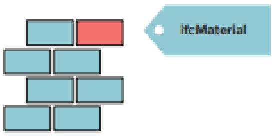
voorbeeld: 22.11



3.7 OBJECTEN VOORZIEN VAN CORRECT MATERIAAL

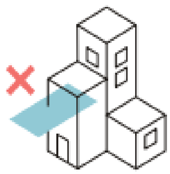
- ✔ Voorzie objecten van een materiaalbeschrijving (ifcMaterial).

voorbeeld: kalkzandsteen



3.8 DOUBLOURES EN DOORSNIJDINGEN

- ✔ In basis zijn doorsnijdingen en doublures in een aspectmodel niet toegestaan. Controleer hierop.



DEZELFDE TAAL LEREN SPREKEN, DOEN WE SAMEN

Bedenk bij het benoemen van objecten of de naam voldoet aan de volgende criteria. Controleer hier op, weet welke informatie je overdraagt.

- ✔ Betekenisvol
- ✔ Begrijpelijk
- ✔ Logisch
- ✔ Inzichtelijk
- ✔ Consistent
- ✔ Herkenbaar

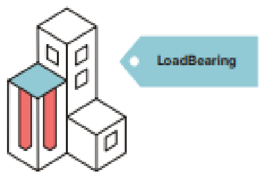
4. HOE BORGEN WE ANDERE/TOEKOMSTIGE OBJECTINFORMATIE?

Objectinformatie wordt geborgd in de juiste property's en propertysets zoals die in IFC zijn gedefinieerd.

voorbeeld: bij balken maken de eigenschappen FireRating, LoadBearing en IsExternal onderdeel uit van de Pset_BeamCommon.

4.1 DRAGEND / NIET DRAGEND - LOADBEARING

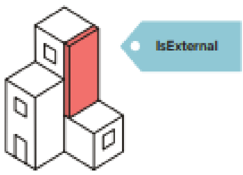
- ✔ Voorzie objecten, wanneer van toepassing, van de eigenschap LoadBearing [True/False].



4.2 IN / UITWENDIG - IS EXTERNAL

- ✔ Voorzie objecten, wanneer van toepassing, van de eigenschap IsExternal [True/False]

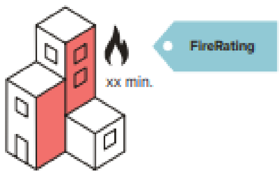
tip: zowel binnenblad als buitenblad van de gevel behoren tot IsExternalTrue.



4.3 BRANDWERENDHEID - FIRERATING

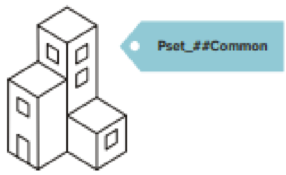
- ✔ Voorzie objecten, wanneer van toepassing, van de eigenschap FireRating.

voorbeeld: Vul hier de wbdbo waarde in minuten in bijvoorbeeld: 30, 60, 90 minuten.



4.4 PROJECTSPECIFIEK

- ✔ Bepaal projectspecifiek welke IFC properties je gebruikt.



II | BIM Model Data Juxtapostition

CENSURED

Identification		Identification		Identification		Identification	
Property	**Value**	**Property**	**Value**	**Property**	**Value**	**Property**	**Value**
Model	Censured	Model	Censured	Model	Censured	Model	Censured
Discipline	Architectural	Discipline	Architectural	Discipline	Architectural	Discipline	Architectural
Name	Railing:NLRS_34_R_UN_railing 850mm_gen:20225481	Name	Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen:21315155	Name	NLRS_63_LF_FB_armatuur_gen:armatuur:23535062	Name	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen:20217430
Phase	definitief	Phase		Phase		Phase	definitief
Type		Type	Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	Type	NLRS_63_LF_FB_armatuur_gen:armatuur	Type	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen
Type Name		Type Name	Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	Type Name	armatuur	Type Name	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen
Predefined Type	NOTDEFINED	Predefined Type		Predefined Type	NOTDEFINED	Predefined Type	FLOOR
Description		Description		Description		Description	
Material	Staal	Material	beton 100 mm	Material		Material	beton 250 mm
Layer	A-FLOR-HRAL-OTLN	Layer	I-WALL-____-OTLN	Layer	E-LITE-EQPM-OTLN	Layer	A-FLOR-____-OTLN
System		System		System		System	
Filled		Filled		Filled		Filled	
Building Envelope		Building Envelope	False	Building Envelope		Building Envelope	
Geometry	Extrusion	Geometry	Extrusion	Geometry	Extrusion	Geometry	Extrusion
Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)
IFC Entity	IfcRailing	IFC Entity	IfcWallStandardCase	IFC Entity	IfcFlowTerminal	IFC Entity	IfcSlab
IFC Type		IFC Type	IfcWalltype	IFC Type	IfcLightFixtureType	IFC Type	IfcSlabType
GUID	adfadfasdfasdf	GUID	alksdjflakjdfl	GUID	hasdfjlkasjdlfjasdff	GUID	asdfasdfasd
BATID	23423412	BATID	64531314135	BATID	54564446	BATID	54564515
Model Categories		Model Categories		Model Categories		Model Categories	

Location		Location		Location		Location	
Property	**Value**	**Property**	**Value**	**Property**	**Value**	**Property**	**Value**
Site	BB-1629A	Site	BB-1629A	Site	BB-1629A	Site	BB-1629A
Building	Censured	Building	Censured	Building	Censured	Building	Censured
Floor	00 begane grond	Floor	02 tweede verdieping	Floor	03 derde verdieping	Floor	01 eerste verdieping
Federated Floor	(A) 00 begane grond	Federated Floor	(A) 02 tweede verdieping	Federated Floor	(A) 03 derde verdieping	Federated Floor	(A) 01 eerste verdieping
System		System		System		System	
Top Elevation	780 mm	Top Elevation	2,590 mm	Top Elevation	2,620 mm	Top Elevation	0 mm
Bottom Elevation	-70 mm	Bottom Elevation	0 mm	Bottom Elevation	2,570 mm	Bottom Elevation	-250 mm
Distance to Next Floor	2,060 mm	Distance to Next Floor	250 mm	Distance to Next Floor	430 mm	Distance to Next Floor	2,840 mm
Global Top Elevation	780 mm	Global Top Elevation	8,270 mm	Global Top Elevation	11,140 mm	Global Top Elevation	2,840 mm
Global Bottom Elevation	-70 mm	Global Bottom Elevation	5,680 mm	Global Bottom Elevation	11,090 mm	Global Bottom Elevation	2,590 mm
Global X	34,612 mm	Global X	45,365 mm	Global X	50,263 mm	Global X	62,735 mm
Global Y	35,728 mm	Global Y	52,913 mm	Global Y	53,379 mm	Global Y	63,440 mm

Quantities		Quantities		Quantities		Quantities	
Property	**Value**	**Property**	**Value**	**Property**	**Value**	**Property**	**Value**
Area		Area	0.97 m2	Area		Area	3.26 m2
Area (minimum)		Area (minimum)	0.97 m2	Area (minimum)		Area (minimum)	
Gross Area		Gross Area	0.97 m2	Gross Area		Gross Area	3.26 m2
Gross Area (minimum)		Gross Area (minimum)	0.97 m2	Gross Area (minimum)		Gross Area (minimum)	
Area of Doors		Area of Doors	0.00 m2	Area of Doors		Area of Doors	
Area of Windows		Area of Windows	0.00 m2	Area of Windows		Area of Windows	
Area of Openings		Area of Openings	0.00 m2	Area of Openings		Area of Openings	0.00 m2
Bottom Area	0.09 m2	Bottom Area	0.04 m2	Bottom Area		Bottom Area	
Height	914 mm	Height	2,590 mm	Height		Height	
Height (minimum)		Height (minimum)	2,590 mm	Height (minimum)		Height (minimum)	
Length		Length	375 mm	Length		Length	
Length (minimum)		Length (minimum)	375 mm	Length (minimum)		Length (minimum)	
Thickness		Thickness	100 mm	Thickness		Thickness	250 mm
Thickness (minimum)		Thickness (minimum)	100 mm	Thickness (minimum)		Thickness (minimum)	
Volume		Volume	0.10 m3	Volume		Volume	0.82 m3
Width		Width		Width		Width	
Frame Length		Frame Length		Frame Length		Frame Length	
Perimeter		Perimeter		Perimeter		Perimeter	7,513 mm
Perimeter of Openings		Perimeter of Openings		Perimeter of Openings		Perimeter of Openings	0 mm
Bounding Box Height	850 mm	Bounding Box Height	2,590 mm	Bounding Box Height	50 mm	Bounding Box Height	250 mm
Bounding Box Length	1,800 mm	Bounding Box Length	375 mm	Bounding Box Length	250 mm	Bounding Box Length	2,280 mm
Bounding Box Width	50 mm	Bounding Box Width	100 mm	Bounding Box Width	100 mm	Bounding Box Width	1,486 mm

Material		Material		Material		Material	
Name	**Thickness**	**Name**	**Thickness**	**Name**	**Thickness**	**Name**	**Thickness**
Staal	0 mm	bestaand	100 mm			beton	250 mm

Profile		Profile		Profile		Profile	
Property	**Value**	**Property**	**Value**	**Property**	**Value**	**Property**	**Value**
Type	Rectangle Profile	Type	Rectangle Profile	Type	Rectangle Profile	Type	Non Uniform L-Shape Profile
Name	NLRs_34_R_UN_railing 850mm_gen	Name		Name	armatuur	Name	
Depth		Depth		Depth		Depth	2,280 mm
Width		Width		Width		Width	1,486 mm
Thickness		Thickness		Thickness		Thickness	1,376 mm
Flange Thickness		Flange Thickness		Flange Thickness		Flange Thickness	1,140 mm
X Dim	1,800 mm	X Dim	375 mm	X Dim	100 mm	X Dim	
Y Dim	50 mm	Y Dim	100 mm	Y Dim	250 mm	Y Dim	

Classification

Classification	Uniformat Classification
Source	From IFC
Reference	16.12
Name	funderingsconstructies; voeten en balken, fundatie balken

Eigenschappen algemeen

Property	Value
Breedte	400 mm
Fase	2. nieuwbouw
Fase gesloopt	
Is.uitwendig	True
Lengte	9,920 mm
NL-SFB Codering	16.12
NL - SFB Omschrijving	funderingsconstructies; voeten en balken, fundatie balken
Naam	Basic Wall: 16_WA_beton ihw 400 gen
Oppervlakte	4.76 m2
Subsysteem	
Type	16_WA_beton ihw 400 gen
Verdieping	Level: ok.fundering
Brandwerendheid	

Pset_ElementShading

Property	Value
Roughness	305 mm

Pset_ProductRequirements

Property	Value
Catergory	Walls

Pset_QuantityTakeOff

Property	Value
Reference	16_WA_beton ihw 400_gen

Pset_ReinforcementBarPitchOfWall

Property	Value
Description	Beton 200mm
Reference	16_WA_beton ihw 400_gen

Pset_WallCommon

Property	Value
Combustible	False
ExtendToStructure	False
IsExternal	True
LoadBearing	True
Reference	16_WA_beton ihw 400_gen
Height	

BaseQuantities

Property	Value
Depth	
Height	
Width	
GrossFootprintArea	
Length	
NetSideArea	
NetVolume	
GrossCeilingArea	
Area	
GrossArea	
GrossVolume	
NetArea	
Perimeter	

Pset_OpeningElementCommon

Property	Value
Reference	
ExtendToStructure	
IsExternal	
LoadBearing	
PitchAngle	

Constraints

Property	Value
Elevation	
Moves With Nearby Elements	
Host	

Classification

Classification	Uniformat Classification
Source	From IFC
Reference	61.50
Name	centrale elektrotechnische voorzieningen; energie, lage spanning, algemeen

Eigenschappen algemeen

Property	Value
Breedte	
Fase	2. nieuwbouw
Fase gesloopt	
Is.uitwendig	True
Lengte	
NL-SFB Codering	61.50
NL - SFB Omschrijving	centrale elektrotechnische voorzieningen; energie, lage spanning, algemeen
Naam	61_GM_PV paneel_30graden:61_GM_PV paneel_30graden
Oppervlakte	2.16 m2
Subsysteem	
Type	61_GM_PV paneel_30graden
Verdieping	Level: 01 eerste verdieping
Brandwerendheid	0

Pset_ElementShading

Property	Value
Roughness	

Pset_ProductRequirements

Property	Value
Catergory	Generic Models

Pset_QuantityTakeOff

Property	Value
Reference	61_GM_PV paneel_30graden

Pset_ReinforcementBarPitchOfWall

Property	Value
Description	
Reference	

Pset_DistributionFlowElementCommon

Property	Value
Combustible	
ExtendToStructure	
IsExternal	
LoadBearing	
Reference	61_GM_PV paneel_30graden
Height	

BaseQuantities

Property	Value
Depth	
Height	
Width	
GrossFootprintArea	
Length	
NetSideArea	
NetVolume	
GrossCeilingArea	
Area	
GrossArea	
GrossVolume	
NetArea	
Perimeter	

Pset_OpeningElementCommon

Property	Value
Reference	
ExtendToStructure	
IsExternal	
LoadBearing	
PitchAngle	

Constraints

Property	Value
Elevation	
Moves With Nearby Elements	
Host	

Classification

Classification	Uniformat Classification
Source	From IFC
Reference	21.11
Name	buitenwanden; niet constructief, massieve wanden

Eigenschappen algemeen

Property	Value
Breedte	140 mm
Fase	2. nieuwbouw
Fase gesloopt	
Is.uitwendig	True
Lengte	5,055 mm
NL-SFB Codering	21.11
NL - SFB Omschrijving	buitenwanden; niet constructief, massieve wanden
Naam	Basic Wall: 21_WA_metselwerk-roodbruin_100_spouw 40
Oppervlakte	0.94 m2
Subsysteem	
Type	21_WA_metselwerk-roodbruin_100_spouw 40
Verdieping	Level: 00 begane grond
Brandwerendheid	

Pset_ElementShading

Property	Value
Roughness	305 mm

Pset_ProductRequirements

Property	Value
Catergory	Walls

Pset_QuantityTakeOff

Property	Value
Reference	21_WA_metselwerk-roodbruin_100_spouw 40

Pset_ReinforcementBarPitchOfWall

Property	Value
Description	
Reference	

Pset_WallCommon

Property	Value
Combustible	
ExtendToStructure	
IsExternal	
LoadBearing	
Reference	
Height	

BaseQuantities

Property	Value
Depth	20 mm
Height	76 mm
Width	653 mm
GrossFootprintArea	
Length	
NetSideArea	
NetVolume	
GrossCeilingArea	
Area	
GrossArea	
GrossVolume	
NetArea	
Perimeter	

Pset_OpeningElementCommon

Property	Value
Reference	21_WA_metselwerk-roodbruin_100_spouw 40
ExtendToStructure	
IsExternal	
LoadBearing	
PitchAngle	

Constraints

Property	Value
Elevation	
Moves With Nearby Elements	
Host	

Classification

Classification	
Source	
Reference	
Name	buitenwanden; constructief, systeemwanden

Eigenschappen algemeen

Property	Value
Breedte	
Fase	
Fase gesloopt	
Is.uitwendig	
Lengte	2,310 mm
NL-SFB Codering	22.13
NL - SFB Omschrijving	binnenwanden; niet-constructief, systeemwanden; vast
Naam	22_WA_knieschot_generiek:22_WA_knieschot_generiek
Oppervlakte	4.56 m2
Subsysteem	
Type	22_WA_knieschot_generiek
Verdieping	
Brandwerendheid	0

Pset_ElementShading

Property	Value
Roughness	

Pset_ProductRequirements

Property	Value
Catergory	Generic Models

Pset_QuantityTakeOff

Property	Value
Reference	22_WA_knieschot_generiek

Pset_ReinforcementBarPitchOfWall

Property	Value
Description	knieschot
Reference	22_WA_knieschot_generiek

Pset_WallCommon

Property	Value
Combustible	False
ExtendToStructure	
IsExternal	False
LoadBearing	
Reference	22_WA_knieschot_generiek
Height	

BaseQuantities

Property	Value
Depth	
Height	
Width	
GrossFootprintArea	
Length	
NetSideArea	
NetVolume	
GrossCeilingArea	
Area	
GrossArea	
GrossVolume	
NetArea	
Perimeter	

Pset_OpeningElementCommon

Property	Value
Reference	
ExtendToStructure	
IsExternal	
LoadBearing	
PitchAngle	

Constraints

Property	Value
Elevation	
Moves With Nearby Elements	
Host	

III | Matching / Semi-Matching / Not matching – Data Juxtaposition

Censured

Identification

Property	Value	Property	Value	Property	Value	Property	Value
Model	Censured	Model	Censured	Model	Censured	Model	Censured
Discipline	Architectural	Discipline	Architectural	Discipline	HVAC	Discipline	Architectural
Name	44_OOA_Multiplexplaat_trap1:44_OOA_Multiplexplaat_trap1:6073441	Name	31_OOA_vulling_bi_deurkader_h0v0:deur:6785362	Name	57_AIR_wall_diffuser_WUAA_return_Solid_Air:WUAA:2156539	Name	74_OOA_toilet_hangend:74_OOA_toilet_hangend:4528777
Phase	n.a.	Phase		Phase		Phase	
Type	44_OOA_Multiplexplaat_trap1	Type	deur	Type	WUAA	Type	74_OOA_toilet_hangend
Type Name	44_OOA_Multiplexplaat_trap1	Type Name	deur	Type Name	WUAA	Type Name	
Predefined Type	NOTDEFINED	Predefined Type		Predefined Type	DIFFUSER	Predefined Type	
Description		Description		Description		Description	
Material	multiplex_verticaal	Material		Material		Material	
Layer	A-GENM-___-OTLN	Layer	A-DOOR-___-OTLN	Layer	M-HVAC-CDFF-OTLN	Layer	P-SANR-FIXT-OTLN
System		System		System	(57.2) 80	System	
Filled		Filled		Filled		Filled	
Building Envelope		Building Envelope	False	Building Envelope		Building Envelope	
Geometry	Extrusion	Geometry	Extrusion	Geometry	Boundary Representation	Geometry	Boundary Representation
Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)	Application	Autodesk Revit 2018 (ENU)
IFC Entity	IfcCovering	IFC Entity	IfcDoor	IFC Entity		IFC Entity	IfcFlowTerminal
IFC Type	IfcCoveringTyper	IFC Type	IfcDoorStyle	IFC Type		IFC Type	
GUID	asdfasdfas	GUID	adfadfadf	GUID	adfadfadf	GUID	adfadfadf
BATID	33541	BATID	234234	BATID	234234	BATID	34154
Model Categories	Architectural	Model Categories	ARC	Model Categories		Model Categories	ARC

Location

Property	Value	Property	Value	Property	Value	Property	Value
Site	Default	Site	Default	Site	Default	Site	Default
Building		Building		Building	-	Building	
Floor	-1 kelderverdieping (b.k. ruwe vloer)	Floor	-1 kelderverdieping (b.k. ruwe vloer)	Floor	-1 kelderverdieping	Floor	00 begane grond (b.k. ruwe vloer)
Federated Floor	(ARC) -1 kelderverdieping (b.k. ruwe vloer)	Federated Floor	(ARC) -1 kelderverdieping (b.k. ruwe vloer)	Federated Floor	(ARC) -1 kelderverdieping (b.k. ruwe vloer)	Federated Floor	(ARC) 00 begane grond (b.k. ruwe vloer)
System		System		System	(57.2) 80	System	
Top Elevation	1,365 mm	Top Elevation	3,960 mm	Top Elevation	2,510 mm	Top Elevation	555 mm
Bottom Elevation	0 mm	Bottom Elevation	1,585 mm	Bottom Elevation	2,390 mm	Bottom Elevation	352 mm
Distance to Next Floor	1,585 mm	Distance to Next Floor	-1,000 mm	Distance to Next Floor	490 mm	Distance to Next Floor	2,445 mm
Global Top Elevation	-185 mm	Global Top Elevation	2,410 mm	Global Top Elevation	1,010 mm	Global Top Elevation	1,965 mm
Global Bottom Elevation	-1,550 mm	Global Bottom Elevation	35 mm	Global Bottom Elevation	890 mm	Global Bottom Elevation	1,762 mm
Global X	37,007 mm	Global X	36,585 mm	Global X	8,907 mm	Global X	38,848 mm
Global Y	10,121 mm	Global Y	14,212 mm	Global Y	-3,153 mm	Global Y	6,787 mm

Quantities

Property	Value	Property	Value	Property	Value	Property	Value
Area		Area	2.49 m2	Area		Area	
Area (minimum)		Area (minimum)		Area (minimum)		Area (minimum)	
Gross Area		Gross Area		Gross Area		Gross Area	
Gross Area (minimum)		Gross Area (minimum)		Gross Area (minimum)		Gross Area (minimum)	
Area of Doors		Area of Doors		Area of Doors		Area of Doors	
Area of Windows		Area of Windows		Area of Windows		Area of Windows	
Area of Openings		Area of Openings		Area of Openings		Area of Openings	
Bottom Area		Bottom Area		Bottom Area		Bottom Area	
Height		Height	2,375 mm	Height		Height	
Height (minimum)		Height (minimum)		Height (minimum)		Height (minimum)	
Length		Length		Length		Length	
Length (minimum)		Length (minimum)		Length (minimum)		Length (minimum)	
Thickness		Thickness		Thckness		Thickness	
Thickness (minimum)		Thickness (minimum)		Thickness (minimum)		Thickness (minimum)	
Volume	0.01 m3	Volume		Volume		Volume	
Width		Width	1,050 mm	Width		Width	
Frame Length		Frame Length	5,800 mm	Frame Length		Frame Length	
Perimeter		Perimeter		Perimeter		Perimeter	
Perimeter of Openings		Perimeter of Openings		Perimeter of Openings		Perimeter of Openings	
Bounding Box Height	1,365 mm	Bounding Box Height	2,375 mm	Bounding Box Height	120 mm	Bounding Box Height	203 mm
Bounding Box Length	1,597 mm	Bounding Box Length	1,050 mm	Bounding Box Length	270 mm	Bounding Box Length	544 mm
Bounding Box Width	290 mm	Bounding Box Width	54 mm	Bounding Box Width	44 mm	Bounding Box Width	367 mm

Material

Name	Thickness	Name	Thickness	Name	Thickness	Name	Thickness
multiplex_verticaal	0 mm	hout-RAL 7006	0 mm	57_Ral_9010	0 mm	riolering	0 mm

Profile

Property	Value	Property	Value	Property	Value	Property	Value
Type	Rectangle Profile	Type	Arbitrary	Type		Type	
Name	44_OOA_Multiplexplaat_trap	Name	deur	Name		Name	
Depth		Depth		Depth		Depth	
Width		Width		Width		Width	
Thickness		Thickness		Thickness		Thickness	
Flange Thickness		Flange Thickness		Flange Thickness		Flange Thickness	
X Dim	2,083 mm	X Dim		X Dim		X Dim	

Censured

Identification

Property	Value (1)	Value (2)	Value (3)	Value (4)
Model	Censured	Censured	Censured	Censured
Name	Basic Wall:16_WA_beton ihw 400_gen:3579201	61_GM_PV paneel_30graden:61_GM_PV paneel_30graden:1943363	Basic Wall:21_WA_metselwerk-lichtbruin_100_spouw-40:3712601	22_WA_knieschot_generiek:22_WA_knieschot_generiek:1929048
Layer	A-WALL-___-OTLN	A-GENM-___-OTLN	A-DOOR-___-OTLN	A-GENM-___-OTLN
IFC Entity	IfcWallStandardCase	IfcEnergyConversionDevice	IfcOpeningElement	IfcWall
GUID	0Q5Wh_pZLBRcCbPD36svMm	lakjdslkfjasd	ldjfalsjdffjasdf	dgasdfadf

Quantities

Property	Value (1)	Value (2)	Value (3)	Value (4)
Bounding Box Height	500 mm	288 mm	2,304 mm	1,070 mm
Bounding Box Length	9,520 mm	1,603 mm	3,420 mm	2,310 mm
Bounding Box Width	400 mm	963 mm	140 mm	246 mm

Censured

Identification

Property	Value (1)	Value (2)	Value (3)	Value (4)
Type		Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	NLRS_63_LF_FB_armatuur_gen:armatuur	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen
Type Name		Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	armatuur	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen
Material	Staal	beton 100 mm		beton 250 mm
IFC Type		IfcWalltype	IfcLightFixtureType	IfcSlabType

Location

Property	Value (1)	Value (2)	Value (3)	Value (4)
Site	BB-1629A	BB-1629A	BB-1629A	BB-1629A
Building	Erasmusweg 1481-1569	Erasmusweg 1481-1569	Erasmusweg 1481-1569	Erasmusweg 1481-1569

Quantities

Property	Value (1)	Value (2)	Value (3)	Value (4)
Area		0.97 m2		3.26 m2
Area of Doors		0.00 m2		
Bottom Area	0.09 m2	0.04 m2		
Height	914 mm	2,590 mm		
Length		375 mm		
Thckness		100 mm		250 mm
Volume		0.10 m3		0.82 m3
Width				
Frame Length				
Perimeter				7,513 mm

Material

Name (1)	Thickness (1)	Name (2)	Thickness (2)	Name (3)	Thickness (3)	Name (4)	Thickness (4)
Staal	0 mm	bestaand	100 mm			beton	250 mm

Profile

Property	Value (1)	Value (2)	Value (3)	Value (4)
Type	Rectangle Profile	Rectangle Profile	Rectangle Profile	Non Uniform L-Shape Profile
Name	NLRs_34_R_UN_railing 850mm_gen		armatuur	
Depth				2,280 mm
Width				1,486 mm
Thickness				1,376 mm
Flange Thickness				1,140 mm
X Dim	1,800 mm	375 mm	100 mm	
Y Dim	50 mm	100 mm	250 mm	

Classification

Property	Value (1)	Value (2)	Value (3)	Value (4)
Reference	34.12	22.11	63.11	23.11
Name	balustrades en leuningen; balustrades, buitenbalustrades	binnenwanden; niet-constructief, massieve wanden	verlichting; standaard, onbewaakt, 220/230 V	vloeren; niet constructief, vrijdragende vloeren

Eigenschappen algemeen

Property	Value (1)	Value (2)	Value (3)	Value (4)
Breedte				
Lengte				
NL-SFB Codering				
NL - SFB Omschrijving				
Naam				
Oppervlakte				
Type				
Brandwerendheid				

Pset_ElementShading

Property	Value (1)	Value (2)	Value (3)	Value (4)
Roughness		914 mm		914 mm

Pset_ProductRequirements

Property	Value (1)	Value (2)	Value (3)	Value (4)
Catergory	Railing:NLRS_34_R_UN_railing 850mm_gen	Walls	Lightning Fixtures	Floors

Pset_QuantityTakeOff

Property	Value (1)	Value (2)	Value (3)	Value (4)
Reference	Raling	Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	NLRS_63_LF_FB_armatuur_gen:armatuur	Floor:NLRS_23_FL_beton dragend binnen-250mm_gen

Pset_ReinforcementBarPitchOfWall

Property	Value (1)	Value (2)	Value (3)	Value (4)
Description		bestaande wand 100 mm		vloer beton dragend 250mm
Reference		Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen		Floor:NLRS_23_FL_beton dragend binnen-250mm_gen

Pset_RailingCommon / Pset_WallCommon / Pset_DistributionFlowElementCommon / Pset_DistributionFlowElementCommon

Property	Value (1) Pset_RailingCommon	Value (2) Pset_WallCommon	Value (3) Pset_DistributionFlowElementCommon	Value (4) Pset_DistributionFlowElementCommon
Reference		Basic Wall:NLRS_22_WA_bestaand binnen beton-100mm_gen	NLRS_63_LF_FB_armatuur_gen:armatuur	

Censured

Identification

Property	Value (1)	Value (2)	Value (3)	Value (4)
Discipline	Architectural	Architectural	Architectural	Architectural
Phase				
Predefined Type				
Description				
System				
Filled			False	
Building Envelope	True			False
Geometry	Extrusion	Boundary Representation	Boundary Representation	Extrusion
Application	Autodesk Revit 2018(ENU)	Autodesk Revit 2018 (ENU)	Autodesk Revit 2018 (ENU)	Autodesk Revit 2018(ENU)
BATID	3579201	35473515		352345245
Model Categories				

Location

Property	Value (1)	Value (2)	Value (3)	Value (4)
Floor	00 begane grond	01 eerste verdieping	00 begane grond	
Federated Floor	00 begane grond	01 eerste verdieping	00 begane grond	
System				
Top Elevation	-480 mm	421 mm	12 mm	
Bottom Elevation	-980 mm	133 mm	-467 mm	
Distance to Next Floor	3,840 mm	2,579 mm	562 mm	
Global Top Elevation	-480 mm	3,421 mm	2,438 mm	
Global Bottom Elevation	-980 mm	3,133 mm	-467 mm	
Global X	5,040 mm	16,995 mm	84,125 mm	
Global Y	-660 mm	6,839 mm	-860 mm	

Quantities

Property	Value (1)	Value (2)	Value (3)	Value (4)
Area (minimum)	4.76 m2			
Gross Area	4.76 m2			
Gross Area (minimum)	4.76 m2			
Area of Windows	0.00 m2			
Area of Openings	0.00 m2			
Height (minimum)	500 mm			
Length (minimum)	9,520 mm			
Thickness (minimum)	400 mm			
Perimeter of Openings				

Material

Name	Thickness (1)	Thickness (2)	Thickness (3)	Thickness (4)

Profile

Property	Value (1)	Value (2)	Value (3)	Value (4)

Classification

	(1)	(2)	(3)	(4)
Classification	Uniformat Classification	Uniformat Classification	Uniformat Classification	
Source	From IFC	From IFC	From IFC	

Eigenschappen algemeen

Property	Value (1)	Value (2)	Value (3)	Value (4)
Fase	2. nieuwbouw	2. nieuwbouw	2. nieuwbouw	
Fase gesloopt				
Is.uitwendig	True	True	True	
Subsysteem				
Verdieping	Level: ok.fundering	Level: 01 eerste verdieping	Level: 00 begane grond	

Pset_ElementShading

Property	Value (1)	Value (2)	Value (3)	Value (4)

Pset_ProductRequirements

Property	Value (1)	Value (2)	Value (3)	Value (4)

Pset_QuantityTakeOff

Property	Value (1)	Value (2)	Value (3)	Value (4)

Pset_ReinforcementBarPitchOfWall

Property	Value (1)	Value (2)	Value (3)	Value (4)

Pset_WallCommon (1, 3, 4) / Pset_DistributionFlowElementCommon (2)

Property	Value (1)	Value (2)	Value (3)	Value (4)
Combustible	False			False
ExtendToStructure	False			
IsExternal	True			False
LoadBearing	True			

BaseQuantities

IV | Single Object BIM Model Data Juxtaposition

Property	Censured	Censured	Censured
Identification			
Property	Value	Value	Value
Model	Censured	Censured	Censured
Name	32_DO_draaideur_Svedex_SL01_opdek:32_DO_draaideur_Svedex_SL01_opdek:1923713	31_OOA_vulling_bi_deurkader_h0v0:deur_app:5387075	NRS_31_DO_UN_deur_SH_gen:deur_links_draaiend:23033101
Type	32_DO_draaideur_Svedex_SL01_opdek	deur_app	deur_links_draaiend
Operation	Single Swing Left	Not Defined	Single Swing Right
Layer	A-DOOR-___-OTLN	A-DOOR-___-OTLN	A-DOOR-____OTLN
Quantities			
Property	Value	Value	Value
Area	2.21 m2	2.29 m2	1.41 m2
Height	2,315 mm	2,465 mm	2,020 mm
Width	954 mm	930 mm	700 mm
Frame Length	5,584 mm	5,860 mm	4,740 mm
Bounding Box Height	2,315 mm	2,465 mm	2,020 mm
Bounding Box Length	954 mm	930 mm	700 mm
Bounding Box Width	160 mm	54 mm	50 mm
Classification			
Reference	32.31	31.22	31.31
Name	binnenwandopeningen; gevuld met deuren, draaideuren	Buitenwandopeningen gevuld met ramen - ramer draaiend aan een kant	buitenwandopeningen; gevuld met deuren, draaideuren
Pset_ProductRequirements			
Property	Value	Value	Value
Catergory	Doors	Doors	Doors
Pset_QuantityTakeOff			
Property	Value	Value	Value
Reference	32_DO_draaideur_Svedex_SL01_opdek	deur_app	NRS_31_DO_UN_deur_SH_gen:deur_links_draaiend
Pset_DoorCommon			
Property	Value	Value	Value
Reference	32_DO_draaideur_Svedex_SL01_opdek	deur_app	NRS_31_DO_UN_deur_SH_gen:deur_links_draaiend
BaseQuantities			
Property	Value	Value	Value
Area	2.35 m2	2.29 m2	1.55 m2
Ifc Dimensions			
Property	Value	Value	Value
Overall Height	2,420 mm	2,465 mm	2,020 mm
Overall Width	1,862 mm	930 mm	700 mm
IfcDoorPanelProperties			
Property	Value	Value	Value
PanelOperation	SWINGING	NOTDEFINED	SWINGING
PanelPosition	NOTDEFINED	NOTDEFINED	NOTDEFINED
Material and Finishes			
Property	Value	Value	Value
materiaal_basis		hout-RAL 7006	

Censured

Identification

Property	Value
Type Name	
IFC Entity	IfcDoor
IFC Type	IfcDoorStyle

Material

Name	Thickness

Profile

Property	Value
Type	
Name	
X Dim	
Y Dim	

Eigenschappen algemeen

Property	Value
Breedte	0 mm
Dikte	0 mm
NL-SFB Codering	32.31
NL - SFB Omschrijving	binnenwandopeningen; gevuld met deuren, draaideuren
Naam	32_DO_draaideur_Svedex_SL01_opdek:32_DO_draaideur_Svedex_SL01_opdek
Oppervlakte	2.35 m2
Type	32_DO_draaideur_Svedex_SL01_opdek

Pset_DoorCommon

Property	Value
FireExit	
SelfClosing	
SmokeStop	

BaseQuantities

Property	Value
Height	
Width	0 mm

Constraints

Property	Value
Moves With Nearby Elements	
Host	
Level	
Offset	

Dimensions

Property	Value
Area	
Volume	
Height	
Width	
dagmaat breedte	
dagmaat hoogte	
dikte	
hoek zwaaisymbool	
kozijn diepte	
positie binnenkant kader	
profiel_breedte_boven	
profiel_breedte_links	
profiel_breedte_onder	
profiel_breedte_rechts	
vulling origin positite in kader	

Identity Data

Property	Value
Assembly Code	
Assembly Description	
Description	
Type Name	
Mark	
Type Mark	

Phasing

Property	Value
Phase Created	

ILS

Censured

Identification

Property	Value
Type Name	deur_app
IFC Entity	
IFC Type	

Material

Name	Thickness
hout-RAL 7006	0 mm

Profile

Property	Value
Type	Arbitrary
Name	deur-app
X Dim	
Y Dim	

Eigenschappen algemeen

Property	Value
Breedte	
Dikte	
NL-SFB Codering	
NL - SFB Omschrijving	
Naam	
Oppervlakte	
Type	

Pset_DoorCommon

Property	Value
FireExit	
SelfClosing	
SmokeStop	

BaseQuantities

Property	Value
Height	2,465 mm
Width	930 mm

Constraints

Property	Value
Moves With Nearby Elements	False
Host	Basic Wall : 21_OOA_isolatie met stucwerk 165
Level	Level: 01 1e verdieping
Offset	530 mm

Dimensions

Property	Value
Area	2.29 m2
Volume	0.10 m3
Height	2,465 mm
Width	930 mm
dagmaat breedte	930 mm
dagmaat hoogte	2,465 mm
dikte	54 mm
hoek zwaaisymbool	15 graden
kozijn diepte	114 mm
positie binnenkant kader	0 mm
profiel_breedte_boven	220 mm
profiel_breedte_links	370 mm
profiel_breedte_onder	370 mm
profiel_breedte_rechts	370 mm
vulling origin positite in kader	10 mm

Identity Data

Property	Value
Assembly Code	31.22
Assembly Description	Buitenwandopeningen gevuld met ramen - ramer draaiend aan een kant
Description	draaiend deel
Type Name	deur_app
Mark	220
Type Mark	30

Phasing

Property	Value
Phase Created	Nieuw

ILS

Censured

Identification

Property	Value
Type Name	
IFC Entity	
IFC Type	

Material

Name	Thickness
Kunststof	0 mm

Profile

Property	Value
Type	Rectangle Profile
Name	deur_links_draaiend
X Dim	700 mm
Y Dim	50 mm

Eigenschappen algemeen

Property	Value
Breedte	
Dikte	
NL-SFB Codering	
NL - SFB Omschrijving	
Naam	
Oppervlakte	
Type	

Pset_DoorCommon

Property	Value
FireExit	False
SelfClosing	False
SmokeStop	False

BaseQuantities

Property	Value
Height	
Width	

Constraints

Property	Value
Moves With Nearby Elements	
Host	
Level	
Offset	

Dimensions

Property	Value
Area	
Volume	
Height	
Width	
dagmaat breedte	
dagmaat hoogte	
dikte	
hoek zwaaisymbool	
kozijn diepte	
positie binnenkant kader	
profiel_breedte_boven	
profiel_breedte_links	
profiel_breedte_onder	
profiel_breedte_rechts	
vulling origin positite in kader	

Identity Data

Property	Value
Assembly Code	
Assembly Description	
Description	
Type Name	
Mark	
Type Mark	

Phasing

Property	Value
Phase Created	

ILS

Censured		Censured		Censured	
Identification		**Identification**		**Identification**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Material		Material		Material	
Quantities		**Quantities**		**Quantities**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Area of Doors		Area of Doors		Area of Doors	
Bottom Area		Bottom Area		Bottom Area	
Length		Length		Length	
Thckness		Thckness		Thckness	
Volume		Volume		Volume	
Perimeter		Perimeter		Perimeter	
Profile		**Profile**		**Profile**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Depth		Depth		Depth	
Width		Width		Width	
Thickness		Thickness		Thickness	
Flange Thickness		Flange Thickness		Flange Thickness	
Eigenschappen algemeen		**Eigenschappen algemeen**		**Eigenschappen algemeen**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Lengte		Lengte		Lengte	
Brandwerendheid		Brandwerendheid		Brandwerendheid	
Pset_ReinforcementBarPitchOfWall		**Pset_ReinforcementBarPitchOfWall**		**Pset_ReinforcementBarPitchOfWall**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Description		Description		Description	
Reference		Reference		Reference	
Pset_DoorCommon		**Pset_DoorCommon**		**Pset_DoorCommon**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Height		Height		Height	
AcousticRating		AcousticRating		AcousticRating	
BaseQuantities		**BaseQuantities**		**BaseQuantities**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Depth		Depth		Depth	
GrossFootprintArea		GrossFootprintArea		GrossFootprintArea	
Length		Length		Length	
NetSideArea		NetSideArea		NetSideArea	
NetVolume		NetVolume		NetVolume	
GrossCeilingArea		GrossCeilingArea		GrossCeilingArea	
GrossArea		GrossArea		GrossArea	
GrossVolume		GrossVolume		GrossVolume	
NetArea		NetArea		NetArea	
Perimeter		Perimeter		Perimeter	
Pset_OpeningElementCommon		**Pset_OpeningElementCommon**		**Pset_OpeningElementCommon**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Reference		Reference		Reference	
ExtendToStructure		ExtendToStructure		ExtendToStructure	
IsExternal		IsExternal		IsExternal	
LoadBearing		LoadBearing		LoadBearing	
PitchAngle		PitchAngle		PitchAngle	
Constraints		**Constraints**		**Constraints**	
Property	**Value**	**Property**	**Value**	**Property**	**Value**
Elevation		Elevation		Elevation	

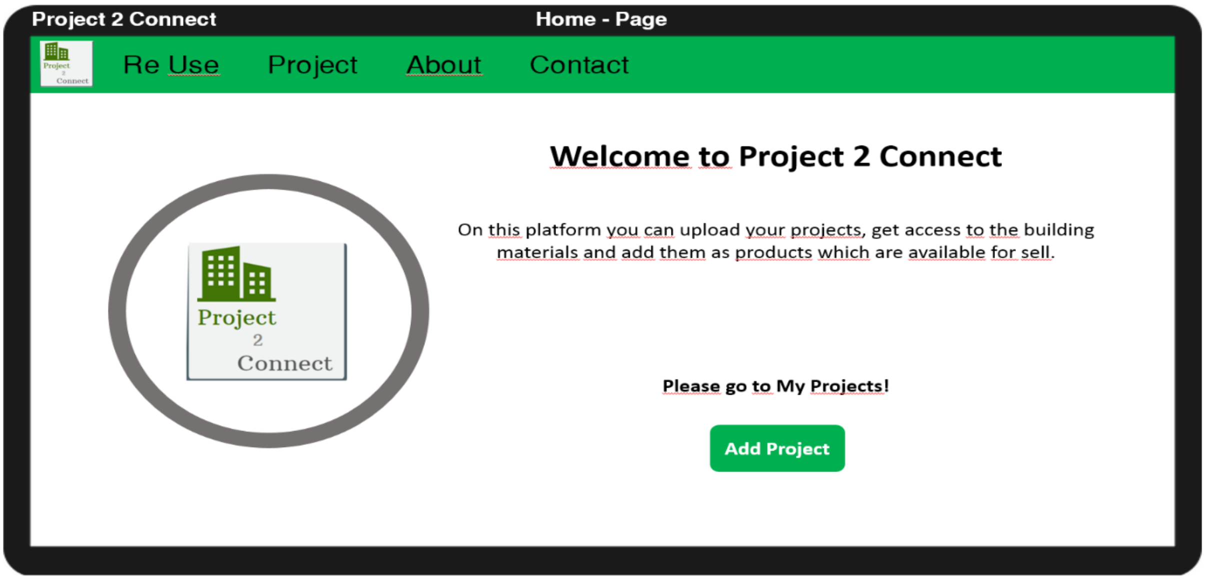
V | ITO Report

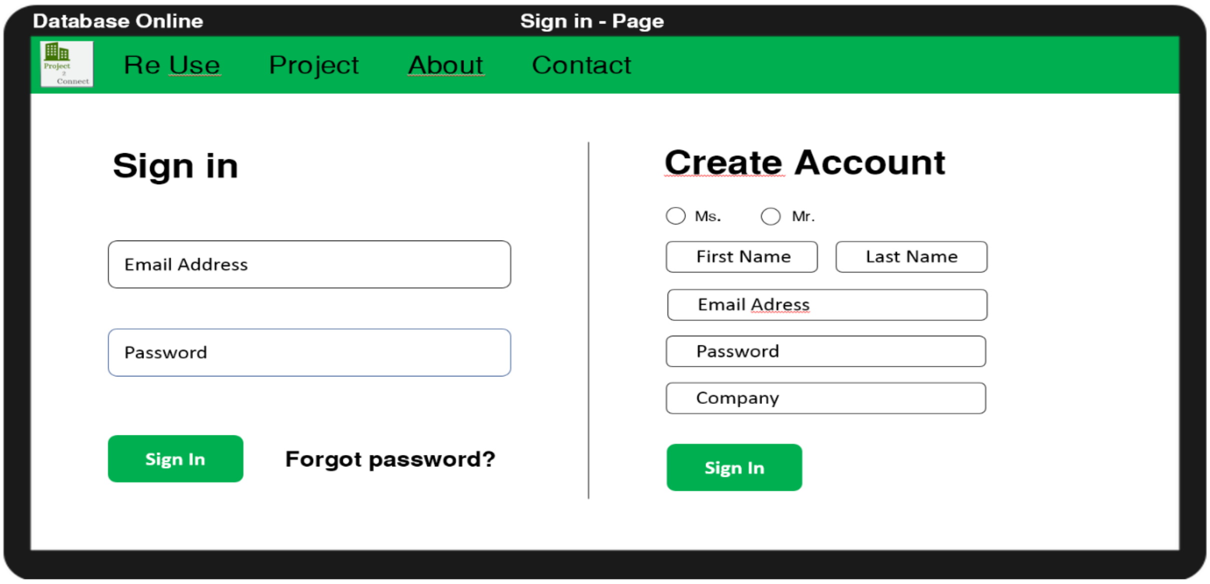
Name	Component	Count	IfcEntity	NlSfb	Material	Area	Height	Length	Width	Thickness	Volume
Name of product	*Classification*	*How many?*	*Not Required*	*Not Required*	*Which material*	*In m2*	*In mm*	*In mm*	*In mm*	*In mm*	*In m3*
Structural Column	Column	1	-	-	Steel	2	2000	300	300	-	0,8
External Window	Window	5	-	-	Wood. Red Paint	1	300	300	-	45	0.15
47_dakafwerking_omloopgoot	Pipe	10	-	-							0,04
47_dakafwerking_omloopgoot	Pipe	10	-	-							0,06
47_nokvorsten	Covering	5	-	-	ISR_Dak_pannen_	0				0	0,04
51_CV_ketel_Intergas_HRE36/30	Gas Terminal	10	-	-	Default						0,08
52_afvoer_bekersifon_generiek	Valve	20	-	-							0
52_hwa_afvoer_noodoverstort	Flow Terminal	12	-	-	00_Kozijn						0
52_hwa_afvoer_stadsuitloop	Flow Terminal	12	-	-	00_Kunststof_PVC_grijs						0,01
52_hwa_afvoer_standleiding	Pipe	20	-	-							0
53_waterleiding_hoekstopkraan_generiek	Valve	30	-	-							0
57_ME_rookafvoer_verzamelkap_aluminium	Object	10	-	-	Grey		0		0		0,17
57_mechanische_ventilatie_Duco_Focus	Air Terminal Box	10	-	-							0,03
57_ventilatierooster_Bergschenhoek_VEF Ex	Air Terminal	70	-	-							0
57_ventilatierooster_ducoline 80 zr_duco	Air Terminal	70	-	-	BERSnl_a_geperforeerd						0
61_GM_PV paneel_30graden	Energy Conversion D	72	-	-	ISR_Zonnepaneel, ISR_Dak_zink						0,07
61_elektra_aansluitpunt-SH_2 fase	Outlet	10	-	-	00_Kunststof_wit						0
64_COM_WPB_beldrukker_generiek	Outlet	10	-	-	00_Kunststof_wit						0
64_EE_WPB_schel ontvanger_generiek	Outlet	10	-	-	00_Kunststof_wit						0
73_aanrechtblok hoek_generiek	Furniture	5	-	-			0		0		2,76
73_aanrechtblok hoek_generiek_oud	Furniture	5	-	-			0		0		2,76
74_PF_FB_wasbak-dubbel-sphinx-320-130-w	Opening	20	-	-		0	6		889		0
74_PF_FB_wasbak-dubbel-sphinx-320-130-w	Sanitary Terminal	10	-	-	bcb_porselein						0,03
74_fontein_generiek	Sanitary Terminal	10	-	-							0,01
74_inbouwreservoir_generiek	Sanitary Terminal	20	-	-							0,03
74_toilet_wand_generiek	Sanitary Terminal	15	-	-							0,03
74_toilet_wand_std	Sanitary Terminal	5	-	-							0,03
74_wastafel_generiek	Sanitary Terminal	10	-	-							0,02
75_ankerpunt AllRisk - Borgpunt XSImpact 36	Discrete Accessory	30	-	-	75_metaal_generiek						0
85_wasdroger_generiek	Sanitary Terminal	15	-	-							0,27
90_FL_terrasvloer_400x600x50	Covering	20	-	-	ISR_Tegels_grijs_	0				0	0,14
90_FL_terrasvloer_400x600x50 ENKEL	Covering	100	-	-	00_Vloer 50 mm	0				0	0,01
90_auto-abstract-model-4-deurs-sedan	Object	10	-	-	bcb_staal, bcb_glas, bcb_alum		0		0		4,68
Area	Space	1	-	-		1064,79	3000				3194,37
Area	Space	5	-	-		114,63	3000				343,9
Area	Space	10	-	-		37,31	3000				111,92
41_WA_tegelwand150x300_15	Wall	5	-	-	00_Tegels_creme	0,21	1160	185		15	0
41_WA_tegelwand150x300_15	Wall	5	-	-	00_Tegels_creme	0,22	1410	155		15	0
41_WA_tegelwand150x300_15	Wall	5	-	-	00_Tegels_creme	0,22	1450	155		15	0
41_WA_tegelwand150x300_15	Wall	5	-	-	00_Tegels_creme	0,44	2590	170		15	0,01

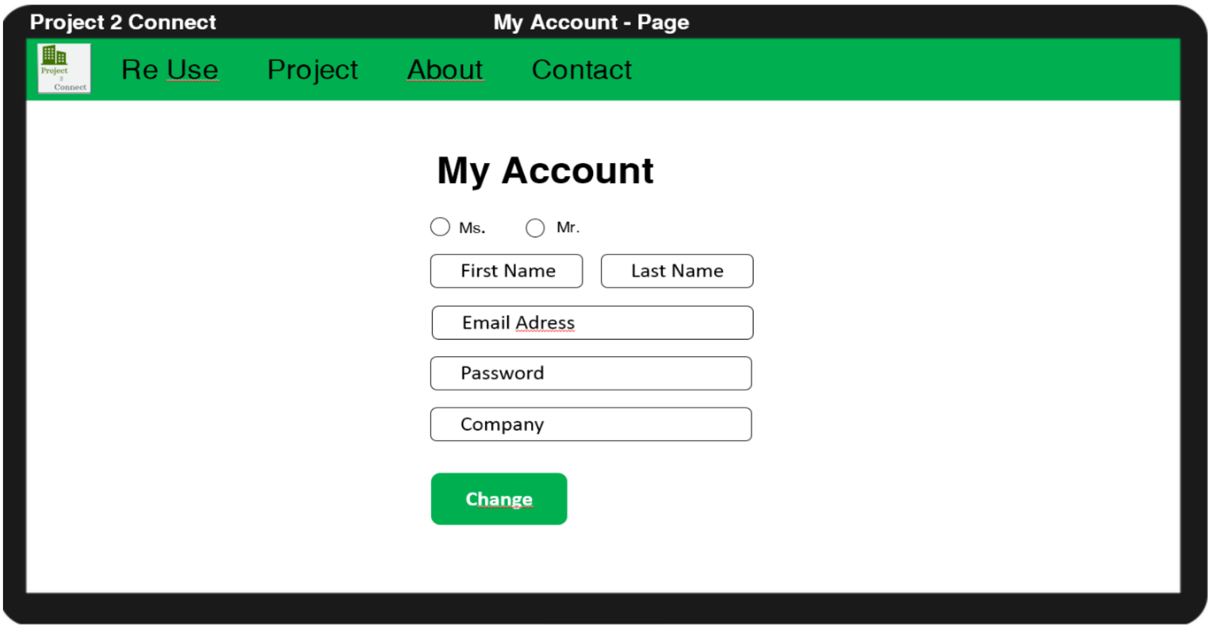
VI | Project Report Template

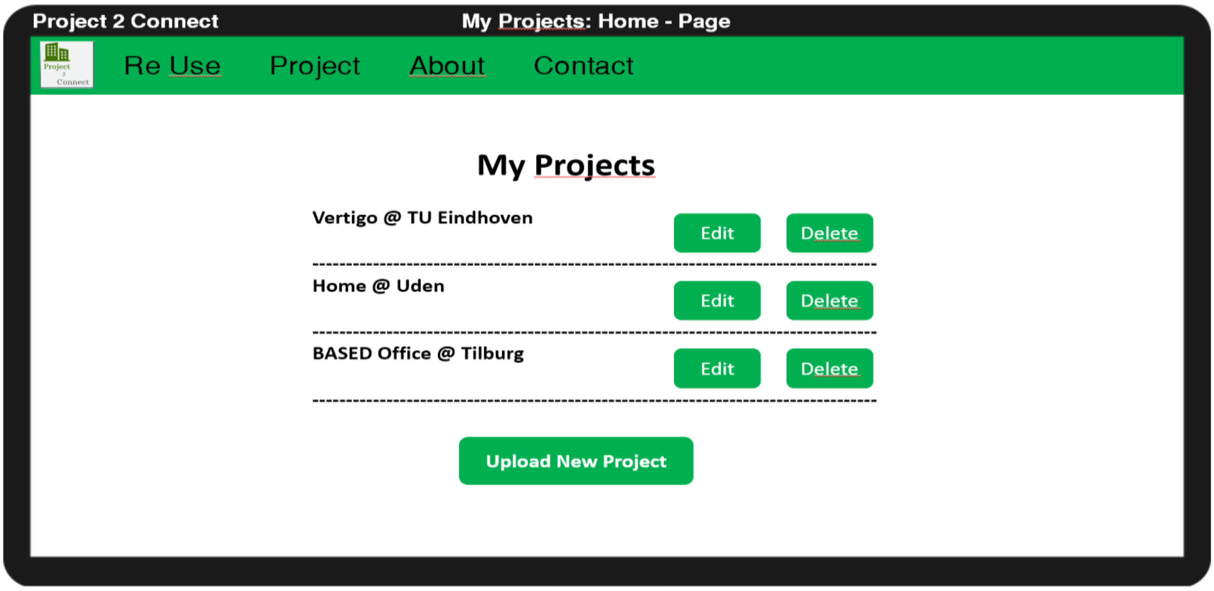
Name	Component	Count	IfcEntity	NlSfb	Material	Area	Height	Length	Width	Thickness	Volume
16_WA_beton ihw 400_gen	Wall	1	IfcWallStandardCase	16.12	00_Beton_generiek 400 mm	4,76	500	9520		400	1,9
21_WA_cellenbeton 100_gen	Wall	4	IfcWallStandardCase	21.11	21_cellenbeton 100 mm	3,43	370	9274		100	0,34
21_WA_iso hard 120_gen	Opening	1	IfcOpeningElement	21.12		9,92	2902		3420		3,86
21_WA_iso hard 120_gen	Wall	1	IfcWall	21.12	00_Isolatie_generiek 120 mm	11,76	3460	4870		120	1,4
21_WA_iso hard 120_gen	Wall	16	IfcWallStandardCase	21.12	00_Isolatie_generiek 120 mm	9,89	3000	4960		120	1,19
21_WA_metselwerk-lichtbruin_100_spouw-40	Opening	3	IfcOpeningElement	21.11		9,92	2902		10260		3,86
21_WA_metselwerk-lichtbruin_100_spouw-40	Opening	40	IfcOpeningElement	21.11		0,04	76		22820		0
21_WA_metselwerk-lichtbruin_100_spouw-40	Opening	72	IfcOpeningElement	21.11		0,05	76		47430		0
21_WA_metselwerk-lichtbruin_100_spouw-40	Wall	1	IfcWallStandardCase	21.11	00_Metselwerk lichtbruin 100 mm, 00_Luchtspouw 4	1,22	125	9800		140	0,16
31_DO_merk A_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, 00_Glas, RAL 9010 - Wit	2,54	2452		5170		0,15
31_DO_merk Asp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, 00_Glas, RAL 9010 - Wit	2,54	2452		5170		0,15
31_DO_merk B_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, RAL 9010 - Wit	2,54	2452		5170		0,15
31_DO_merk Bsp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL Groen - Groen, RAL 9010 - Wit	2,54	2452		5170		0,15
31_DO_merk C_Rechts_3420x2452mm*	Door	10	IfcDoor	31.31	00_Glas, RAL 9010 - Wit, 00_Metselwerk_rood_spek	8,43	2465		34200		0,38
31_DO_merk Csp_links_3420x2452mm*	Door	10	IfcDoor	31.31	00_Glas, RAL 9010 - Wit, 00_Metselwerk_rood_spek	8,42	2462		34200		0,38
31_DO_merk D_rechts_1984x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	4,82	2452		9830		0,23
31_DO_merk Dsp_links_1984x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	4,82	2452		9830		0,23
31_DO_merk E_rechts_1034x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	2,54	2452		5170		0,13
31_DO_merk Esp_links_1034x2452mm*	Door	5	IfcDoor	31.31	RAL 9010 - Wit, 00_Glas	2,54	2452		5170		0,13
31_WI_merk F_1550x1852mm*	Window	2	IfcWindow	31.20	00_Glas, RAL 9010 - Wit	2,89	1862		3100		0,13
31_WI_merk G_1550x1550*	Window	10	IfcWindow	31.20	00_Glas, RAL 9010 - Wit	2,4	1550		15500		0,1
31_WI_merk Gsp_1550x1550*	Window	10	IfcWindow	31.20	00_Glas, RAL 9010 - Wit	2,4	1550		15500		0,1
31_latei_metselwerk	Member	52	IfcMember	31.0							0
32_DO_binnenkozijn_Svedex_Match_BO1_2315_780_2640	Door	10	IfcDoor	32.31	RAL 2000 - Geeloranje	1,94	2356		8250		0
32_DO_binnenkozijn_Svedex_Match_BO1_2315_880_2640	Door	30	IfcDoor	32.31	RAL 2000 - Geeloranje	2,18	2356		27750		0
32_DO_binnenkozijn_Svedex_Match_BO1_2315_930_2640 2	Door	80	IfcDoor	32.31	RAL 2000 - Geeloranje	2,3	2356		78000		0
32_DO_draaideur_Svedex_SL01_opdek	Door	10	IfcDoor	32.31		1,86	2315		8040		0,07
32_DO_draaideur_Svedex_SL01_opdek	Door	30	IfcDoor	32.31		2,09	2315		27120		0,08
32_DO_draaideur_Svedex_SL01_opdek	Door	80	IfcDoor	32.31		2,21	2315		76320		0,08
37_WI_GGL_Integra_M04	Window	30	IfcWindow	37.24	Aluminium. Color NCS 7500-N, white paint color NCS	0,79	982		24095		0,04
41_WA_multiplex 18_gen	Wall	10	IfcWallStandardCase	42.12	00_Multiplex 18 mm	2,3	2610	880		18	0,04
41_WA_tegelwand150x300_15	Opening	2	IfcOpeningElement	42.12		0,91	1200		1475		0,01
41_WA_tegelwand150x300_15	Opening	3	IfcOpeningElement	42.12		0,9	1180		2212		0,01
41_WA_tegelwand150x300_15	Opening	5	IfcOpeningElement	42.12		0,04	1180		75		0
41_WA_tegelwand150x300_15	Opening	5	IfcOpeningElement	42.12		0,86	1160		3687		0,01
41_WA_tegelwand150x300_15	Opening	5	IfcOpeningElement	42.12		1,07	1450		3650		0,02
41_WA_tegelwand150x300_15	Opening	5	IfcOpeningElement	42.12		1,07	1470		3650		0,02
41_WA_tegelwand150x300_15	Opening	15	IfcOpeningElement	42.12		0,25	1180		2603		0
41_WA_tegelwand150x300_15	Wall	1	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	1,12	1410	795		15	0,02
41_WA_tegelwand150x300_15	Wall	2	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	2,1	1932	1604		15	0,03
41_WA_tegelwand150x300_15	Wall	3	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	2,11	1932	1604		15	0,03
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	2,7	2590	1045		15	0,04
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	3,37	2590	1300		15	0,05
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	3,39	2590	1308		15	0,05
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	6	2630	2744		15	0,09
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	6,01	2630	2751		15	0,09
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	6,51	2590	2515		15	0,1
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	6,55	2590	2530		15	0,1
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	8,53	2590	3295		15	0,13
41_WA_tegelwand150x300_15	Wall	5	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	8,57	2590	3310		15	0,13
41_WA_tegelwand150x300_15	Wall	10	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	0,18	2630	70		15	0
41_WA_tegelwand150x300_15	Wall	10	IfcWallStandardCase	42.12	00_Tegels_creme_300x150 15 mm	2,72	2590	1050		15	0,04

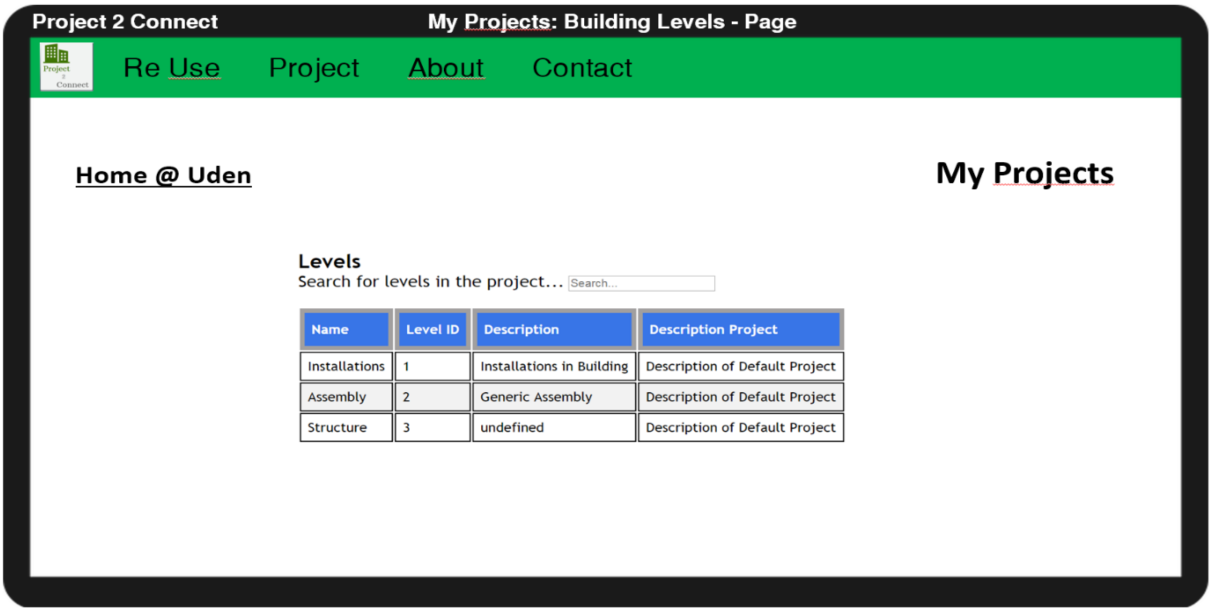
VII | Drafted Scenarios

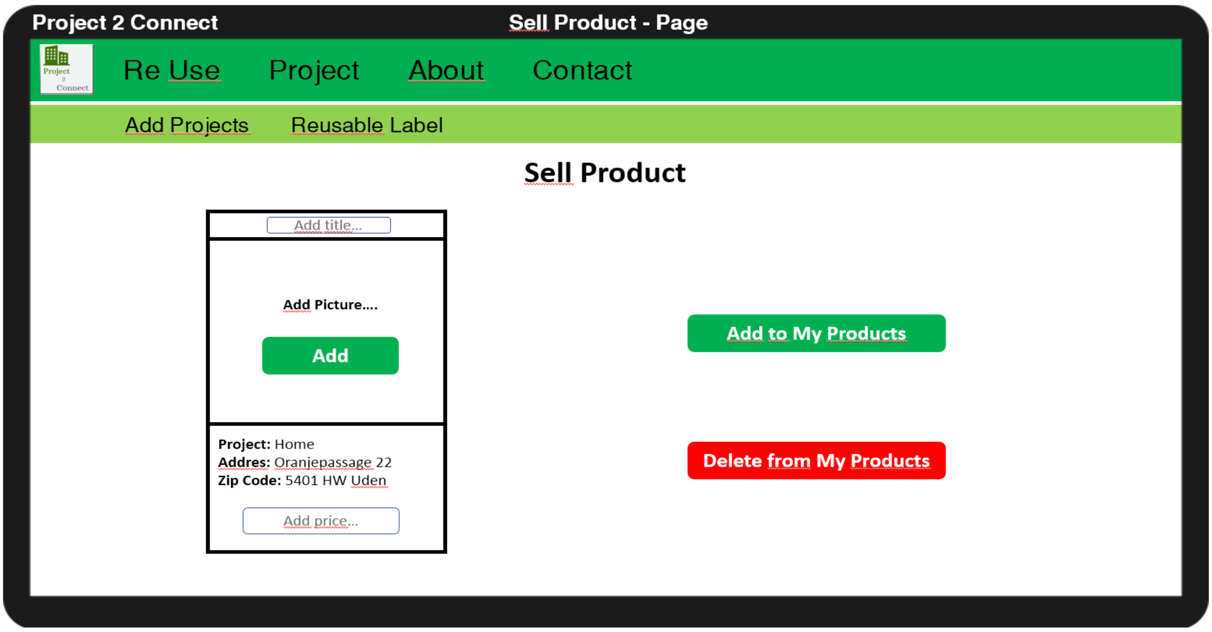


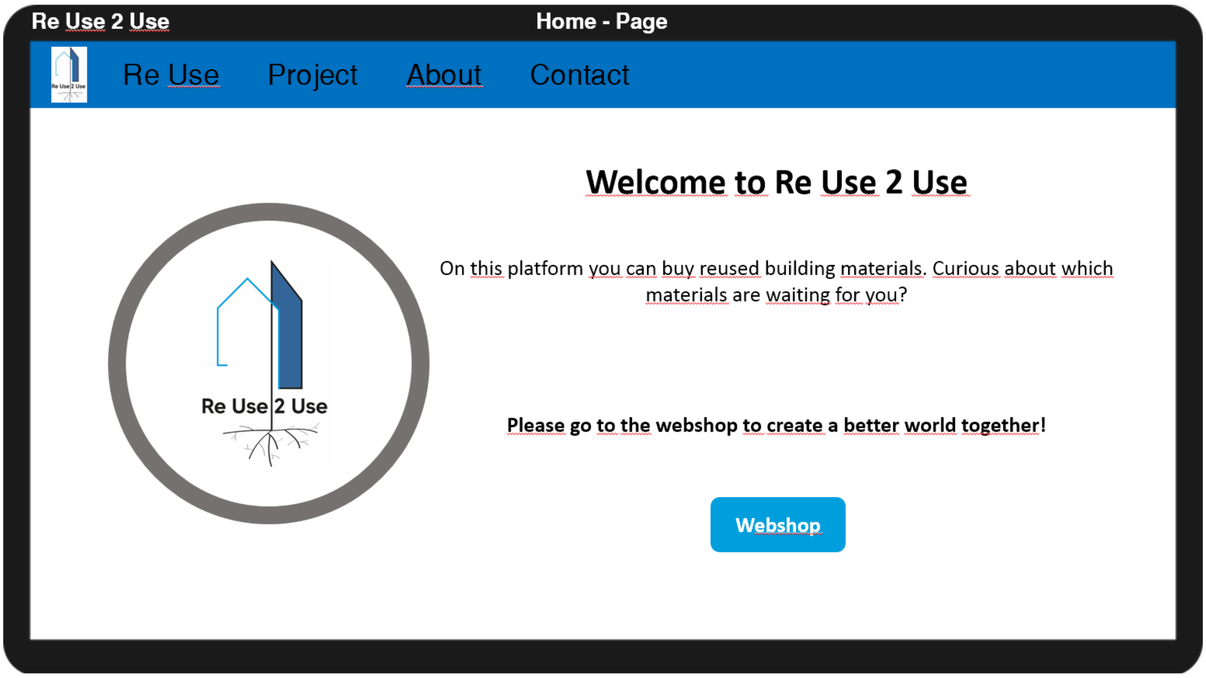


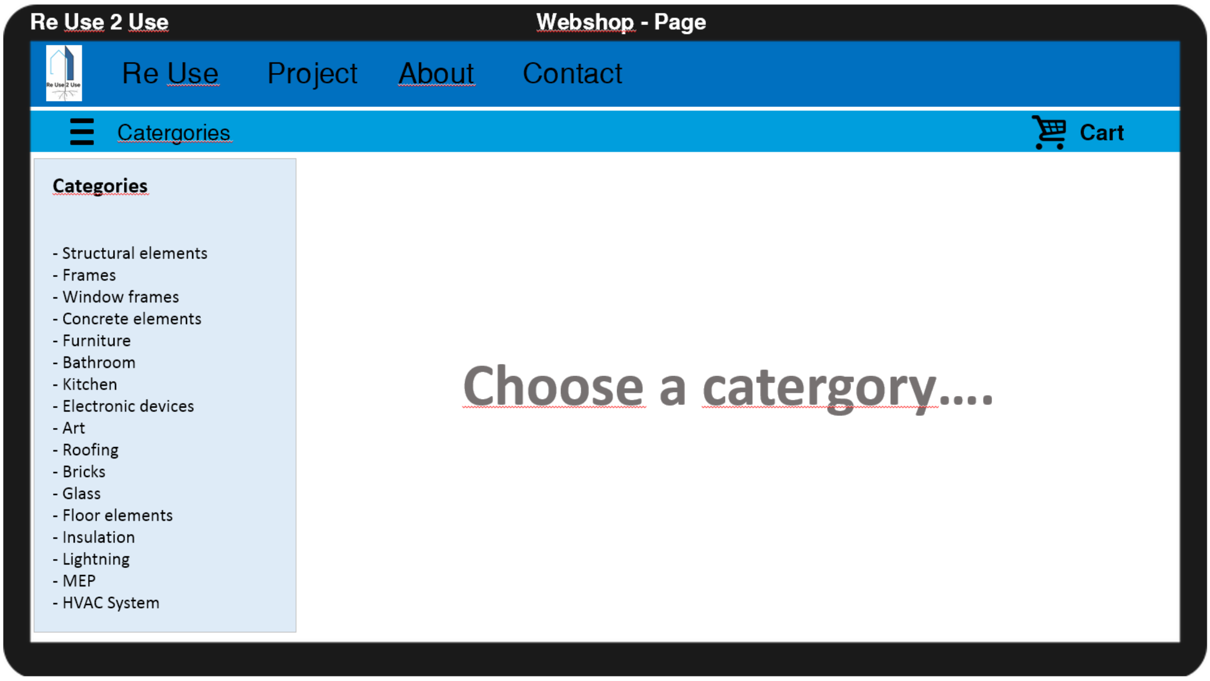



Project 2 Connect
My Projects: Home - Page
Re Use
Project
About
Contact
My Projects
Vertigo @ TU Eindhoven
Edit
Delete
Home @ Uden
Edit
Delete
BASED Office @ Tilburg
Edit
Delete
Upload New Project


Project 2 Connect
My Projects: Building Levels - Page
Re Use
Project
About
Contact
Home @ Uden
My Projects
Levels
Search for levels in the project...
Search...
Name
Level ID
Description
Description Project
Installations
1
Installations in Building
Description of Default Project
Assembly
2
Generic Assembly
Description of Default Project
Structure
3
undefined
Description of Default Project


Project 2 Connect
Sell Product - Page
Re Use
Project
About
Contact
Add Projects
Reusable Label
Sell Product
Add title...
Add Picture....
Add
Project: Home
Addres: Oranjepassage 22
Zip Code: 5401 HW Uden
Add price...
Add to My Products
Delete from My Products


Re Use 2 Use
Home - Page
Re Use
Project
About
Contact
Re Use 2 Use
Welcome to Re Use 2 Use
On this platform you can buy reused building materials. Curious about which materials are waiting for you?
Please go to the webshop to create a better world together!
Webshop


Re Use 2 Use
Webshop - Page
Re Use
Project
About
Contact
Catergories
Cart
Categories
- Structural elements
- Frames
- Window frames
- Concrete elements
- Furniture
- Bathroom
- Kitchen
- Electronic devices
- Art
- Roofing
- Bricks
- Glass
- Floor elements
- Insulation
- Lightning
- MEP
- HVAC System
Choose a catergory....

VIII | Back-End Online Application

For the complete script of the online application, please contact me.

```
var formidable = require('formidable');
var path = require('path');
var fs = require('fs');
var async = require('async');

var Project = require('../models/project');
var User = require('../models/user');

var {arrayAverage} = require('../myFunctions');


module.exports = (app) => {

    app.get('/database/my-project', (req, res) => {
        var success = req.flash('success');
        res.render('database/my-project', {title: 'Project Registration', user: req.user, success:success, noErrors: success.length > 0});
    });

    app.post('/database/my-project', (req, res) => {

        var newProject = new Project();
        newProject.name = req.body.name;
        newProject.address = req.body.address;
        newProject.city = req.body.city;
        newProject.country = req.body.country;
        newProject.mobile = req.body.mobile;
        newProject.email = req.body.email;
        newProject.file = req.body.upload;

        newProject.save((err) => {
            if(err){
                console.log(err);
            }

            console.log(newProject);

            res.redirect('/database/projects');
        })
    });

    app.post('/upload', (req, res) => {
        var form = new formidable.IncomingForm();

        form.uploadDir = path.join(__dirname, '../public/uploads');

        form.on('file', (field, file) => {
           fs.rename(file.path, path.join(form.uploadDir, file.name), (err) => {
```

```
app.get('/database/home', (req, res) => {
    Project.find({}, (err, result) => {
        res.render('database/home', {title: 'Hompage || Database Online', user: req.user, data: result});
    });
});

app.get('/database/projects', (req, res) => {
    Project.find({}, (err, result) => {
        res.render('database/projects', {title: 'Projects || Database Online', user: req.user, data: result});
    });
});

app.get('/project-profile/:id', (req, res) => {
    Project.findOne({'_id':req.params.id}, (err, data) => {
        var avg = arrayAverage(data.ratingNumber);

        res.render('database/project-profile', {title: 'Project Name', user:req.user, id: req.params.id, data:data, average: avg});
    });
});

app.get('/database/add-product/:id', (req, res) => {
    Project.findOne({'_id':req.params.id}, (err, data) => {
        res.render('database/add-product', {title: 'Add Product', user:req.user, data: data});
    });
});

app.post('/database/add-product/:id', (req, res, next) => {
    async.parallel([
        function(callback){
            Project.update({
                '_id': req.params.id,
                'products.productId': {$ne: req.user._id}
            },
            {
                $push: {products: {productId: req.user._id, productName:req.body.name, productComponent:req.body.component, productCount:req.body.count, product
```

```
var formidable = require('formidable');
var path = require('path');
var fs = require('fs');
var async = require('async');

var Project = require('../models/project');
var User = require('../models/user');

var {arrayAverage} = require('../myFunctions');


module.exports = (app) => {

    app.get('/reuse2use/webshop', (req, res) => {
        Project.find({}, (err, result) => {
            res.render('reuse2use/webshop', {title: 'Webshop || Re Use 2 Use', user: req.user, data: result});
        });
    });

    app.get('/reuse2use/product-info/:id', (req, res) => {
        Project.findOne({'_id':req.params.id}, (err, data) => {
            var avg = arrayAverage(data.ratingNumber);

            res.render('reuse2use/product-info', {title: 'Product Information', user:req.user, id: req.params.id, data:data, average: avg});
        });
    });

    app.get('/:name/products', (req, res) => {
        Project.findOne({'name':req.params.name}, (err, data) => {
            res.render('reuse2use/products', {title: 'Project Products', user: req.user, data: data});
        });
    });

  app.get('/database/converter', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/converter', {title: 'Converter', user: req.user, data: data});
      });
  });

  app.get('/database/solibri', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/solibri', {title: 'Solibri', user: req.user, data: data});
      });
  });

  app.get('/database/home-project', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/home-project', {title: 'Project Home', user: req.user, data: data});
      });
  });

  app.get('/database/private', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/private', {title: 'Private User', user: req.user, data: data});
      });
  });

  app.get('/database/getstarted', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/getstarted', {title: 'Get Started!', user: req.user, data: data});
      });
  });

  app.get('/database/reuseablelabel', (req, res) => {
      Project.findOne({'name':req.params.name}, (err, data) => {
          res.render('database/reuseablelabel', {title: 'Reusable Label', user: req.user, data: data});
      });

    app.get('/database/search', (req, res) => {
        res.render('database/search', {title: 'Find a project', user:req.user});
    });

    app.post('/database/search', (req, res) => {
        var name = req.body.search;
        var regex = new RegExp(name, 'i');

        Project.find({'$or': [{'name':regex}]}, (err, data) => {
            if(err){
                console.log(err);
            }

            res.redirect('/project-profile/'+data[0]._id);
        });
    });
```

```
<% layout('layout2') %>

<div id="wrapper">

    <% include ../navigation/sidebar-do %>

    <script src="/javascript/upload.js"></script>

    <body>


    <div id="page-content-wrapper-ss">

            <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
            <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

            <script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
            <script src="/javascript/table.js"></script>


            <div class="row no-gutters">


                <div class="col-left">

                    <div class="leftside">

                        <div class="bodyDiv">
                            <h4 style="">Add Product</h4>

                            <div class="form-group">
                                <label for="">Project</label><br>
                                <label><%= data.name %></label>
                            </div>

                            <form action="" method="post">
                                <div class="form-group ">
                                    <label for="">Name</label>
                                    <input id="role" class="form-control" name="name" type="text" style="margin-bottom:20px;">
                                </div>

                                <div class="form-group ">
                                    <label for="">Component</label>
                                    <input id="component" class="form-control" name="component" type="text" style="margin-bottom:20px;">
                                </div>
                <div class="form-group ">
                    <label for="">Email</label>
                    <span id="errorMsg6"></span>
                    <input id="email" class="form-control" name="email" type="text" style="margin-bottom:20px;">
                </div>

                <div class="form-group ">
                    <label for="">Project Report</label><br>

                    <button class="btn btn-lg" type="button" data-toggle="modal" data-target="#file">Add File</button>
                    <span id="completed"></span>

                    <div id="file" class="modal" tabindex="-1" role="dialog">
                        <div class="modal-dialog">
                            <div class="modal-content">
                                <div class="modal-header">
                                    <button type="button" id="close" class="close" data-dismiss="modal">&times;</button>
                                    <div class="modal-title">Upload Report</div>
                                </div>

                                <div class="modal-body">
                                    <div class="row">
                                        <div class="progress">
                                            <div class="progress-bar" role="progressbar"></div>
                                        </div>

                                        <button class="btn btn-lg upload-btn" type="button">Upload Report</button>
                                        <input type="file" class="form-control" name="upload" id="upload-input" style="display:none">
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
```

```
body {
    overflow-x: hidden;
    background: #efefef;
 }

/* Toggle Styles */

#wrapper {
    padding-left: 0;
    -webkit-transition: all 0.5s ease;
    -moz-transition: all 0.5s ease;
    -o-transition: all 0.5s ease;
    transition: all 0.5s ease;
}

#wrapper.toggled {
    padding-left: 250px;
}

#sidebar-wrapper {
    z-index: 1000;
    position: fixed;
    left: 250px;
    width: 0;
    height: 100%;
    margin-left: -250px;
    overflow-y: auto;
    background: #333333;
    -webkit-transition: all 0.5s ease;
    -moz-transition: all 0.5s ease;
    -o-transition: all 0.5s ease;
    transition: all 0.5s ease;
    background-color: rgb(52, 102, 154);
}

#sidebar-wrapper-ru2u {
    z-index: 1000;
    position: fixed;
    left: 250px;
    width: 0;
    height: 100%;
    margin-left: -250px;
    overflow-y: auto;
    background: #333333;
    -webkit-transition: all 0.5s ease;
    -moz-transition: all 0.5s ease;
    -o-transition: all 0.5s ease;
    transition: all 0.5s ease;
    background-color: rgb(52, 102, 154);
}
```

```
.productWebshopDiv{
    margin: 20px 0px 0px 38px;
    background-color: white;
    height: 400px;
    width: 250px;
    box-shadow: 5px 5px 5px #888888;
    padding: 0;
}

.productWebshopItemImg{
    display: inline;;
    width: 100%;
    height: 150px;
    margin: 0;
    padding: 10px 0px 0px 10px;
    float: center;
}



.productWebshopItemInfo h1{
    font-size: 25px;
    color: black;
    margin-bottom: 3px;
    font-family: Arial, Helvetica, sans-serif;
}

.productWebshopItemInfo h2{
    font-size: 18px;
    color: blue;
    margin-bottom: 3px;
    text-align: center;
}

.productWebshopItemPrice h2{
    font-size: 28px;
    color: red;
    margin-bottom: 3px;
    text-align: center;
}

.productWebshopItemInfo h5{
    font-size: 20px;
    color: black;
    margin-bottom: 3px;
    margin-left: 40px;
}
```

IX | Extension Activity Diagram