



MASTER THESIS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER COMPUTER SCIENCE

Real-life Artifact Traceability in Augmented Reality

Author:
Jeroen Ceyssens
1436995
Master Computer Science

Coordinators:
Dr. Davy Vanacken
Dr. Gustavo Roveló Ruiz
Mr. Sven Coppers

04 june 2019

Abstract

Within different sectors, artifacts are designed and created to highlight specific information that needs to be communicated to both internal and external stakeholders. Depending on the sector, these artifacts are either stored and displayed on a digital system or are present as objects in the real world. It is easy to display more information on objects on a digital system, however the same cannot be said about objects in a real environment. Any information that these objects fail to portray, has a chance of not being present in the understanding of the model by the stakeholders. For this reason, it is of utmost importance that the required information about these models are present, either in a real or digital environment.

We establish a system that provides artifact traceability information for real objects, using the technique of Augmented Reality to display information on objects in a real environment. Using this system, stakeholders can discover information related to the components and designs of different artifacts. We test the system for the two sectors computer science and architecture, where the evaluated artifact types are adjusted to the use case being presented. For computer science this includes the evaluation of application view designs and for architecture this includes tracing the design changes of a real maquette and understanding the full structure of a project.

After testing the system for both use cases, we found that Augmented Reality is not considered to be beneficial for providing artifact traceability for the computer science sector, while for the sector of architecture it was observed to be very beneficial. Within the sector of architecture, the decision cards format we used for logging the decisions was shown to be less useful compared to within the computer science sector. Based on these findings, a modified version of the decision cards format was proposed, which should successfully provide traceability for both sectors computer science and architecture. Furthermore, guidelines were established on how artifact traceability can be provided in future work for both sectors computer science and architecture.

Acknowledgment

I would like to express my gratitude to Dr. Davy Vanacken, Dr. Gustavo Roveló Ruiz and Mr. Sven Coppers for guiding me throughout the entire process of this research. Without their guidance, it would have been impossible to accomplish the goals set for this thesis. I would also like to give a special thanks to Prof. Dr. Kris Luyten for providing additional feedback during the early stages of this research, and to my parents and girlfriend for all the support they have given me throughout my work. Finally I would like to thank the users that participated in the pilot and usability studies for this research, which helped me analyze the uses of the system.

Dutch Summary

Introductie

In bijna elke sector in het werkveld wordt verwacht dat projectresultaten voldoen aan een aantal einddoelen of -vereisten die door belanghebbenden van het project worden vastgesteld. Om te zorgen dat alle eisen van belanghebbenden verzadigd zijn, moeten ontwerpkeuzes en tussentijdse resultaten worden gecommuniceerd naar de belanghebbenden toe. Op deze manier kunnen deze de huidige voortgang van het project of product evalueren en feedback geven over waar wijzigingen moeten worden aangebracht. Hiervoor wordt de techniek “traceerbaarheid van artefacten” gebruikt, waarmee gebruikers de wijzigingen en vereisten gedurende een project- of productlevenscyclus kunnen traceren

Voorgaand is er al onderzoek gedaan om artefact traceerbaarheid toe te passen voor gebieden zoals informatica, industrieel productontwerp en architectuur. Er is echter één aspect van traceerbaarheid van artefacten dat op dit moment nog steeds grotendeels onontgonnen blijft, namelijk het vermogen om traceerbaarheidsinformatie te verkrijgen voor artefacten die momenteel in een echte omgeving aanwezig zijn. Als gevolg hiervan moeten gebruikers momenteel zoeken naar de informatie over het artefact doorheen verschillende programma’s, systemen of zelfs op fysieke locaties (bijvoorbeeld papieren die rond het kantoor liggen). Dit kan een wijdverspreid probleem zijn voor elke sector, omdat gebruikers hun aandacht worden afgeleid van het werkelijke artefact van interesse, simpelweg om meer informatie erover te vinden. Dit is vooral van belang in sectoren zoals architectuur en productontwerp, waar ze communiceren met hun belanghebbenden met behulp van fysieke eindproducten. Als een belanghebbende meer gedetailleerde informatie over het product wil weten, zal deze informatie gescheiden van het echte model gevisualiseerd moeten worden, waardoor de effectiviteit van de communicatie afneemt en zelfs de interpretatie van het product verstoord kan worden [16].

We lossen dit probleem op door een nieuw tracementingssysteem voor artefacten te bouwen, dat AR (Augmented Reality) gebruikt om traceerbaarheidsinformatie in de echte omgeving te visualiseren voor zowel echte als digitale artefacten. Op deze manier kan informatie over verschillende soorten artefacten worden gepresenteerd aan de belanghebbenden, zonder dat deze op verschillende systemen hoeven te zoeken naar specifieke informatie en zonder dat ze de aandacht verliezen van het object dat ze waren aan het verkennen. We testen deze combinatie van traceerbaarheid van artefacten en AR voor de sectoren van informatica en architectuur, om de haalbaarheid en praktische toepassingen van een dergelijk systeem voor beide velden te testen. We hebben het vakgebied van de informatica gekozen, omdat er op dit moment al verschillende methoden voor traceerbaarheid van artefacten voor dit gebied bestaan. Dit betekent dat we ons AR-systeem eenvoudig kunnen vergelijken met de praktijken die al door andere systemen werden gepresenteerd.

Verder hebben we de sector van architectuur geselecteerd, omdat architecten al gebruik maken van fysieke modellen voor de communicatie met belanghebbenden te ondersteunen. Dit betekent dat we het gebruik van ons systeem kunnen testen om aanvullende informatie voor deze echte modellen te verstrekken.

Hoewel computerwetenschap meestal geen fysieke of 3D-modellen voor verschillende programma’s en toepassingen bevat, is het nog steeds interessant om te evalueren of een AR-systeem nuttig kan zijn om extra informatie te geven voor de use cases die in deze sector aanwezig zijn. Omdat architecten meer gebruik maken van fysieke en 3D-modellen voor communicatie, verwachten we dat we hier meer merkbare voordelen zullen zien voor het gebruik van een AR-systeem voor het traceren van veranderingen binnen artefacten. In de architectuur zijn traceerbaarheidspraktijken echter veel minder gebruikelijk en voorbeelden hiervan zijn moeilijker te vinden, wat betekent dat het ook interessant zal zijn om te onderzoeken hoe deze praktijken door architecten worden geëvalueerd.

Hoe probleem oplossen

Een manier om dit probleem aan te pakken is door gebruik te maken van een techniek genaamd “Augmented Reality”, waarbij mensen de mogelijkheid hebben om informatie in digitaal formaat gevisualiseerd te zien in de echte omgeving. De huidige meest voorkomende vorm van Augmented Reality bestaat uit het dragen van een headset door de gebruiker. Hierbij worden “hologrammen” (= digitale objecten in de omgeving) geprojecteerd vanuit het zicht van de gebruiker, waarmee deze kan interageren door gebruik te maken van “gestures” (= bewegingen met de handen en armen herkend door het systeem). Een voorbeeld hiervan is te zien op Figuur 1 hierbeneden, waarbij de gebruiker kijkt naar hologrammen in de omgeving en hiermee interageert.

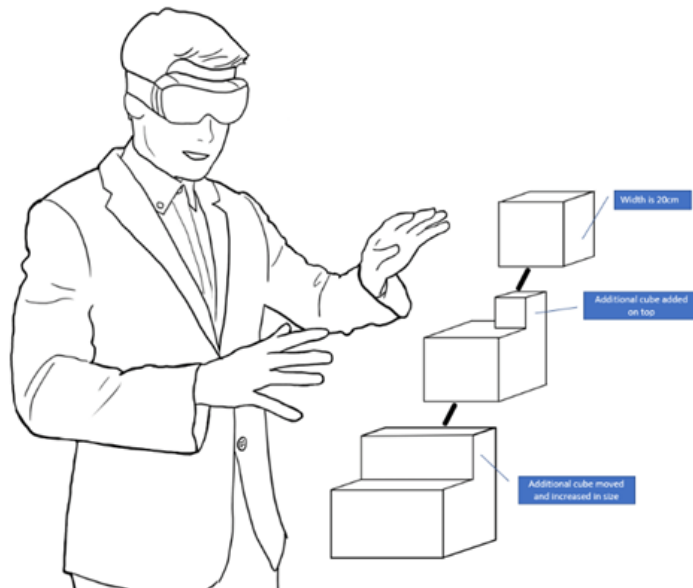


Figure 1: Gebruiker draagt Augmented Reality headset en interageert met hologrammen

Een van de meest voorkomende gebruiken van Augmented Reality op dit moment is het gebruik binnen marketing. Er zijn bijvoorbeeld al verschillende applicaties uitgebracht door bedrijven zoals IKEA, waarbij gebruikers de mogelijkheid hebben om producten te bekijken vanuit hun eigen woonkamer op realistische schaal. Een voorbeeld hiervan is te zien op Figuur 2 hierbeneden waarbij gebruikers de mogelijkheid hebben om stoelen te plaatsen in hun woonkamer op realistische schaal. De stoelen zelf kunnen te allen tijde ook veranderd worden om verschillende designs te kunnen bekijken. Het grootste verschil met ons idee, is dat hier de gebruiker de stoelen bekijkt aan de hand van een applicatie op een smartphone, in plaats van door het dragen van een headset zoals getoond op Figuur 1.

Wat wij hebben behandeld

In onze studie hebben wij gekeken naar hoe wij Augmented Reality kunnen toepassen binnen de sectoren van informatica en architectuur om meer informatie te bieden over zowel echte, als digitale objecten. Deze informatie omvat concreet: de verschillende onderdelen van de objecten, de verschillende versies van de objecten, de beslissingen gemaakt tijdens het design van de objecten en

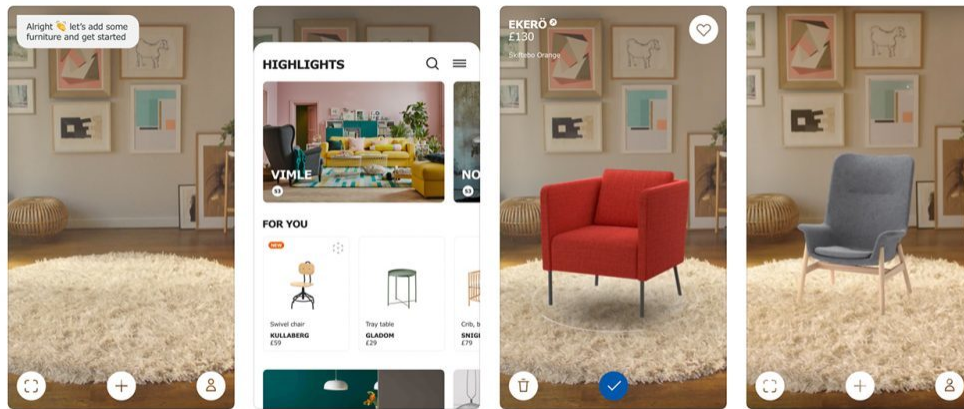


Figure 2: Screenshots van de IKEA Place app (Source: <https://www.chatelaine.com/wp-content/uploads/2018/05/Ikea-Place-App-1000x433.jpg>)

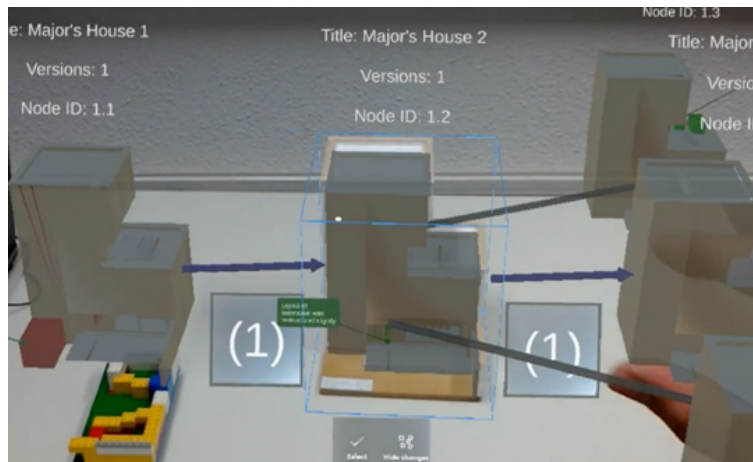


Figure 3: Voorbeeldweergave van versies en annotaties op een maquette

hoe objecten onderling gerelateerd zijn aan elkaar. Op Figuur 3 hierbeneden staat een voorbeeld van hoe deze informatie getoond wordt in het Augmented Reality systeem voor de sector van architectuur.

Om de verschillende onderdelen aan te duiden op de objecten maken wij gebruik van annotaties (= kleine notities) verbonden met specifieke delen van het object. Om de versies van het object tonen, maken wij gebruik van digitale kopieën van het object, die in de volgorde waarin ze gemaakt zijn staan. De beslissingen worden omschreven aan de hand van een titel, beschrijving en extra commentaar en worden vervolgens getoond tussen twee objecten die verbonden zijn met links. En als laatste, om de relaties tussen de objecten te tonen wordt een link gevormd tussen de twee objecten met een label om de exacte relatie te specificeren.

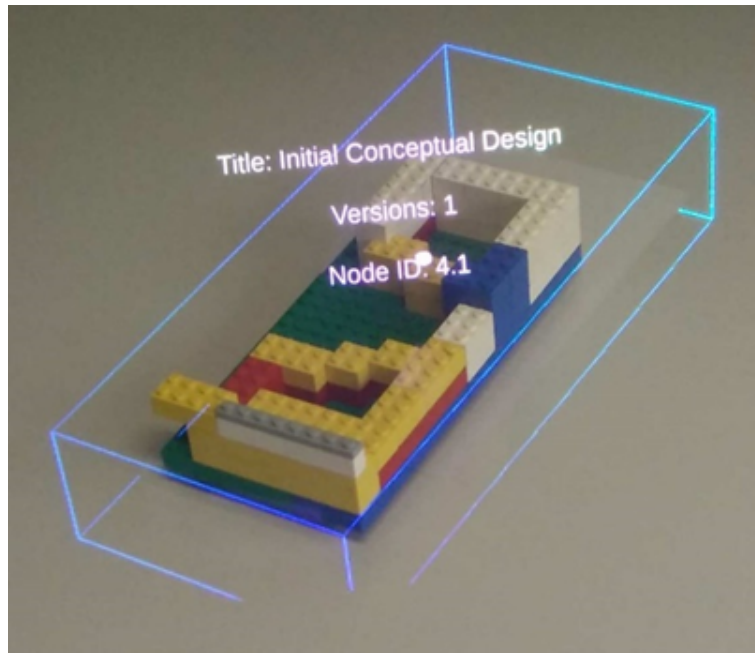


Figure 4: Echt object rechtstreeks getoond als deel van het systeem

Wat we hebben getest

Om de echte objecten te tonen in de visualisatie, hebben we twee aparte methodes voor het systeem getest. Het tonen van de objecten aan de hand van digitale kopieën zoals eerder gezien op Figuur 3 en ze rechtstreeks tonen als deel van het systeem zoals te zien hierbeneden op Figuur 4. Het doel van deze test was om te bepalen welke methode de meeste hoeveelheid informatie biedt voor de gebruikers en hoe gebruikers deze methodes interpretererden.

Verder hebben we ook getest of de gebruikers konden achterhalen hoe de objecten over tijd zijn aangepast, welke van onze visualisatie technieken ze hiervoor hebben gebruikt en hoe ze deze technieken ervaarden. Binnen onze studie bestonden de objecten voor de sector informatica uit verschillende designs voor een nieuwe Android applicatie voor het motiveren van patiënten om te gaan wandelen. Deze designs werden in Augmented Reality getoond van zodra een gebruiker, tijdens het dragen van de headset, de applicatie op een Android gsm bekeek.

Voor de sector architectuur waren er twee projecten die behandeld werden. Het eerste project (Figuur 3) bestond uit een nieuw huis voor een burgemeester en had enkel de verschillende versies van de maquette als digitale objecten. Het tweede project (Figuur 4) bestond uit een project om een nieuw detentiecentrum in Genk, waarbij de verschillende modellen bestonden uit: een weergave van Genk, informatieve posters over het project met afbeeldingen en kleurencodes, een conceptueel design van het detentiecentrum en een uitgewerkt design voor het detentiecentrum. De versies van het conceptueel design omvatte twee echte modellen die rechtstreeks getoond werden in het systeem zonder digitaal kopie.

Wat we concludeerden

Tijdens het onderzoek hebben we gevonden dat binnen de sector van informatica, gebruikers het nuttiger vonden om informatie over de objecten te verkrijgen via een traditioneel 2D systeem. Dit terwijl in de sector van architectuur, de gebruikers juist meer waarde vonden in het gebruik van Augmented Reality voor het bekijken van de 3D objecten.

Vanwege deze reden concluderen we dat, voor het vertonen van informatie over de geschiedenis en de delen van objecten, de beste aanpak bestaat uit het bouwen van een systeem binnen dezelfde dimensie als het object dat bekeken wordt (2D systeem voor 2D schermen, 3D systeem zoals Augmented Reality voor 3D objecten).

Alle gebruikers waren succesvol in het vinden van de informatie over de projecten, waardoor we concluderen dat de praktijken gebruikt in onze visualisaties succesvol waren in het ondersteunen van de gebruikers voor het vinden van de veranderingen, beslissingen en onderdelen van de projecten.

Table of Contents

Introduction	18
1 Concept	19
1.1 Artifact Traceability	19
1.2 Augmented Reality	20
2 Use case	23
2.1 Software GUI Designs	23
2.2 Architectural Building Projects	28
2.2.1 Maquette Versions Traceability	28
2.2.2 Project Structure Understanding	30
3 Related Work	37
3.1 Artifact Traceability	37
3.1.1 Impact Analysis	37
3.1.2 Coverage Analysis	38
3.1.3 Derivation Analysis	38
3.2 AR Use Cases	40
3.2.1 Object Placement	41
3.2.2 Annotations	42
3.3 Artifact traceability in AR	45
4 Implementation	47
4.1 General overview	47
4.1.1 NodeJS Server	47
4.1.2 AR System	48
4.2 AR Hardware	48
4.3 Artifact Links and Versions Visualization	50
4.3.1 Digital Artifact Versions Visualization	51
4.3.2 Real Artifact Versions Visualization	52
4.4 Annotations and Decision Cards Integration	54
4.5 Real Object Integration	56
5 Computer Science Study	61
5.1 What to evaluate	61
5.2 Study Setup	62
5.2.1 Application scenarios	64
5.2.2 Questionnaire	66
5.3 Observations and Analysis	66
5.3.1 Artifact Traceability User Methodology	66
5.3.2 Practicability of Artifact Traceability In AR	69
5.3.3 AR Limitations	69
5.3.4 System Usability Scale Evaluation	70
6 Architecture Study	73
6.1 What to evaluate	73
6.2 Study Setup	74
6.2.1 Maquette Versions Traceability	75
6.2.2 Project Structure Analysis	76
6.3 Observations and Analysis	78
6.3.1 Artifact Traceability User Methodology	79

6.3.2	Real Artifact Interpretation	81
6.3.3	AR Limitations	82
6.3.4	Questionnaire Evaluation	82
7	Discussion	83
7.1	Findings Interpretation	83
7.1.1	General Artifact Traceability Guidelines	84
7.1.2	Artifact Traceability Guidelines for software GUI's	85
7.1.3	Artifact Traceability Guidelines for Real models in Architecture	85
7.2	Limitations	86
7.2.1	Hardware Limitations	86
7.2.2	Experimental Limitations	87
7.2.3	Real Model Interpretation	87
7.3	Future Work	87
7.3.1	Collaboration Possibilities	88
7.3.2	Creation of Traceability Means	88
7.3.3	Full Scale System Usability	89
7.3.4	Consideration of Physical Environments	89
8	Conclusion	91
	References	95
	Appendices	97
A	Walk With Me Screenshots	97
B	Major's House Project	99
C	Conceptual Design Detention Center Project	101
D	Consent Form	103
E	MuiCSer Explanation	105
F	User Steps Training Phase	107
G	User Steps Computer Science	109
H	Guidor Steps Computer Science	111
I	Computer Science Questionnaire	113
J	Computer Science Study Questionnaire Results	115
K	User Steps Architecture	117
L	Guidor Steps Architecture	119
M	Architecture System 1 Questionnaire	121
N	Architecture System 2 Questionnaire	123
O	Architecture Study Questionnaire Results	125

List of Figures

1	Gebruiker draagt Augmented Reality headset en interageert met hologrammen	6
2	Screenshots van de IKEA Place app (Source: https://www.chatelaine.com/wp-content/uploads/2018/05/Ikea-Place-App-1000x433.jpg)	7
3	Voorbeeldweergave van versies en annotaties op een maquette	7
4	Echt object rechtstreeks getoond als deel van het systeem	8
5	Artifact traceability practice to provide links between annotations and objects	20
6	Example decision card for the artifact portrayed in Figure 5	21
7	User wears an AR head mounted display to look at digital objects in his environment	22
8	The MuiCSer process framework for User-Centered Software Engineering [22]	24
9	Example prototypes for a NewsWizard interface with (a) the low-fidelity prototype and (b) the high-fidelity prototype [22].	25
10	Overview of Scene 1 in the system	26
11	Overview of Scene 2 in the system	27
12	Overview of Scene 3 in the system	27
13	Overview of Scene 4 in the system	28
14	Example infographic panel of an architectural project from Archmedium (Source: https://www.flickr.com/photos/archmedium/11082543946/)	29
15	Different versions of the physical maquette for the major's house project	30
16	Physical Lego maquette for the first version with (a) the front view, (b) the top view and (c) the original model version in Blender.	31
17	Physical Lego maquette for the second version with (a) the front view, (b) the top view and (c) the original model in Blender.	32
18	The conceptual design with (a) the models linked to it, (b) the versions contained within and (c) the versions for the second real model.	33
19	The rendered version of the model with (a) the model linked to the overview of Genk and (b) the versions contained within.	33
20	The infographic posters with (a) the overview of both posters, (b) the versions of the front side and (c) the versions of the back side.	34
21	An overview of Genk visualized in AR	35
22	Example Requirements Traceability Matrix (Source: https://project-management.com/requirements-traceability-matrix-rtm/)	38
23	Design Rationale editor [11]	40
24	Git Flow Diagram of a project workflow (Source: http://socialsharing.info/git-flow-diagram/gitflow-the-easy-release-management-workflow-git-flow-diagram/)	41
25	Screenshots from the IKEA Place app (Source: https://www.chatelaine.com/wp-content/uploads/2018/05/Ikea-Place-App-1000x433.jpg)	42
26	(a) Letters and brochures from Strata showing a table in AR (Source: https://ar.strata.com/home-2/industries/), (b) Brochure from Northern Lighting using Augment to display models (Source: https://www.augment.com/blog/5-companies-using-augmented-reality-print-campaigns-brochures/)	43
27	Mobile Augmented Reality visualization of hotel complex in Finland 2011 [2]	43
28	Use of mobile Augmented Reality in a shipyard with (a) the localization of pipes in a workshop, (b) additional information about a specific pipe and (c) the contents of the shelf displayed on a tablet [9].	44
29	Copenhagen airport Mobile Augmented Reality navigation system [2]	45
30	Visualization of building progress in <i>ConstructAide</i> [25]	46
31	Screenshots of "Augmented Reality Solutions for Construction Inspection"[23]	46
32	General overview of the implemented system	47
33	Unity Editor showing a scene from our system	48

34	Different AR headsets with (a) the First edition of Microsoft Hololens, (b) the Meta 2 and (c) the Magic Leap One (Sources: https://www.extremetech.com/wp-content/uploads/2015/10/HoloLens.jpg , https://dab1nmslvvntp.cloudfront.net/wp-content/uploads/2016/03/1457324978meta2.jpg , https://www.spiria.com/sites/default/files/blog-2017/magic-leap-one-1.jpg).	49
35	Visualization of linked objects in AR with (a) the user having the ability to show the links for model and (b) the link between objects that follows from pressing on the button.	51
36	Visualization of different types of linked objects in AR.	52
37	Branching visualizations of artifact versions with (a) the use case where two versions are made in parallel, but only one was selected and (b) the use case where two versions are made in parallel and both were selected and merged together.	53
38	Versions visualization for a real artifact with (a) the digital object being aligned with the model, (b) the first version being aligned with the object, (c) the second version being aligned with the object and (d) the version changes being shown in red and green colors.	54
39	Annotations support for artifacts with (a) multiple annotations giving context on digital artifact parts and (b) an annotation providing context for a real artifact. . .	55
40	Decision cards visualization with (a) the overview of buttons underneath links, (b) overview of available decision cards for button and (c) decision cards being shown between the models.	57
41	Application view detected by the system using Vuforia with (a) the clipping plane not showing the full screen close-by and (b) the detected screen fully shown in front of the smartphone.	58
42	Digital object aligned with a real model	58
43	Inside of the model reveals itself when users approach the model.	59
44	A real model as part of the system with (a) the information about a real object being shown, (b) a real object having been selected and (c) the visualization of how the model is interpreted by the system.	60
45	Computer Science Study setup	63
46	Training phase with (a) a cube as the final model and (b) the versions of the model. .	64
47	Latin Square that matches the participant with the order that they will cover the scenes	65
48	Comparison graph of needing decision cards vs. instantly using decision cards	67
49	(a) User looking at the decision card and versions of Scene 1, (b) User looking at the second version of Scene 1.	67
50	User looking at the decision card and versions of Scene 3	68
51	(a) SUS results for every participant and the mean SUS score, (b) Mean SUS score for every question.	71
52	Architecture Study setup	74
53	The conceptual design versions with (a) the full versions visualization and (b) the versions of the real model	78
54	User looking at the decision cards but not being able to see the changes on the model above	79
55	User placing the poster closer to the conceptual designs to help in tracing the changes	81
56	Scene 1, the option to start a walk with (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version.	97
57	Scene 3, the statistics visualized for walks with (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version.	97
58	Scene 4, feedback on a walk and sharing the experience with a community (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version. . . .	98

59	Scene 2, the progress shown during the walk with (a) the low-fidelity prototype, (b) the first high-fidelity prototype, (c) the second high-fidelity prototype and (d) the final version.	98
60	Physical maquette for the project of the major's house with (a) the front view and (b) the back view.	99
61	The conceptual designs for the detention center project	101
62	The textured designs for the detention center project	102
63	The infographic posters for the detention center project	102
64	SUS results for every question and user for the Computer Science study	115
65	Questionnaire results for every question and user for the Architecture Study System	1125
66	Questionnaire results for every question and user for the Architecture Study System	2125

Introduction

In almost every sector in the workfield, project results are expected to satisfy some end goals or requirements that are identified by stakeholders. To ensure all stakeholder requirements end up satisfied, design choices and intermediate results need to be communicated. This way, stakeholders can evaluate the current progress of the project or product and give feedback on where changes need to be applied. For this, a technique called “artifact traceability” is used, which allows users to trace the changes and requirements throughout a project or product lifecycle.

Berenbach et al. mentioned the three most common needs for traceability are *coverage analysis*, *impact analysis* and *derivation analysis*. They also mentioned that neglecting any of these needs could result in late delivery, overspending on resources and even complete project failure [7]. For this reason, research has already been conducted to apply artifact traceability for fields like computer science, industrial product design and architecture.

However, there is one aspect of artifact traceability that currently remains widely unexplored. This is the ability to acquire traceability information for artifacts currently present in a real environment. Consequently, right now users need to search for the information about the artifact through different programs, systems or even in physical locations (e.g. papers lying around the office). This can be a prevalent problem for every sector, since users will have to divert their attention away from the actual artifact of interest, only to discover more information about it. This is specifically of relevance in sectors like architecture and product design, where they communicate with their stakeholders using physical end products. If a stakeholder wants to know more detailed information about the product, they will have to display this information separate from the real model, which reduces the effectiveness of communication and can even interfere with the interpretation of the product [16].

We solve this issue by building a new artifact traceability system, which uses AR (Augmented Reality) to visualize traceability information in the real environment for both real and digital artifacts. This way, information about different types of artifacts can be presented towards the stakeholders, without them needing to search for specific information across different systems. This also ensures that the stakeholders do not lose focus on the object they were exploring in the first place. We test this combination of artifact traceability and AR for both demographics computer science and architecture to test the feasibility and practical uses of such a system for both fields.

We selected the field of computer science since there are already several common artifact traceability practices for this field, meaning we can easily compare our AR system to the practices already presented within other systems. Meanwhile, we selected the field of architecture due to architects already making use of real models for providing stakeholder communication. This means we can test the use of our system for providing additional information for these real models.

While computer science does not usually include physical or 3D models for different programs and applications, it will still be interesting to see whether an AR system can be considered to be useful for providing additional information for the use cases present in this sector. Since architects already make more extensive use of physical and 3D models for communication, we expect them to have more noticeable benefits from using such an AR system. However, within architecture traceability practices are a lot less common, meaning it will also be interesting to explore how these practices are evaluated by the users.

Within this research, we will first discuss the concepts that we cover and what techniques we apply for these concepts, to highlight what we need to provide in our system to achieve our goals. Next we discuss the specific use cases that we cover within our research for the fields of computer science and architecture. After the use cases, we will discuss the works that have already been carried out for artifact traceability and AR and how these works are related to the ones we provide in our research. Afterwards, we give a description of the system and the features that we have

implemented, to which later we will discuss for each sector separately, what study we have carried out and what we have observed within these studies. Finally, we will discuss the findings of both studies by establishing guidelines on how to provide artifact traceability and by addressing the shortcomings that are still present within the current system and the used techniques.

1 Concept

In this section we introduce the concept of our research, in relation to the two topics “Artifact Traceability” and “Augmented Reality”. We first cover the topic of “Artifact Traceability” since it also highlights the goal of our current research, compared to “Augmented Reality” which can be seen more as a means to achieving this goal.

1.1 Artifact Traceability

According to the Cambridge English Dictionary, the definition of “traceability” is “the ability to find or follow something” or “the ability to discover information about where and how a product was made”¹. The definition of “artifact” is defined within the same dictionary as “an object that is made by a person, such as a tool or a decoration, especially one that is of historical interest”². Combining the two definitions, we can define the meaning of “artifact traceability” as “the ability to discover information about where and how a particular artifact was created”.

There are several traceability practices that we can enforce, but we specifically chose for tracing the design rationale of artifacts. We pursue this type of traceability, since we want to make a system that focuses on the improvement of communication with stakeholders about the design choices of an artifact. The artifact traceability practices we use within our implemented system are based on those of an already existing system called “Helaba”, which was introduced by Gutierrez Lopez et al. Their system was used to trace the design rationale of User-Centered design processes in software and included several guidelines on how to approach this. However, we attempt to apply and extend upon these practices, so they become applicable within multiple sectors as well [21]. The guidelines mentioned by the research included the following:

- Provide a visual connection between artifacts and communication related to the artifact.
- Provide a method for sharing and receiving feedback from team members and stakeholders of the project in a shared workspace.
- Provide awareness over artifact evolution and previous design decisions.

We will mostly focus on providing the first and last guidelines of the project, since our goal is mainly on visualizing effective ways to communicate design choices to stakeholders (in section 4 we define how we have implemented these guidelines). The second guideline, providing a shared workspace, is more relevant for teamwork collaboration when the system is deployed and actively in use by teams. However, for providing analysis on the project by stakeholders, it is currently not required.

To communicate information related to an artifact, within our system we provide the concept of “annotations”. Annotations are text boxes that contain information about the artifact and can be linked to the artifact using a connecting line. An example of this can be seen on Figure 5, where annotations are created and linked to models to explain more information about the different components. Note that the annotations point toward the part that they provide more context on, which allows users to see the visual connection between the explanation and the model.

Instead of only linking the annotations with the models, the models themselves can also be linked together using arrows to visualize the order in which the models were created (see Figure 5). We call these different models the “versions” or “iterations” of the final product, where each of them portrays a representation of the product at a specific point in time. When multiple versions are present for a specific product, some of these versions may no longer follow each other sequentially

¹Cambridge Business English Dictionary, 2018, “traceability”, <https://dictionary.cambridge.org/dictionary/english/traceability>

²Cambridge Business English Dictionary, 2018, “artifact”, <https://dictionary.cambridge.org/dictionary/english/artifact>

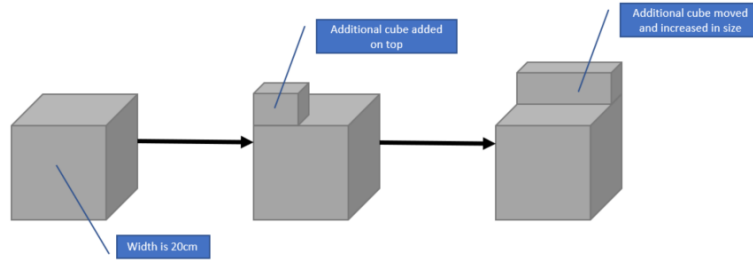


Figure 5: Artifact traceability practice to provide links between annotations and objects

but will instead have been made simultaneously with each other. In such a case, the links would have to split into different paths showing what versions had been built simultaneously and which versions have followed from each of these parallel models. This is the concept of “branching”, which we discuss further in section 3.1.3.

Using a similar method, an artifact can also be linked together with models that are not different versions. An example of this are models that are of relevance to understanding the structure of the artifact. For an industrial product, such models can include manual papers, different components of the product, tools that need to be used together with the product, etc. The visual links between the artifact and these additional models, would at least have to specify a label or should consist of different types of links, to explain the relation that exists between these models. For example in Figure 5, the arrows between the different versions can be considered as links with the label “follows from” or “is a sequence of”.

The aforementioned examples of visualizing links between objects, object versions and annotations are example practices of the first guideline “provide a visual connection between artifacts and communication related to the artifact.” and the the first part of the third guideline “provide awareness over artifact evolution”. However, the second part of the third guideline “providing awareness over previous design decisions”, is also relevant for us to cover in the concept of our research. For this, *Helaba* applied the use of a technique called “decision cards”, which was introduced by Gutierrez Lopez et al. to capture the design rationale for supporting awareness and traceability of the User-Centered design process in software. Decision cards provide a lightweight format for logging the decisions during the design process, since users can easily add additional notes and commentary to further emphasize on the meaning of the decision. An example decision card is shown on Figure 6, which logs the decision between the first and second versions of Figure 5. The fields of a decision card generally include: a title, a description, the people involved in making the decision and extra commentary, artifacts or discussions involving the decision [20].

Since they are so lightweight, using them for other types of projects and sectors seems to be feasible as well. However, before we can consider them to be fully usable for other sectors, tests need to be conducted where they are actively applied for different types of projects within different sectors. Since we also need to provide artifact traceability for multiple sectors within our research, we will conduct the tests required to analyze the use of the decision cards for these sectors as well.

1.2 Augmented Reality

To ensure we can visualize the traceability information about a specific real object without having the users shift focus, we use a technique called AR (Augmented Reality). This is a visualization and interaction technique that overlays digital information on a real-life environment. The visualization consists of 3D objects and windows floating in physical space that users can look at or even

Figure 6: Example decision card for the artifact portrayed in Figure 5

manipulate. There are several different ways AR can be visualized (more information on this can be found in section 3.2). We specifically use the technique of providing a head mounted display that the user needs to wear to visualize the digital objects from the user’s point of view. Furthermore, this method allows us to portray relevant information about any of the physical objects in the environment around the user, which is also the use case we focus on within our research.

Aside from being able to overlay information about specific objects in the environment, other benefits of using head mounted AR include:

- Being able to visualize concepts that would not be clear in 2D due to spatial reasoning and physical screen limitations.
- High level interactions due to large degrees of freedom by use of gestures.
- Enhanced remote collaboration and communication by, for example, allowing local users to perform physical tasks under guidance of remote users [37].

We will mostly make use of the first and second additional benefits of head mounted AR. The first allows us to enforce the ability to understand the artifact and project structures, by visualizing the 3D aspects more clearly than on traditional systems and by making full use of the space around the user to visualize the artifacts and their relations. Meanwhile the second benefit makes it possible for users to freely explore and manipulate the artifacts using gestures, which can support the user in exploring all their properties. The third benefit is an interesting use case for head mounted AR within remote settings, however we keep the focus within this research on communication with stakeholders present in the same location.

To apply the concept of artifact traceability for such a visualization technique, the models that are shown in the environment will need to use the same practices as seen in section 1.1. An example of this is shown in Figure 7, where a user is wearing a head mounted device to explore different artifacts that are linked together and annotated. These artifacts can include both real and digital models, since we want to be able to provide traceability for both types of artifacts and since multiple real models can also be linked to multiple digital models.

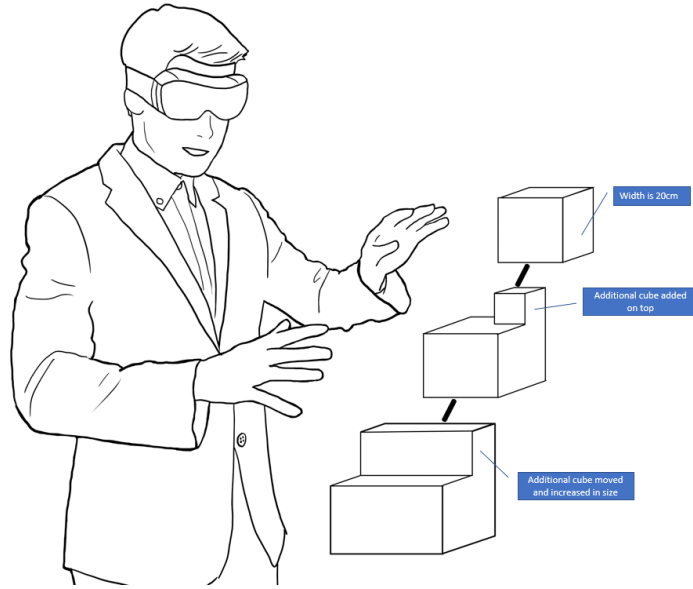


Figure 7: User wears an AR head mounted display to look at digital objects in his environment

To provide information about real models, there is also a need to explore how these models will be represented within the AR system. A first option is to present the model itself without any indication of it having a digital counterpart, where all the information about the model is shown directly on top of it. In this case, there is a more direct link between the digital objects and the real environment. A second option for visualizing the real model, is by presenting a digital copy of the object that represents the model within the system and can be interacted with. All the traceability information will be visualized on top of this digital copy instead, providing clearer distinction between what is real and what is digital. We explore and test both of these concepts within our research, so we can discover which practice is best for providing traceability information for the real objects within AR.

2 Use case

To demonstrate the use of AR for providing artifact traceability, we need to consider several possible use cases within multiple sectors. We focus within our research on two main sectors, computer science and architecture. However, we cannot focus on all the use cases for both sectors, so we need to limit ourselves to only providing use cases that are most relevant within these sectors when it comes to artifact traceability. For this reason, we select for computer science the use of software graphical user interface designs and for architecture we select the use of architectural models for building projects. We explain these use cases as follows.

2.1 Software GUI Designs

Since we focus within our research on extending the use and guidelines of *Helaba* with Augmented Reality practices, we want to test how our system fares with a similar demographic. This way, we can find whether employing AR provides an improvement compared to *Helaba* for the traceability of artifacts. The study of Gutierrez Lopez et al. focused on the demographic of User-Centered Design Processes of software, meaning this is the demographic we will also have to cover within our research. This implies that we need to test our system with models from a software program or application that applies User-Centered Design practices [21]. We can describe these User-Centered Design practices as follows.

User-Centered Design practices generally consist of including the user throughout the several stages of software design. This way, stakeholder input can be received early in the software design lifecycle when changes can still be applied at a low-cost. To support developers in applying these practices, frameworks have been established to identify the stages that need to be followed and the artifacts that need to be created within the project lifecycle. An example of such a framework is the MuiCSer process framework (shown on Figure 8), which was established by Haesen et al. Between every stage of this framework, designers are required to test their artifacts with either stakeholders or end users, so they can evaluate whether their current assumptions are correct. We will first explain the different parts or stages of this framework, so afterwards we can highlight the parts that we focus on in our research [22].

The first two stages of the framework include the analysis of the user needs for the new (or legacy) system and building a structured interaction analysis to specify the application requirements. During the structured analysis phase, developers are required to refer to the defined user needs, to analyze what application requirements are necessary to cover these needs. The following stage is about constructing low-fidelity prototypes, which are simple mockups to highlight the different views and interactions with the system. These mockups are created based on the previously specified application requirements, to evaluate how these requirements will be visualized within the system. An example low-fidelity prototype can be seen on Figure 9a. After creating these low-fidelity prototypes, lab tests are conducted to evaluate and test based on the user input, whether the project team has made the correct assumptions about the interactions provided by the system [22].

The following step includes building high-fidelity prototypes, which are realistic simulated versions of the system (Figure 9b) used to test the feasibility and overall interpretation of the system. This stage is most often followed by a field test with end users, so designers can observe how the users interpret and use the system in question and whether there are still faults to be careful about when turning it into a full system. In the final stage of MuiCSer, the full system is built using the knowledge acquired from the previous stages to ensure users will experience the system positively and effectively [22].

Since our research is focused most on the traceability of artifacts, more specifically the changes to specific artifacts and the decisions within these changes, we focus in our use case for the most

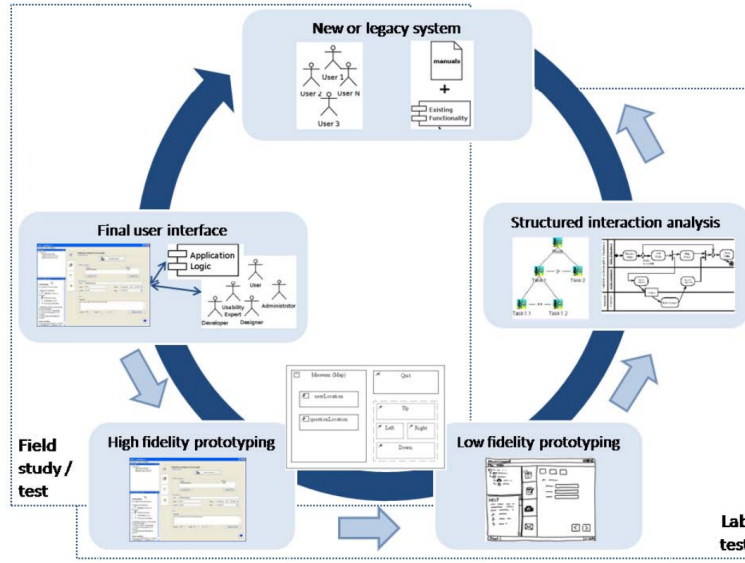


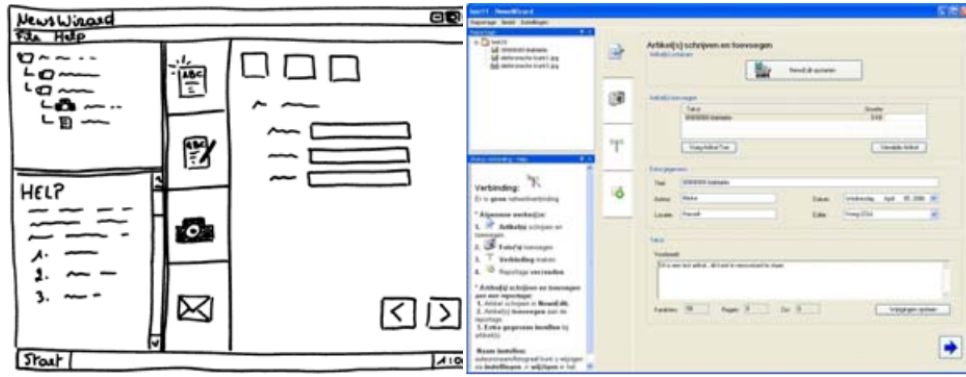
Figure 8: The MuiCSer process framework for User-Centered Software Engineering [22]

part on the low-fidelity prototype, high-fidelity prototype and final version stages of the MuiCSer framework. The idea we want to employ with this, is that users can look at the screen of a smartphone or computer that displays an application or program that is in development or has already been developed, to find the changes that were most important within the design of this application or program.

For example, users can look at a finished view of the NewsWizard interface in Figure 9b and discover the changes that were made for this particular view compared to Figure 9a. This way, instead of only allowing users to see the phases of the application separately, they can now explore the full evolution for each application view. Consequently, stakeholders will be able follow up with the developers about specific features that were included or excluded in later design phases compared to previous design phases. In a similar method, new team members can also be brought up to date about the reasons certain features were implemented or considered within the system, making it faster for them to be able to adjust to the practices within the new work environment.

The idea for the use of AR within this system would be primarily to provide users the ability to “grab” a screen away from the monitor and explore its artifacts and important changes. This way the full system views can be explored in more tangible ways than before, which could help users in finding the changes more efficiently. There are also no more physical limitations like screen size, making it so users can explore the full evolution of the artifacts without having to scroll through a system. Like seen in Figure 9a, artifacts during the design process can also include mockups that are either drawn on paper or visualized using a drawing tool. Here gestures in AR can be used to simulate the feeling of holding and interacting with these paper mockups, while still allowing users to see the links between these mockups and their later counterparts.

Originally, for real physical models of low-fidelity prototypes, no additional information on these prototypes could be provided within the real environment. However, using AR to visualize this information makes it so users can be made more aware of the links and relations existing between these artifacts. To further emphasize this, our previously mentioned situation can be reversed where instead of grabbing an application or system from a screen, users can look at the paper mockups of the design process to find how it has influenced the final version of the system. However, this is



(a)

(b)

Figure 9: Example prototypes for a NewsWizard interface with (a) the low-fidelity prototype and (b) the high-fidelity prototype [22].

a matter of perspective since both methods would be able to be applied simultaneously and would most likely have similar outcomes. It is only the entry point for the artifacts that is different, the project structures would still remain the same,

An example project that applied the practices of the MuiCSer framework, is an application called “WalkWithMe”. The goal of this application is to help motivate patients that suffered from stroke to rehabilitate themselves by going for walks. Since the application used the MuiCSer framework for the design process, it already contained low-fidelity prototypes (mockups), high-fidelity prototypes (simulated prototypes) and an implemented final version.

According to the designers, there were two decisions made throughout the design process that were specifically of importance. The first important decision was that the designers suddenly had to switch from landscape to portrait mode between the high-fidelity prototypes and final versions. This happened because they wanted to pursue gamification aspects by putting everything into landscape mode, to motivate the users to walk. However later in the design process, they found that using the application in landscape mode would make the application harder to use during walks. Consequently, a lot of views had to be switched over to portrait mode within the final versions. A second decision that was of importance throughout the application, was that at first the designers assumed the application would revolve around setting the amount of steps for a walk. However, after the high-fidelity prototyping phase they realized that setting the amount of time that a user wants to walk, was more practical and higher evaluated by the test participants.

Using the screenshots provided by the application (see Appendix A), four different scenarios can be constructed to be used within our system. Each scenario consists of a final application view, its own respective prototypes (low-fidelity and high-fidelity) and a series of decision cards and annotations used to explain the components and the made decisions during the design phases. The two important decisions are already examples of which decision cards are created, together with some other significant changes that are relevant to mention for the particular views. Other information about the different parts of the designs will be visualized using the aforementioned annotations. Using this setup, we arrive at the following four scenarios:

Scene 1 (Figure 10): The user starts a new walk and sets up how far or long they want to walk. There are three decision cards in total for this scene, one between the first two versions and two between the last two versions. The first decision that was taken was about changing

the distance selection to setting the amount of steps instead, while visualizing how many kilometers are equal to this amount of steps. The last two decisions are the previously mentioned landscape to portrait switch and the change from amount of steps to amount of time to walk. The first annotation in the scene mentions for the second version that the character can be selected by the user in the settings. The other two annotations in the final version explain that this application view is also the homescreen of the application and that the weekly goal can be set in the settings.

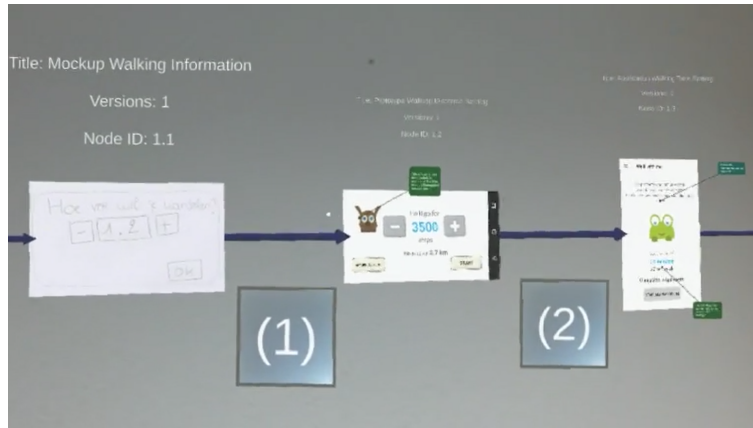


Figure 10: Overview of Scene 1 in the system

Scene 2 (Figure 11): The user is currently walking and can see the current progress within the application, either through a moving character in the game or by visualized statistics about the walk. For this scene two high-fidelity prototypes are made to highlight that the character can move along with the user and that statistics show the current progress. There are three decision cards in total in this scene, one between the low-fidelity and high-fidelity prototypes and again two between the high-fidelity prototypes and the final version. The first decision card mentions that the current progress was numerically added to the visualization, since this is more useful for the users to know exactly how many steps they still have left. The last two decisions are again the previously mentioned landscape to portrait switch and the change from amount of steps to amount of time to walk. In total, two annotations are present in the scene where the first explains for a high-fidelity prototype that the character will move according to the distance that was walked by the user. The second annotation mentions for the final version that the amount of steps will still be mentioned above in the final version.

Scene 3 (Figure 12): The user can see the statistics for a specific walk and can explore them for other walks as well. For this scenario there is only one decision card, which is between the first and second versions. This decision card mentions that, instead of the character text box containing the statistics, a separate view was created to let users navigate through a history of walk statistics. There are also two annotations present, where one explains for the first view that the character mentions the statistics after the walk, while the other mentions for the second view where the user can select the day (at the top of the screen).

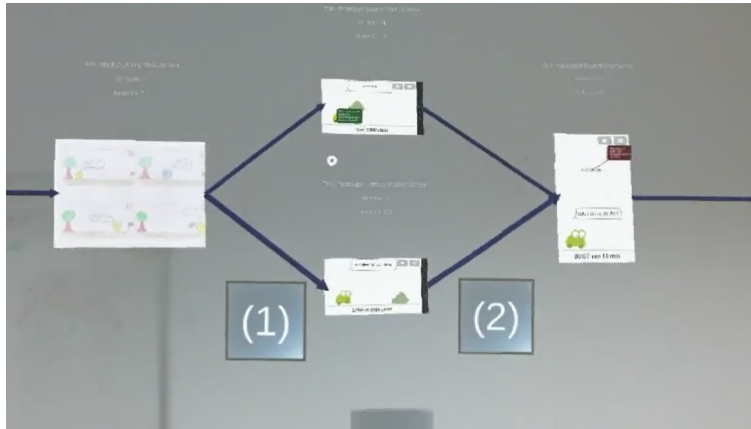


Figure 11: Overview of Scene 2 in the system

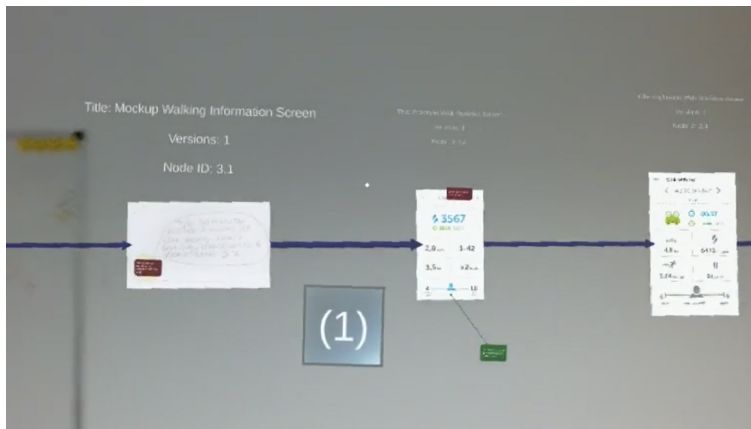


Figure 12: Overview of Scene 3 in the system

Scene 4 (Figure 13): The user is presented with three versions that are all about providing feedback about a walk, but each version presents it in a different way. There are three decision cards present within these versions. The first decision card is between the first two versions and mentions that users can add extra commentary to the walk exhaustion selected within the first version, so that the user and the doctor can follow up more closely on the current rehabilitation progress. The second decision card which is between the last two versions, explains that users can now share their walking experience with their friends, so they can keep each other motivated to walk more often. Finally, the third decision card mentions the switch from landscape to portrait mode between the final two versions, like what we have seen previously. In this scene, there is also only one annotation present which mentions that the user can select an avatar in the settings to represent him/herself in the community within the final version.

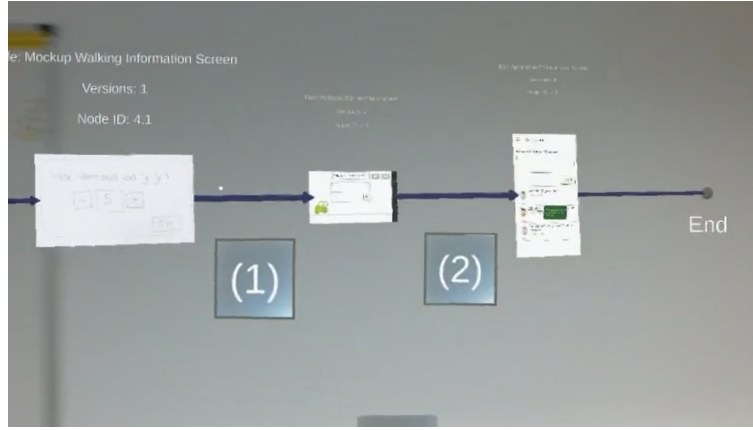


Figure 13: Overview of Scene 4 in the system

2.2 Architectural Building Projects

Within the field of architecture, it is a common practice to create miniature physical models of projects for stakeholder communication. These physical miniature models are often called “maquettes”. Compared to being able to see the models on a computer screen, real-life replicas allow stakeholders to quickly get a grasp of the different perspectives and 3D properties of the model. However, there is a limit to how much information can be provided using only a physical maquette. For this reason, architects make it a custom to add infographic panels that are placed close to the maquette, which provides more detailed information about the project to the stakeholders. An example of such an infographic panel can be seen on Figure 14. Typically these infographic panels include high resolution image renders of how the project would look like in real-life, detailed information about the layouts of the project and general keywords to explain different parts of the project [16].

A problem with this approach is that stakeholders need to divert their attention away from the model to acquire more information, since they cannot look at both the infographics and the maquette simultaneously. This leaves a lot of the translation of the displayed information, to the applicability on the real model, up to the stakeholders. However, without these infographics, a lot of information about the physical maquette goes completely lost. There are no descriptions about the different parts of the maquette and no renders are available to visualize how the model would look like in real-life. In other words, a lot is left to the imagination of the stakeholders to fill in the information gaps. We attempt to solve these issues by using Augmented Reality to overlay the required information to provide artifact traceability for real maquettes. We have two separate use cases that we cover for this: maquette version traceability and project structure understanding. Both use cases allow us to test the feasibility of artifact traceability in Augmented Reality for architectural practices, but they each cover a different aspect on how the traceability can be provided.

2.2.1 Maquette Versions Traceability

Like we mentioned previously, architects create maquettes to communicate design decisions of projects to stakeholders. During a project’s lifecycle, architects often iterate over the layout design and materials used within the construction. However, due to the high cost in time and resources, they often only make one maquette which is used to represent the final version of the model. This means that all the previous versions and decisions go lost in time. Sometimes architects can even

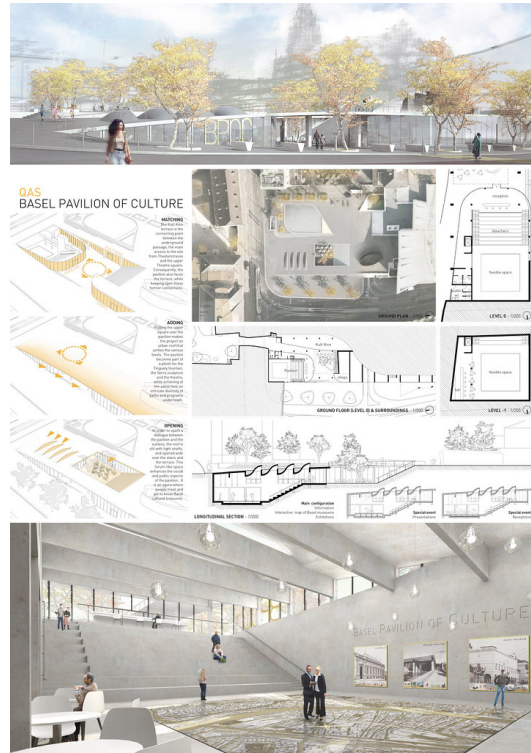


Figure 14: Example infographic panel of an architectural project from Archmedium (Source: <https://www.flickr.com/photos/archmedium/11082543946/>)

decide to not include specific parts of the model due to resource, time or feasibility constraints. To maximize stakeholder communication for the design process of the project and the maquette, previous versions need to be explorable and all important components of the maquette should be visualized.

By using AR for this use case, we can make the versions and changes physically appear in front of the user, similar to how maquettes are shown in front of stakeholders. Here, it would be interesting if one of the digital copies of the maquette is aligned with the real model and linked together with the other versions, similar to the traceability guideline techniques seen in section 1.1. Using this method, annotations can also be provided to overlay information on top of the digital copies to ensure all parts of the model versions are understood. Therefore, the best of both worlds, visualizing a physical model for spatial reasoning and providing additional information using a digital format, can be achieved. Using this method, stakeholders will also no longer have to divert their attention away from the model to be able to discover more information about the different components and decisions within the model.

To portray this use case more clearly, we will look at an example of an architectural project with a physical maquette. The goal of this example project is to construct a new house for the major of a city. In this project, a physical maquette was constructed to portray the overall structure of the building (see Appendix B). However, before the maquette was built, several digital versions of the model were created, resulting in versions that can no longer be explored by the stakeholders. These different versions are shown on Figure 15, where the techniques we previously mentioned in section 1 have been applied (versions appearing with the model, overlaying information on aligned copy and use of annotations and decision cards). The versions of the maquette present within this

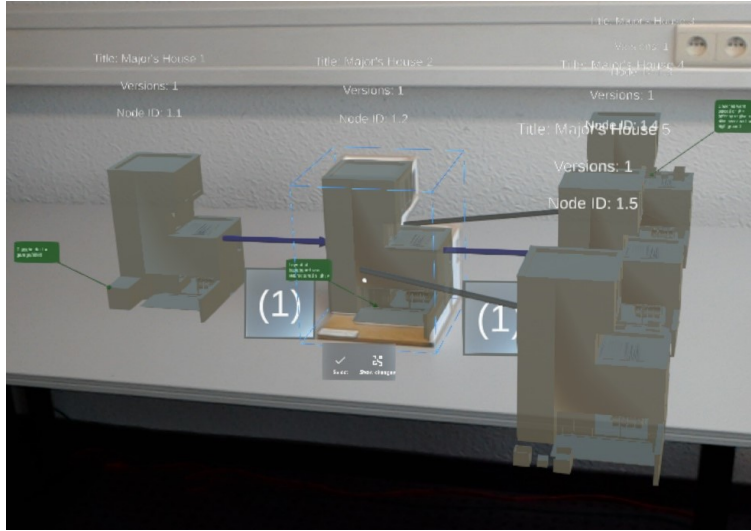


Figure 15: Different versions of the physical maquette for the major's house project

project can be explained as follows:

- Version 1 (left):** A starting point for the maquette design was made and a placeholder was added in front of the building, in case a garage would have to be constructed in a later phase. An annotation explains the block in front of the building is representative as the placeholder of the garage.
- Version 2 (middle):** The placeholder for the garage was removed and the baseboard of the maquette design was restructured slightly. This version follows sequentially from version 1 and a decision card mentions the removal of the placeholder for the garage with the mentioned reason that the garage will no longer be part of the design. Furthermore, the version also has an annotation that mentions the restructuring of the baseboard.
- Version 3 (top-right):** Extra couches were placed on the balcony of the building and an annotation mentions that they were placed here to give a nice overview from a high ground. This model follows sequentially from version 2 but is an alternative to version 4 and 5.
- Version 4 (middle-right):** Extra couches were placed in front of the veranda and a decision card mentions the placement of the couches here with extra commentary that the couches will not be included on the physical maquette. This model follows sequentially from version 2, but is an alternative to version 3 and 5 and is also the final version that was decided on for the model.
- Version 5 (bottom-right)** Extra couches were placed on the lawn and no annotations or decision cards are present. This model follows sequentially from version 2, but is an alternative to version 3 and 4.

2.2.2 Project Structure Understanding

We have seen that architects often include infographics to explain information about project layouts and design choices that cannot be visualized using only the maquette. However, it can be questioned to what degree stakeholders can understand the project structure using only these two formats. For

stakeholders that are part of the architectural domain, this might not pose to be much of an issue, since they are accustomed to the way information is portrayed within this sector. But the same does not count when communication is required with stakeholders that are part of a multi-disciplinary team.

To ensure the project structures and design choices are interpreted by everyone in the multi-disciplinary team, sufficient information about the project should be made available. Architects are currently still limited in the amount of information and detail they can show on one infographic and consequently, often must prioritize which information they do show. In AR we do not have this limitation, since artifacts can be loaded dynamically in the environment, meaning that if users want more information about a specific part, they can request it using the system. Users can also look at multiple objects simultaneously by being able to drag them closer together, meaning the links between these objects and their implications can be portrayed more clearly to the user. Similar to the use case in section 2.2.1, the physical maquettes are used for project versions within the system. However in this case, we will also include all of the artifacts required to understanding the full project structure as digital supporting models.

A project that these practices can be applied for, is a project exploring the conceptual designs for a new detention center in Genk (see Appendix C). This project was collected from a student Architecture that had to make it for a course called “Conceptual Thinking”, which explores the aspects of using colors to highlight the different functionalities of buildings. This project is perfect for the representation of this use case, since it includes many artifacts that are related to each other including rendered versions of the model, an infographics poster with a front and back side and an overview of Genk with the rendered model visualized inside.

However, inside this project there was no physical maquette constructed to communicate the design to stakeholders. Because we want to explore linking different artifacts to a real model, we have chosen to recreate two of the model versions ourselves using Lego³. These reconstructions can be seen on Figures 16 and 17. We used Lego to construct these models, since it provides us the ability to quickly build the models, while maintaining the colors used in the conceptual design as close as possible. Within our system, these two models should only be portrayed as physical models instead of an aligned digital object, since we want to directly link the digital artifacts to real models in the environment. To exemplify how we can provide these links, we will now highlight the different versions and relations existing between the artifact.

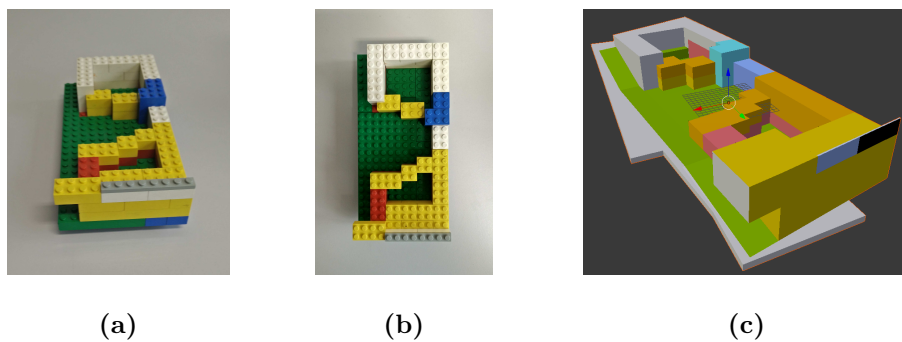


Figure 16: Physical Lego maquette for the first version with (a) the front view, (b) the top view and (c) the original model version in Blender.

³Website of Lego: <https://www.lego.com/en-us>

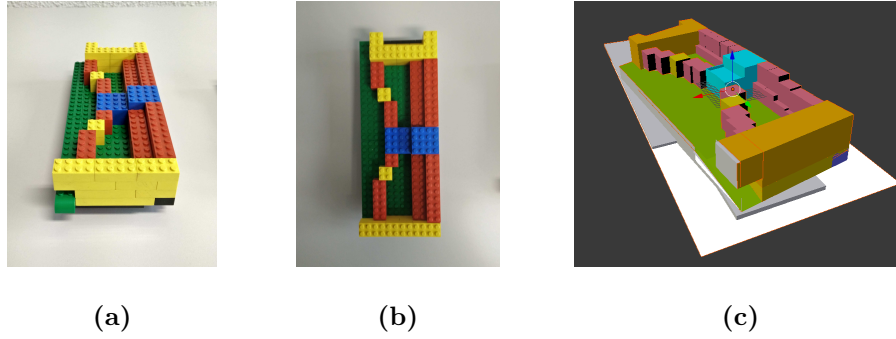
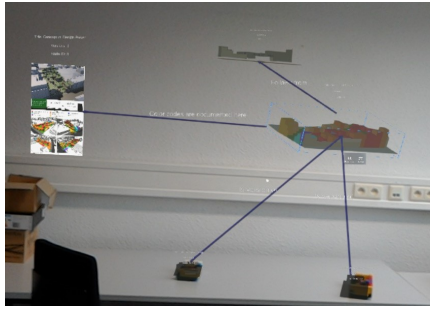


Figure 17: Physical Lego maquette for the second version with (a) the front view, (b) the top view and (c) the original model in Blender.

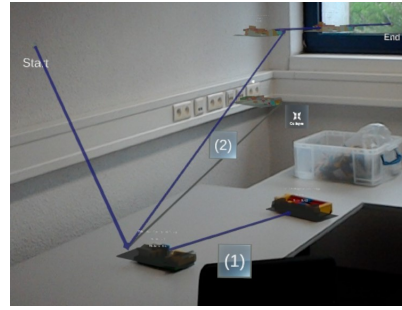
Conceptual Design (Figure 18): This model is the conceptual design of the project, that uses the different colors to highlight the building functionalities. As seen on Figure 18a, it is linked with several other artifacts including: two physical maquettes that are earlier versions of the design, the infographics poster since the color codes of the model are documented there and lastly, the rendered model since the render was created using the conceptual design as a reference. The conceptual design consists in total of six versions (seen on Figure 18b), where the first two are represented as real models and the others remain digital. In general, the structure consists of the first real model being the entry point of the versions and from it, three different paths are followed that represent alternative designs that were explored. The first path goes to the other real model (the second version), and does not continue from there since this version was not selected in the end. This path also contains a decision card to highlight an increase in confined residences (color red), which caused a decrease in recreational activity building blocks (color yellow or orange). The second real model itself also consists of two versions (seen on Figure 18c), that follow each other sequentially and where only minor changes were made. The second path goes to the third version which is also the version that was continued with. This path contains two decision cards which highlight that there was a change in flow between the public and private spaces and that the third version was selected because it provided the most amount of residences while retaining the separation between the private and public spaces. Finally, the third path goes to another alternative design. However, this path was not continued with and does not contain any decision cards.

Rendered version of model (Figure 19): This model is a rendered version of the conceptual design and is linked to the overview of Genk as being a part of it, since the rendered model can be seen in the overview. This link contains one decision card which explains the model was positioned at the specific location since it will be used as both a public space with a road for pedestrians and as a private space for the different residents. The rendered version is also linked to the conceptual design (as seen previously) and is linked to the poster since the rendered version is visualized inside the poster. The rendered version contains two versions that were made in parallel with each other (see Figure 19b), where only one of them was selected in the end.

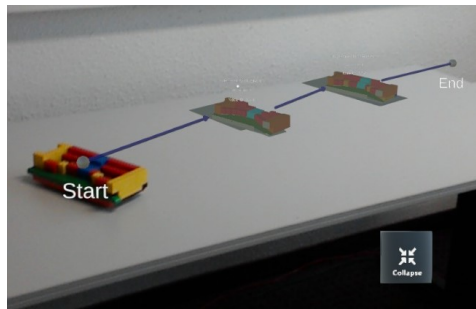
Infographics poster (Figure 20): For the infographics poster, there are two separate models, one for the front side (left of Figure 20a) and one for the back side (right of Figure 20a). Both of these models are linked together since they are both part of one poster, where the label of the link mentions the two models to be interdependent of each other. For the front



(a)

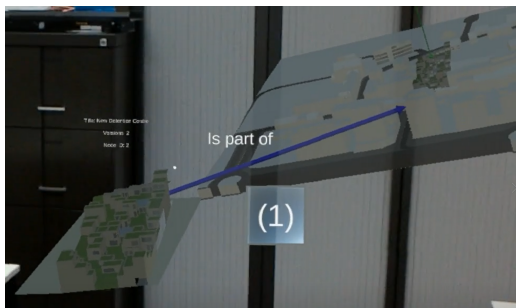


(b)

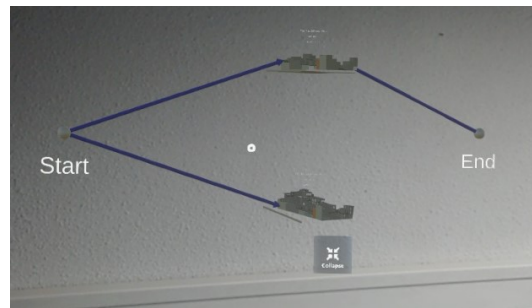


(c)

Figure 18: The conceptual design with (a) the models linked to it, (b) the versions contained within and (c) the versions for the second real model.



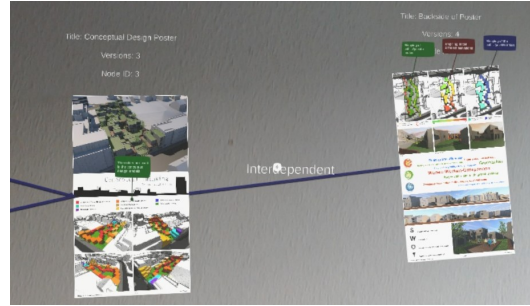
(a)



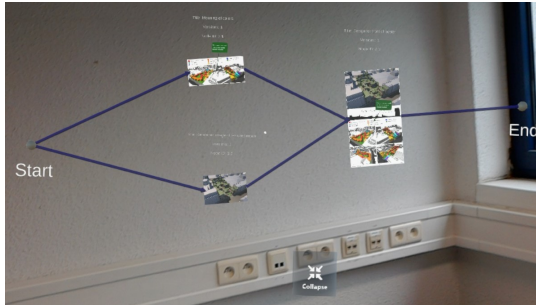
(b)

Figure 19: The rendered version of the model with (a) the model linked to the overview of Genk and (b) the versions contained within.

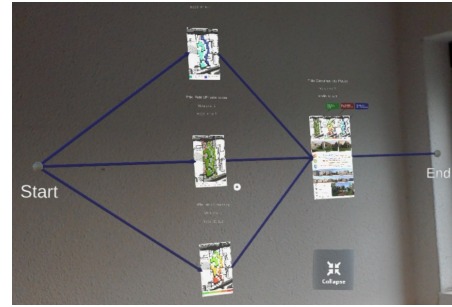
side, an annotation is provided to highlight that the mentioned colors are used to build the conceptual design. As seen in Figure 20b, the front side consists of two parts that have been merged together into one version. These two parts are the color codes and the high quality rendered image. The color codes have the same annotation as the final version, namely the use of the color codes within the conceptual design. For the back side of the poster (Figure 20c), three annotations are provided to explain the different components of the poster. These components are the heightmap, public/private space map and the routing map. They are also visualized as the different versions seen on Figure 20c and merge together into the final version of the poster back side.



(a)



(b)



(c)

Figure 20: The infographic posters with (a) the overview of both posters, (b) the versions of the front side and (c) the versions of the back side.

Overview of Genk (Figure 21): This model is an overview of the center of Genk and has the rendered version of the conceptual design contained within itself. There is also one annotation that mentions the detention center being visualized more clearly compared to the other models. Like previously seen, a link exists between this model and the rendered version, but this is the only link for this model.

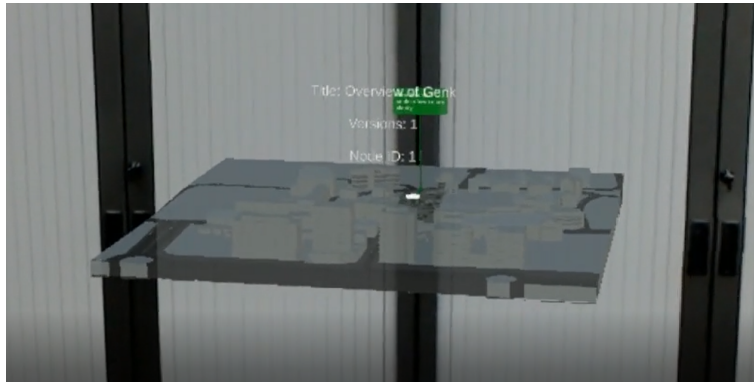


Figure 21: An overview of Genk visualized in AR

3 Related Work

Both artifact traceability and Augmented Reality have their own techniques, research and implications which we attempted to combine within our research. Because both these parts are inherently separable, we first discuss for both parts what techniques and research are of relevance to our system. Afterwards, we mention other research that has combined these two factors so we can evaluate the contribution of our own research compared to others within the same context.

3.1 Artifact Traceability

Research has already been conducted to apply artifact traceability for fields like software development [38, 5, 21], industrial product design [29, 35], architecture [25, 12] and business process contexts [40, 17, 39]. An example research was conducted by Kriglstein et al. where they focused on creating a visualization approach for providing a comparative analysis of categorical datasets in business process models. They discovered that visualizations of process versions can greatly support the decision making of the end users by presenting the different relations, patterns and anomalies between the process versions [28]. However, this does not necessarily have to be the case for process versions alone. A different study carried out by Demian et al. looked at the use of a version visualization system for storytelling designs. Similar to the research of Kriglstein et al., they found that the visual explorations of design evolution helps in understanding and reusing previous designs and also supports subsequent decision making [14].

There are several different needs for artifact traceability, where the key purposes change depending on the considered field. Berenbach et al. mentioned the three most common needs for traceability includes: impact analysis, coverage analysis and derivation analysis [7]. We discuss them here further, so we can compare later how our research might satisfy these needs.

3.1.1 Impact Analysis

Lee et al. defined change impact analysis as “identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change”. In their research, they proposed a goal-driven requirements traceability approach to develop and manage requirement changes along three dimensions. This also includes the analysis of requirement change impacts. In impact analysis, when a user proposes changes during software evolution, change impacts are analyzed to find both the affected requirements and the affected use cases [30].

Because of the high costs some changes can have, visualizing the impact of changes correctly plays a crucial role in ensuring project teams make the right decisions. Since changes can affect the entire ecosystem of a project or process, what are important costs can vary depending on the context it is viewed from. Rovegard et al. found that the assessment of the importance of impact analysis issues depends on the organizational level. They analyzed three different organizational levels: **operative** (concerned with realizing the project according to plan), **tactical** (concerned with planning of time and resources) and **strategic** (which addresses long-term goals and product aspects). As one would expect, these three different organizational levels assess the importance of impact analysis issues depending on their own concerns. On the operative level important issues were related to the realization of a change or a chosen solution. On the tactical level they were related to project and planning aspects and finally, on the strategic level, they were related to process and product quality [32].

Due to the dynamic nature of changes and because the parts of impact costs that are considered to be important can vary, effectively visualizing the impacts of these changes becomes a difficult task. This claim is backed by the fact that, at the current time, finding research that attempted this kind of visualization is hard to find. Research that did try however, focused specifically on the change impact detection of meta-models. Within this research, they defined a methodology

on how to prevent information erosion during the change impact detection. However, the problem of adapting the diversity of artifacts when the meta-model undergoes modifications still remains intrinsically difficult [15].

Since our research will focus on analyzing how AR can be used to document the evolution and changes of real artifacts, adding change impact analysis to the system would add a large amount of complexity outside of the main research question. While we do not include this type of analysis as one of the core goals, providing traceability for other purposes could include positive results for the analysis of change impact as well.

3.1.2 Coverage Analysis

Coverage analysis determines whether all requirements and specifications in a contract have been satisfied by the end results [7]. The practice of coverage analysis is mostly prevalent in “Software Engineering”, where requirements are defined as software features and the resulting software is tested on whether it fulfills these requirements. One of the most commonly known coverage analysis tools is the “requirements traceability matrix”⁴ from the field project management. The requirements traceability matrix is used within every process, from project initialization to final implementation, to track all the requirements and whether they are being met by the current process and design. As an example of this, the different requirements in Figure 22 each include their own testing strategy and contain a status on whether they are already finished. There are different variations of the requirements traceability matrix where some fields have either been removed or added depending on the practices of the company that uses it.

Unique Req ID	Requirement description	Source /Requestor	Org /Dept	Business Justification/Need	WBS Deliverable	Test Strategy	UAT Responsibility	Status	Active/ Inactive Flag	Comments
1	Change the table component on the dashboard to a graph.	Ella Allen	Sales	Better representation of the data and improved readability	Task 1.1 Task 4.7	Use cases to be developed.	Follow the test steps as defined in use cases and report any defects.	Done	Active	Jan 5:- Testing started. Jan 8:- Defected reported. Jan 9: Defect fixed Jan 10: UAT Continued
2	Add a drop down list for the regions	Tonya Harper	Sales	Will enable Area managers to understand their market more accurately	Task 1.2	Load testing to be done.	Load runner to be used to simulate a load and the regions will be verified for accurate representation	In Progress	Active	Make sure US territories hawaii and puerto rico are included in a separate regional unit.
3	Create a new category hierarchy to sorting the result set	Sammy Butler	Pricing	Will help the pricing department by automating the selection of categorized data.	Task 1.3	Assigned business users to perform unit testing as well as UAT	Check the categories in the base tables in the EDW.	Hold	Cancelled	It was determined that Pricing requirements were out of scope for this phase
4	...	...	...	...	...	...	...	...	...	...
5	...	...	...	...	...	...	...	...	...	...
6	...	...	...	...	...	...	...	...	...	...
7	...	...	...	...	...	...	...	...	...	...
8	...	...	...	...	...	...	...	...	...	...

Figure 22: Example Requirements Traceability Matrix (Source: <https://project-management.com/requirements-traceability-matrix-rtm/>)

While we cover some case of software traceability within our research, the main functionality consists of being able to gather information about a real object for multiple sectors. Considering not every field has a practice of defining requirements like in Software Engineering, focusing on coverage analysis for our system would imply that these other fields would have to start using the same practices to be able to make use of our system. Instead of enforcing this, we focus on providing traceability information of real objects by using information that is already known about these objects.

3.1.3 Derivation Analysis

Derivation analysis is the practice of discovering the origin and rationale of a function. The main flow of derivation analysis is to trace information from a feature, back to the stakeholder requests

⁴“Requirements Traceability Matrix” definition on <https://project-management.com/requirements-traceability-matrix-rtm/>

or business goals that led to the decision to put the feature in the product [7]. This type of analysis will also be the main focus for our system, since the use cases we provide consist of tracing the design rationale of software GUI's and tracing the design changes in architectural building models (see section 2).

Most often, research providing means for derivation analysis also focus on capturing the design rationale of software. For example, Gutierrez Lopez et al. who defined the decision cards (see section 1.1), found that decision cards allowed designers to elaborate their decisions freely and could be used in combination with design artifacts to support awareness and traceability on the design process [20]. The follow-up study called *Helaba* investigated the required characteristics necessary to facilitate traceability of creative design processes in a digital system. Decision cards were used within *Helaba* to log the decisions between the different artifact versions. Furthermore, users could add location specific commentary on the artifact by placing a pointer on the object and selecting the area of relevance. The main advantage of using decision cards for documenting the traceability of the artifacts, is in its lightweight format that allows users to freely choose how they document the decision, while still providing enough information for others to understand the mentioned decisions [21].

Another example of capturing design rationale for derivation analysis was a research by Bracewell et al. where they created a “Design Rationale editor” that allows Software Engineering designers to record their rationale as the design proceeds. Instead of focusing on the format of the individual decisions, this editor focuses on how the different decisions are linked together. They do this by creating a graph of different annotation types that are linked together to form the thought process of the decisions within the system. As shown in Figure 23, the supported annotation types are an idea (light bulb), a description (plus or minus sign with border) and a question (question mark). Depending on the color of the icon (green or red), the statement made by the annotation was either accepted or rejected. Within the editor, they also allow objects to be included in the graph so the different annotations can be linked to objects as well [11]. One of the disadvantages of the editor compared to the decision cards, is that it becomes more time-consuming to document everything during the design process. A lot more information needs to be documented and links need to be drawn correctly to avoid wrong derivations by the end users.

Like seen in section 1, we focus within our research on the use of decision cards for tracing the design rationale. We do not directly apply the practices used within the *Design Rationale editor*, since we want to focus on providing information about the design rationale present in the changes of different artifact versions. Decision cards provide more support for this use case compared to the *Design Rationale editor*, since the decision cards format also focuses on logging the decisions between artifact versions, while the *Design Rationale editor* provides more information on the final decisions of relevance. However, when applied correctly, using the lightweight format of the decision cards should allow us to portray the same information provided by the *Design Rationale editor* (e.g. specific decisions that were omitted and accepted).

To provide derivation analysis for projects, often a format called “branching” is used to visualize the flow of changes within the projects. Branching is defined in the Computer Weekly News as “*The history data of a prior navigation session is presented to the user as a graphical view structured as a branching visualization of the nodes and node branches. The user can interact with the graphical view to move forward and backward in the time span of the session and retrieve the document as any given point in the previous session. Moreover, the branching visualization serves as a map that traces the user navigation over documents and content during the session.*” [1].

One of the most commonly known examples of branching, is the visualization of a “Git workflow” in software development (shown on Figure 24). Every node on the branch represents a “commit” of code that developers can choose to go back to and explore. This practice is commonly also referred to as the “semantic zoom” within a system, which we define as “the ability to let a user zoom in

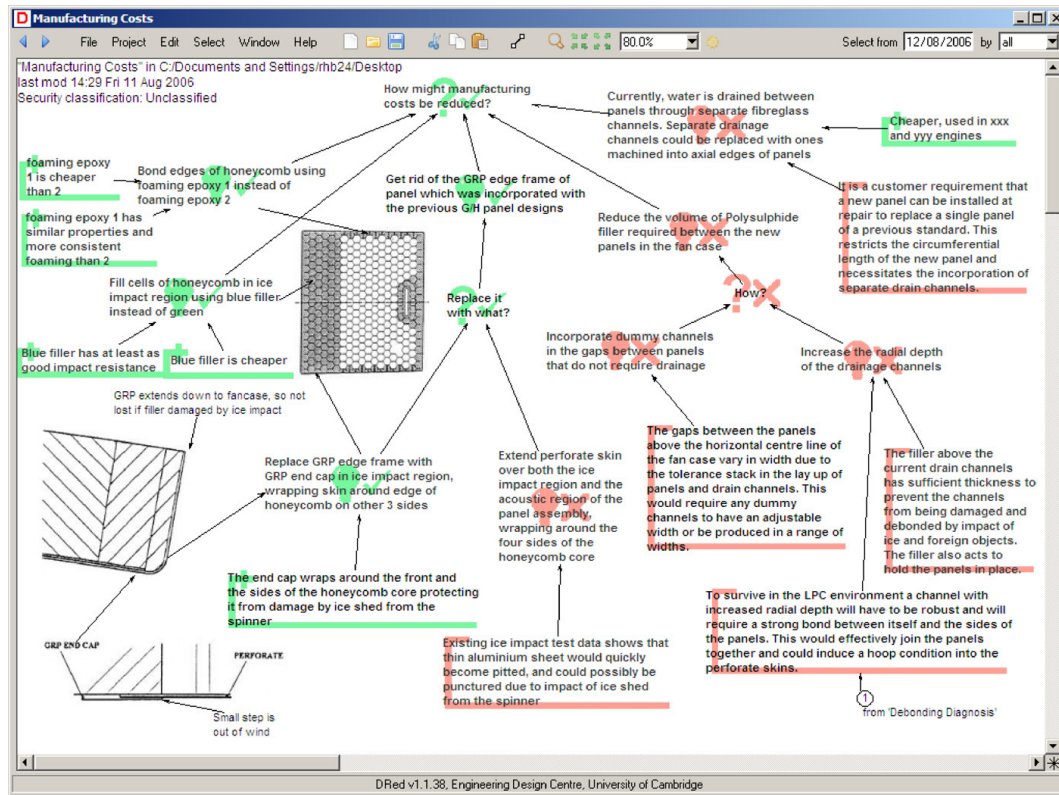


Figure 23: Design Rationale editor [11]

and out of a particular view in a collection of multiple views”⁵. Nodes are shown to be in parallel branches when they were created simultaneously or separated from each other but are shown on the same branch when they sequentially follow each other. Branches can even merge to combine the code of both branches together into one version that incorporates the changes included within both branches.

Since we want to provide artifact traceability by linking different versions and artifacts together (see section 1.1), we also make use of this visualization technique. In section 4, we further emphasize on how we have applied this practice within our system.

3.2 AR Use Cases

Research on AR has been carried out as early as 1960, where a see-through device was used to present 3D graphics. There are several different ways Augmented Reality can be visualized, with examples being objects portrayed in space on a smartphone camera view [10, 18, 31], real-time holograms using optical reflections in physical space [27] and visualizations on a head mounted display that overlays the view on the real world [13]. Currently research on AR is still on the rise with several tools already being used commercially in multiple sectors[4]. We discuss three aspects of AR that have been explored over the past few years and are relevant within our research as well. These aspects are: object placement, annotations and artifact traceability.

⁵Explanation of “semantic zoom” based on the following link from Techopedia: <https://www.techopedia.com/definition/27605/semantic-zoom-windows8>

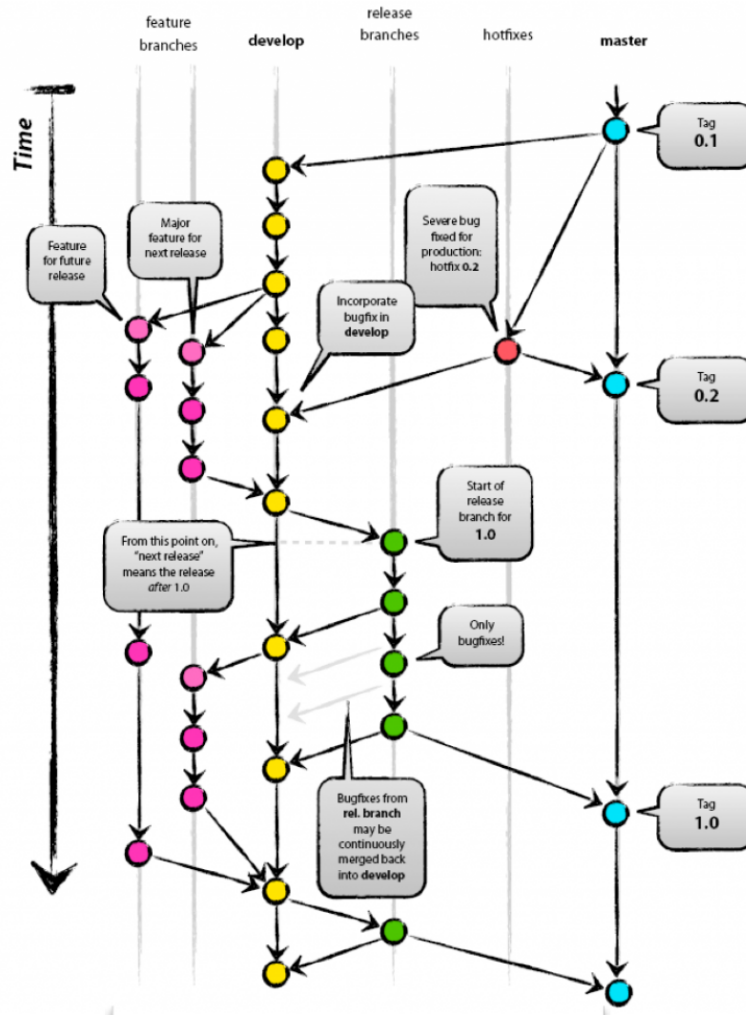


Figure 24: Git Flow Diagram of a project workflow (Source: <http://socialsharing.info/git-flow-diagram/gitflow-the-easy-release-management-workflow-git-flow-diagram/>)

3.2.1 Object Placement

The current most common use of AR is the visualization of objects that would be hard to physically place in front of the user otherwise. Examples of this are most commonly found within marketing, where users can look at products a company has to offer, using AR to get a better view of the product from the comfort of their own home [33]. One example of this, is an application made by IKEA in 2018, where users can virtually place furniture inside their house by using AR on a mobile device to explore the different designs of the furniture (see Figure 25)[10]. However, IKEA is not the only company that has built such a system. In 2018 we saw a rise of similar applications [31, 34], because of the new mobile AR technologies titled ARKit and ARCore, made by Apple and Google in 2017. These toolkits allow developers to analyze the environment with depth sensing sensors inside the phone to place objects more accurately than what they could before. Furthermore, it provides the possibility to do accurate measurements of the environment, ensuring that the scale of the virtual objects stay consistent with the real-life measurements [18, 8].

A different example on how marketing makes use of object placement in AR, is a system to

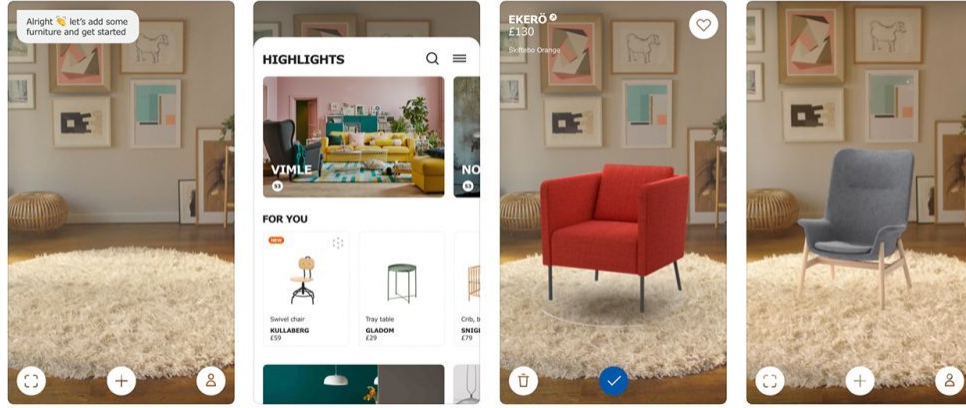


Figure 25: Screenshots from the IKEA Place app (Source: <https://www.chatelaine.com/wp-content/uploads/2018/05/Ikea-Place-App-1000x433.jpg>)

visualize products shown inside of a physical catalog (as shown in Figure 26). When users are reading a catalog and find an item of interest, they can visualize it within their phone or tablet on top of the catalog. Like in the previous example, users can grasp the real-life spatial properties of the item, without having it physically present before them. The system knows what model to show and where to place the model by use of markers or images that are scanned within the application. Because of its simplicity (no need for accurate depth sensing), this technique has already existed for a longer time with one of the earliest examples being from Blanco-Novoa et al. in the year 2000 [26]. Despite the age of this method, it is still one of the most commonly used for making virtual objects appear at certain specific locations in an AR environment.

In some cases, it also becomes necessary to visualize objects that are on a larger scale than those that can be visualized in front of the user in a room. Adascalitei et al. mentioned the use of mobile AR for visualizing buildings on the construction site before they have been built. The research mentioned a virtual tour of a hotel complex in 2011 in Finland, where mobile devices were used to show the properties of the hotel complex before the construction had started (see Figure 27). The participants of the virtual tour noted that using AR helped them to estimate and understand the volumes and construction plans of the buildings [2].

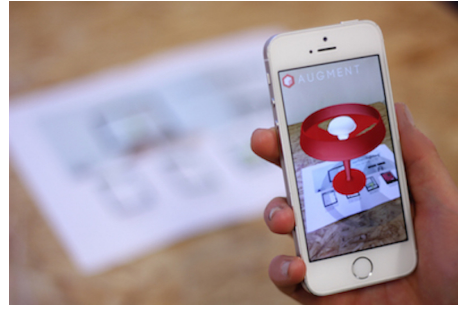
All the aforementioned examples emphasize the fact that, no matter the scale, mapping virtual objects on a real environment using Augmented Reality proves to be useful for understanding the spatial properties and components of specific virtual objects. Since we need to provide artifact traceability for both real and digital objects within our system, we will also be required to use the practice of object placement within our system. For example, we can place the digital objects within the AR environment to help users understand the spatial properties of the models within the system. We can also use the markers to align the digital object with a real object in the environment to acquire more detailed information, like seen in section 2.

3.2.2 Annotations

The previously mentioned systems only visualize the shape, location and size of the objects, but do not include other details like design choices, material information or component descriptions. While these details could be mentioned in the object list of the IKEA application on Figure 25 or in the pages of the pamphlets on Figure 26, there is a limit to the amount of information that can be presented in 2D due to lack of space and perception. In AR this limitation is not as present, since



(a)



(b)

Figure 26: (a) Letters and brochures from Strata showing a table in AR (Source: <https://ar.strata.com/home-2/industries/>), (b) Brochure from Northern Lighting using Augment to display models (Source: <https://www.augment.com/blog/5-companies-using-augmented-reality-print-campaigns-brochures/>)



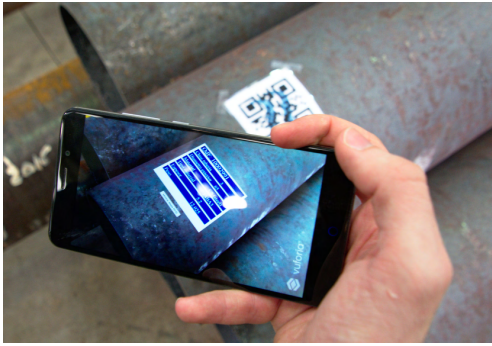
Figure 27: Mobile Augmented Reality visualization of hotel complex in Finland 2011 [2]

the full physical environment around the user can be utilized for visualizing information about the objects.

One way to visualize information about these objects in AR, is by showing annotations to the user like we have seen in section 1. Blanco Novoa et al. researched a system for an industry shipyard where they use AR for: localizing the pipes within the building, visualize information about these pipes and visualize information about the contents of shelves within the shipyard (see Figure 28). They achieve this by using tags attached to different objects all over the shipyard, that can be scanned with the camera of the mobile (see Figure 28a). Using this method, the AR system knows what information to show and at what specific location to show it [9]. This is an example on how AR can be used to acquire information “on the go”. By opening the application and looking at certain objects, users can quickly acquire the information they need. To receive the same type of information using traditional methods, users must either search for the right physical papers lying around somewhere in the environment or need to navigate through a digital system in search of the desired information. Kumar Gupta et al. mentioned another system that allows this type of information acquirement, where a customer feedback system within airplanes allows the airplane staff to see the customer satisfaction and possible problems for every single airplane seat [19].



(a)



(b)



(c)

Figure 28: Use of mobile Augmented Reality in a shipyard with (a) the localization of pipes in a workshop, (b) additional information about a specific pipe and (c) the contents of the shelf displayed on a tablet [9].

Aside from being able to acquire information about specific objects in the environment, AR annotations can also be used to guide users within a city, large complex structure or within a specific facility. An example of this, is a system used in Copenhagen airport (see Figure 29), where the AR application helps users navigate through the airport by using the airport's Wi-Fi infrastructure for tracking the user. In normal situations, users would have to figure out the directions by reading the signs scattered across the airport. However, in a large, busy and complex airport this can sometimes prove to be difficult. Within the AR application, annotations show the users what steps they need to take and where they need to go, making it easier to navigate through the airport [2].

Another example on how annotations can be used to guide users, is a remote guidance AR tool built by Adcock et al.. Here a remote user can guide a local user by drawing annotations that appear in front of the local user using a projector. The remote user can see the perspective of the local user through a camera on top of the helmet that the local user is wearing [3]. This example indicates how AR and annotations can provide new ways for users to collaborate remotely for carrying out complex tasks in a real environment, something which would be a lot harder without the use of AR.

Like seen in section 1, we also make use of annotations within our research. In our case, these annotations are provided to deliver more information about the specific components of models. The decision cards and link labels we provide within the system can also be considered some form of



Figure 29: Copenhagen airport Mobile Augmented Reality navigation system [2]

annotations, since they are used to visualize information about models in the AR environment. Compared to the practices we have seen within this section, our annotations will most likely look like the ones seen in Figure 28a, but then for locations within components instead of locations within the environment.

3.3 Artifact traceability in AR

We previously discussed examples of artifact traceability and Augmented Reality separately within sections 3.1 and 3.2. However, within our research we focus on combining these two techniques to provide artifact traceability in AR. We now discuss other examples that have also provided some form of artifact traceability using AR, so we can mention what we contribute for this field with our research.

A first example of artifact traceability in AR, is research conducted by Karsch et al. which involves visualizing the progress of building constructions. However, in their research they used a different approach than the ones seen previously in Figure 27. In their system, called *ConstructAide*, they take as input several pictures of buildings, 3D models of the buildings and an accurate mapping of the models to each picture. The result is a 4D tool (as shown in Figure 30), where the user can explore for every part of an image what the end result will look like, what it used to look like in the beginning and how the construction is progressing. This tool complies more with the concept of artifact traceability than the previous examples, considering the progression of the buildings is properly presented and can be explored. *ConstructAide* also uses a different kind of AR compared to previous examples, since the visualizations are not happening in real time through a camera, but instead are presented onto pictures that have already been taken [25].

A different example of artifact traceability in AR is a system by SRI International⁶ (Figure 31), which visualizes what the building will look like in the future and what areas of the construction are not following the construction plans and will need to be revised [23]. Furthermore, it allows users to take measurements at problematic locations to get more information about any errors present within those locations compared to the construction plans. This tool enhances the concepts of the previous example, by giving real time information on the current construction progress and providing visualizations for any issues that had occurred.

While these examples have covered the progression of building constructions as part of the traceability process, within our research we focus on providing traceability for the building designs instead. During the process of building design, multiple versions are created to explore the different possibilities that can be taken for the building. This design process usually occurs before

⁶Website of the research institute of SRI International: <https://www.sri.com/>

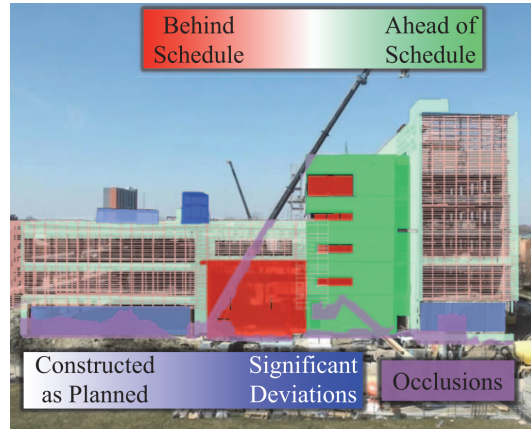


Figure 30: Visualization of building progress in *ConstructAide*[25]



Figure 31: Screenshots of “Augmented Reality Solutions for Construction Inspection” [23]

the construction of the building, since it also decides how the construction will need to proceed. However, while we do not specifically cover the construction process of the buildings like the previous examples, the traceability information for the designs covered within our system can still provide information about what parts to construct and why they need to be constructed that way. This means benefits might still be found for our system within the same use cases as the covered examples.

4 Implementation

In this section we discuss the following parts: the general overview of the system, the AR hardware used, the visualization of artifact versions, the types of artifacts we support, what annotations we provide and how we detect and align digital objects with real objects. We start with the general overview of the system, since it provides the information required for understanding how the different parts of the system have been implemented.

4.1 General overview

As shown on Figure 32, our system consists of two main parts: a NodeJS server ⁷ that runs on a local computer and listens for HTTP requests and an AR headset that uses Unity to visualize objects in the environment and sends HTTP requests to the server. We will discuss both elements here separately to provide a general overview on how the system operates.

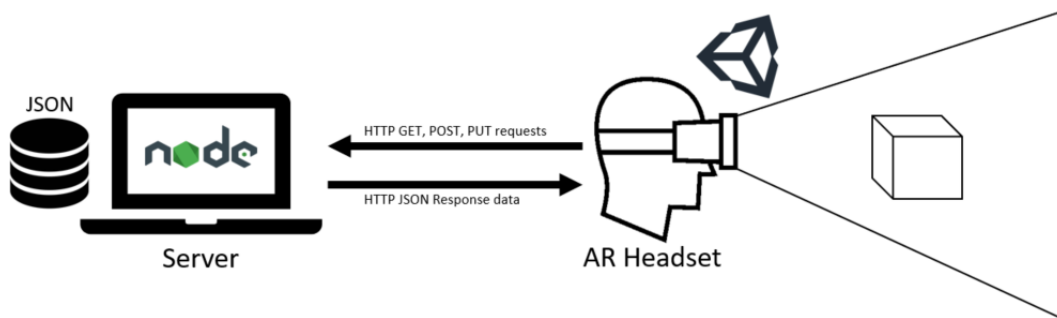


Figure 32: General overview of the implemented system

4.1.1 NodeJS Server

The server seen on Figure 32 is used to manage the different project configurations, which includes the artifact versions, artifact links and annotations. All this data is stored in JSON (Javascript Object Notation) format⁸ on the server, including elements like artifact positions, rotations and sizes within the AR environment. This allows us to manage where objects will appear in AR, so we can quickly build different project scenarios that need to be visualized within the AR environment.

The server provides a REST (Representational State Transfer) API ⁹, which defines the different HTTP endpoints and methods that the server will listen for and how it will respond to the requests. To provide this functionality, we use a NodeJS library called “Express”¹⁰, which is used to build a minimalist web framework very fast. We used NodeJS for the server since there are a lot of libraries, like Express, available on NPM (the Node Package Manager)¹¹ that allow us to build a lot of functionality using only a minimum amount of time and effort. Since NodeJS is essentially Javascript, manipulating and saving JSON data is supported natively, which makes providing the project management on the server efficient and simple.

⁷Website of NodeJS: <https://nodejs.org/en/>

⁸Official JSON format documentation: <https://json.org/>

⁹REST API website: <https://restfulapi.net/>

¹⁰Website of Express library: <https://expressjs.com/>

¹¹Website of NPM: <https://www.npmjs.com/>

Since the project management is separated from the AR system, it becomes possible for multiple AR systems to interact with the project simultaneously and change the project configurations at run-time.

4.1.2 AR System

For the AR system we make use of Unity¹² to build the different artifacts and provide ways to interact with these artifacts. Unity is one of the most widely used and supported 3D development platforms, including for AR use cases. The most common language used to develop with Unity is C#¹³, which is also the language that we will be using to implement our system. To construct the components for the AR system, we make use of the Unity Editor shown on Figure 33, which gives developers the ability to build their own 3D components to use within their system¹⁴.

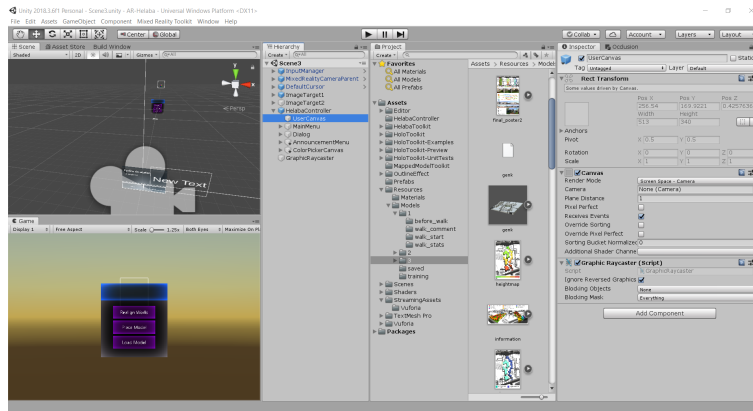


Figure 33: Unity Editor showing a scene from our system

By using the Unity development platform for AR, objects can be spawned dynamically in the environment that users can manipulate using different gestures. When our Unity program is booted on the AR headset, HTTP requests will be sent to the server endpoints to request the current state of the stored project. Based on the JSON data received from the server, the Unity program will save an internal state of the project configuration as Unity objects. This way it will know what objects to visualize, what properties and functionalities they have and where to place them in the environment. When the user interacts with objects in our system, HTTP requests will also be sent to the server to update the state of the manipulated object within the project configuration. Example interactions are moving, scaling and rotating the objects within the environment. The AR system will currently also poll the server for changes every five seconds. However, if multiple users need to be supported in the system, this time would have to decrease to polling every second to ensure users are up to date with the interactions of each other.

4.2 AR Hardware

Like mentioned in section 1.2, we use a head mounted display since it allows us to overlay information about any object in the user his environment. There are already a variety of AR headsets available

¹²Website of Unity: <https://unity.com/>

¹³Blog of Unity explaining supported languages: <https://blogs.unity3d.com/2014/09/03/documentation-unity-scripting-languages-and-you/>

¹⁴Unity Editor website: <https://unity3d.com/unity/editor>

for developers to use, with examples being Microsoft Hololens¹⁵ (Figure 34a), Meta 2¹⁶ (Figure 34b) and Magic Leap One¹⁷ (Figure 34c).

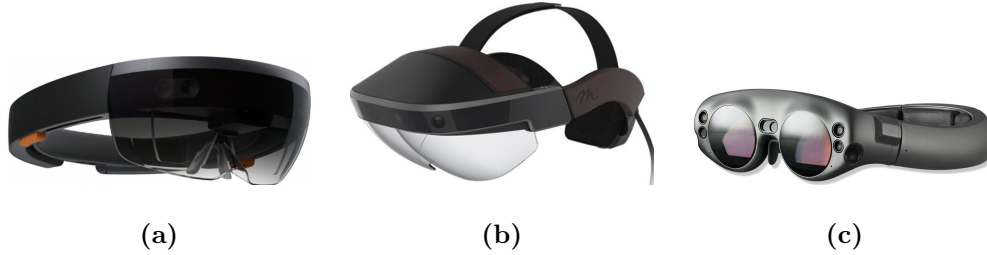


Figure 34: Different AR headsets with **(a)** the First edition of Microsoft Hololens, **(b)** the Meta 2 and **(c)** the Magic Leap One (Sources: <https://www.extremetech.com/wp-content/uploads/2015/10/HoloLens.jpg>, <https://dab1nmslvvntp.cloudfront.net/wp-content/uploads/2016/03/1457324978meta2.jpg>, <https://www.spiria.com/sites/default/files/blog-2017/magic-leap-one-1.jpg>).

There are several requirements that the headset needs to fulfill for us to be able to use it for our research. First off, since we are providing traceability for specific artifacts, we need the ability to integrate our own models into the system so digital copies of the real artifacts can be visualized and manipulated (for example by using Unity). Secondly, in section 2 we mentioned the need to detect the artifact that the user is looking at and the need to align digital objects with these real objects. For these features we will need a front camera that we can access for providing the required detection mechanisms. Lastly, we need gesture support for the user to be able to interact with the system using high degrees of freedom, to simulate the tangible feeling of handling paper mockups mentioned in 2.1.

Within the institute, we already possess a first edition Microsoft Hololens, which has a front camera that can be accessed, Unity support for development (meaning custom objects can be integrated) and gestures are supported natively. In other words, this headset fulfills all the requirements which we stated, meaning we can use it for the implementation of our system. There are some known minor issues about the use of Microsoft Hololens. For example, there is only a small field of view that is not able to visualize all the digital objects in the environment simultaneously. Also, for Hololens it is recommended to set the clipping plane far enough to avoid nausea for the users, but this setting makes parts of objects disappear when users come too close [6]. However, these minor issues do not pose a problem for our research since we can allow users to increase the size of objects to make them more clearly visible from a distance and we can even turn the clipping plane issue to our advantage (more on how we do this can be found in section 4.5).

When using Unity to develop for the Microsoft Hololens, we specifically employ the use of the Mixed Reality Toolkit¹⁸ developed by Microsoft. This toolkit provides basic building blocks for Unity development on Hololens, Windows Mixed Reality and OpenVR. Example functionalities that we imported from this toolkit are manipulation gestures for the artifacts and bounding boxes to highlight the artifacts with. This does mean that when we want to support other AR hardware that does not include the aforementioned technologies, different Unity toolkits or libraries might need to be used. However, for our current use cases it is enough for us to only focus on the development of the system on the Microsoft Hololens.

¹⁵Documentation of Microsoft Hololens first edition: <https://docs.microsoft.com/en-us/hololens/>

¹⁶Website of Meta AR Headset: <https://metavision.com/>

¹⁷Website of Magic Leap One: <https://www.magicleap.com/magic-leap-one>

¹⁸Documentation of the Mixed Reality Toolkit: <https://microsoft.github.io/MixedRealityToolkit-Unity/README.html>

4.3 Artifact Links and Versions Visualization

To provide artifact traceability for the use cases mentioned in section 2, we need to be able to visualize the different versions of an artifact using the AR system. For this, artifacts need to be linked like the concepts defined in section 1. This is to ensure the visualization provides the required traceability information so users can discern: how artifacts are related to each other, in what order artifacts were created, what versions an artifact contains, what has changes between the versions and which versions were selected or discarded. We will first cover how the artifacts are linked together, since this is also relevant to explaining how the versions of an artifact are visualized.

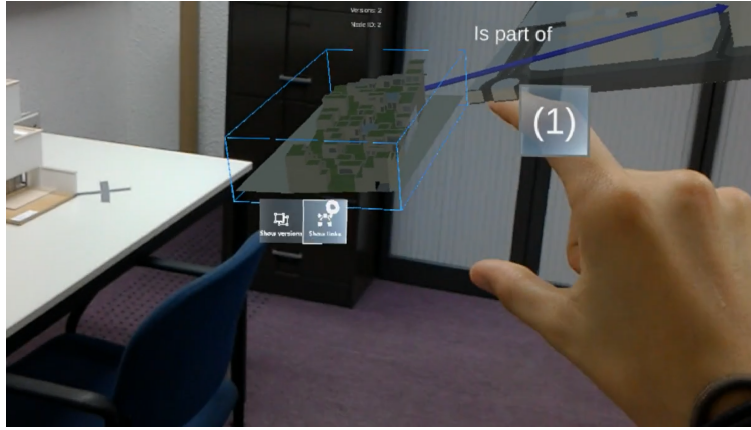
First off, we provided the ability to selective choose for which objects links need to be visualized. Users can first tap on an artifact to select it, in which a bounding box and several options will appear for that artifact. This is visualized on Figure 35a. In this image, the first button is used to visualize the versions of the object and the second to see the links between the current artifact and the others. To avoid cluttering the room with arrows, visualizing the links for one artifact will also hide the links that were shown for a previously selected artifact. Users can also select to hide the links of an artifact in case they obscure the vision of the artifact. This is shown with the “Hide links” button on the selected artifact in Figure 35b. Some links can even include decision cards between the objects, which we explain further in section 4.4. Applying this method of providing links and filtering them, allows users to only focus on the aspects of the artifact they are currently exploring, while still being able to see which other models are of relevance to this artifact.

Figure 35b shows one of the links visualized for the artifact that the user has selected. The link is visualized as an arrow to show which of the two objects came first and which came after. Furthermore, a label “Follows from” is provided to clarify the meaning of the link between the two artifacts. In this case, the relation mentions the second object to be the continuation of the first object. Note that in Figure 35a, a link with another object is also visible that defines a different relation type (“Is part of”). This visualization is consistent with the guidelines and visualizations we mentioned in section 1.1, since it provides a visual connection between the different artifacts.

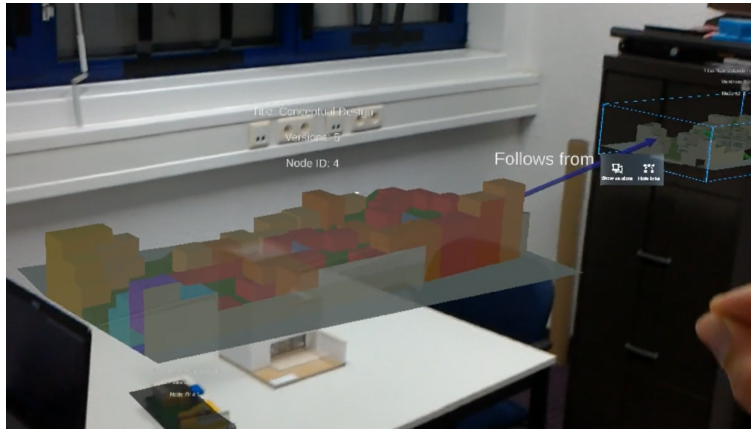
In our system we support both 2D and 3D artifacts that users can move, scale and rotate using pinch gestures in mid-air. When users point the cursor of the Hololens towards the object and then pinch and drag with one hand, the object will move around together with the dragging motion of the hand. Meanwhile, if the user uses the same motion but with two hands, the object will scale and rotate depending on the movements of the two hands. For example, moving the two hands away from each other will increase the size of the object, while putting them closer together will decrease the size of the object. When moving the two hands in parallel to each other but in opposite directions, the object will rotate together with the directions of the movements.

When the objects are manipulated using these gestures, the aforementioned links will move as well and the endpoints of the links will automatically adjust themselves to be on the side closest to the other object. This allows users to freely manipulate and move the objects while simultaneously being able to see the links related to the object. An example of this is shown in Figure 36, where the different types of artifacts are moved closer together and the link endpoints have adjusted themselves to be on the side closest to the opposing model. It also shows how all different types of artifacts (2D, 3D, real or digital) can be linked together.

In section 2, we defined the different use cases of the system. For the practice of version visualization, we can generally define them as “visualizing versions for digital artifacts” and “visualizing versions for real artifacts”. Since both use cases have similar version visualization practices, we first define the practices used for the versions of the digital objects to which later we mention what differs in the real objects version visualization in comparison.



(a)



(b)

Figure 35: Visualization of linked objects in AR with (a) the user having the ability to show the links for model and (b) the link between objects that follows from pressing on the button.

4.3.1 Digital Artifact Versions Visualization

To visualize the versions of a digital object, users will have to press the first button on the artifact in Figure 35a. The versions visualization is very similar to the object relation visualization, since one object being a later version of another can also be considered a relation between artifacts. However, to ensure the models are visualized in the right order and to clarify which objects were discarded and which were selected, we put the versions visualization into a specific format called “branching”, which we have previously introduced in section 3.1.3.

Figure 37 shows the two types of branching that we support in our system, abandoned branches (a) and merged branches (b). Abandoned branches occur when multiple models are made in parallel and put into separate branches, but only one branch was selected to continue working with. Merged branches occur when multiple models are made in parallel and are, at some point, merged together so the changes of both models are incorporated into the merged design (like the workings mentioned in the git workflow). While the versions view is open, the specific version nodes cannot be moved separately from each other, to ensure the flow of the artifacts is always shown correctly. Inside the

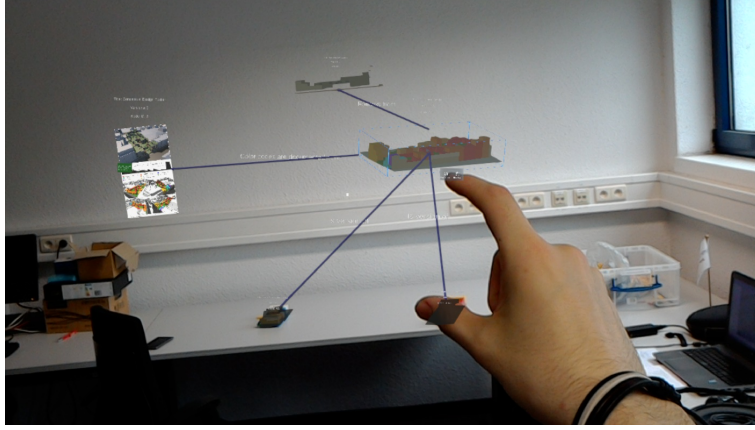


Figure 36: Visualization of different types of linked objects in AR.

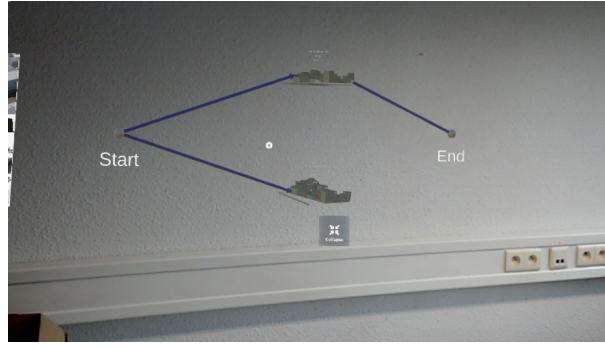
versions visualization, a “start” node and “end” node are added to clarify to the users where the versions view starts and with what version it ends, This feature is included because it is possible for any branch to contain the final model, so users need the ability tell where the final version is situated.

Inside the versions visualization for an artifact, the different versions and their attached annotations are shown in smaller size compared to the original model. Consistent with the definition of branching and semantic zoom we mentioned previously in section 3.1.3, the user can select a specific version node to retrieve the artifact that is portrayed by the node. In our system, users can do this selecting the object and then pressing the “Select” button shown on Figure 37b. The selected object will be presented separately from the version view, will become fully manipulatable (movable, scalable and rotatable) and can be explored using the same buttons as the other artifacts. In summary, version nodes are considered to be full-fledged artifacts themselves and can also have their own versions and defined links. However, an extra button “Go back” was provided to warn the user they are in a specific version and can go back to the full version view. A “Collapse” button is also provided in the versions view to allow users to go back to the original model that was explored. This button can be seen on Figure 37a and will always be present in the center underneath the versions view.

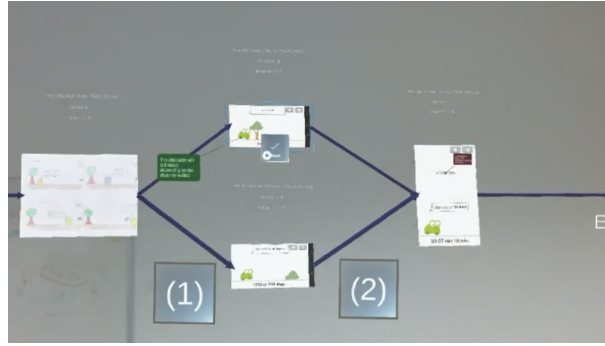
One final thing to mention for the versions visualization, is that every artifact, including the versions, has three text fields floating above them. These text fields are the name, the node ID and the number of versions of the artifact. The node ID is used to represent the order in which the artifact appears within the system. For example, imagine an artifact has node ID 5. This would mean that the first version of this model has ID 5.1 and the next model that was created has ID 5.2. In the case that model 5.2 has versions of its own, they will be labeled as 5.2.1, 5.2.2 and henceforth. Based on this information, users can deduct how deeply nested they have currently explored the model versions and in what order the models appeared in, even for parallel branches.

4.3.2 Real Artifact Versions Visualization

We have mentioned in section 2.2 that we support the alignment of digital models to a real artifact to communicate maquette changes to the stakeholders. An example of this feature can be seen on Figure 38a, where a digital copy of the artifact is aligned with the real artifact itself. In section 4.5, we discuss how we align the digital model to the real object, while currently we will only focus on the versions visualization for this model.



(a)



(b)

Figure 37: Branching visualizations of artifact versions with (a) the use case where two versions are made in parallel, but only one was selected and (b) the use case where two versions are made in parallel and both were selected and merged together.

Like mentioned previously, we use similar practices for visualizing real artifact versions compared to digital artifact versions. For example, we still make use of the branching technique to visualize parallel versions, we still use arrows to provide links between versions and we still provide the three text fields for the different version nodes. However, the branching visualization we use for this case is no longer along the y-axis, but instead along the z-axis and we do not use “start” and “end” nodes anymore. An example of this is shown on Figure 38c.

Between Figure 38a and 38b, the user has selected to look at the versions of the real model, to which the final version has remained on the position of the real model and the second to last version is visualized right next to it. If at this point, the user looks directly at the model and swipes to the right, the models will move towards the right as well which brings the visualization on Figure 38c. When the user decides to swipe left when looking at the visualization of Figure 38c, the models will move to the left and will go back the visualization seen in Figure 38b. To move to the objects seen in the back or front of the visualization in Figure 38c, users need to swipe sideways up or sideways down to the left, again in the opposite direction of the versions in the visualization.

In other words, this feature allows users to see the parts of the model that were once added or removed compared to the physical model. At all times users can only see one level next to the model that is currently aligned with the real object. This feature is incorporated because the visualization is entirely dependent upon the location of the real artifact. If there are a lot of objects right next

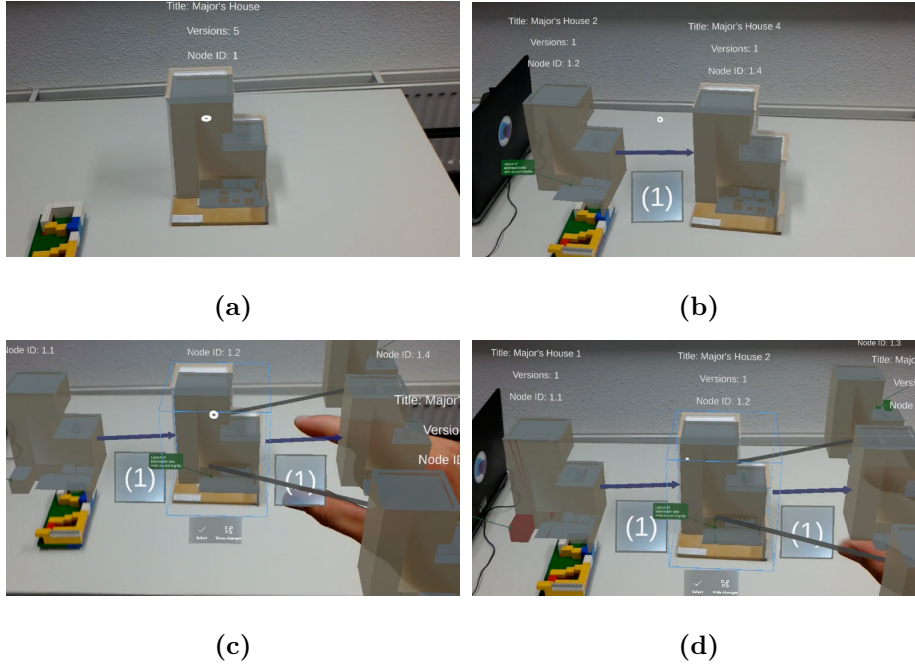


Figure 38: Versions visualization for a real artifact with (a) the digital object being aligned with the model, (b) the first version being aligned with the object, (c) the second version being aligned with the object and (d) the version changes being shown in red and green colors.

to the real artifact, too many sequential versions would clutter the view of the real environment or vice versa. Objects can still be selected separately for semantic zoom, but this time the selected object will be aligned to the real model as well, like seen on Figure 38a.

Users are also provided with the ability to “look at the changes” between the object versions (shown on Figure 38d). This means that, compared to the version that is currently selected, the previous version will show in red what was removed, and the next version will show in green what was added. The current version will also show in green what was added compared the previous version. Swiping along the models to align a different version with the model will also automatically update the colors to show the changes compared to the newly selected model.

This visualization allows users to instantly detect the changes that were present between the different versions of the model using only visual components. However, we can only apply this practice for the design changes of a real maquette since these model versions do not include a lot of drastic changes compared to other models.

4.4 Annotations and Decision Cards Integration

According to the guidelines mentioned in section 1.1, to provide artifact traceability for design rationale we do not only need to provide links between artifacts, but also between annotations and artifacts. Figure 39 shows how we have visualized this within our system. Each annotation consists of three parts, the annotation box containing the message of the annotation, a pointer that is attached to a specific position on top of the artifact and a line connecting the two pointers and the box. The user can drag and scale the annotation box similarly to normal artifacts, but the pointer will always remain stationary at one position on top of the model. When a user does manipulate the annotation box, the line will automatically update itself to ensure it always connects the two

elements. The color of the annotation can be changed to provide extra clarification by making them more distinguishable from other annotations or by providing a color that is linked to a specific part or change in the model (e.g. artifact component that was specifically added can have a green colored annotation).

Using the annotation box and the connection with the pointer, more detailed information is provided about specific parts of the artifacts, similar to the practices mentioned in the guidelines of section 1.1. Figure 39b shows how this can help for the architecture use case (section 2.2) to provide information about the real maquette that architects cannot include with the model itself. In Figure 39a, you can also see extra information being added about the infographic poster of an architecture project, which handles the spatial limitation problem we have mentioned in the use cases.



Figure 39: Annotations support for artifacts with **(a)** multiple annotations giving context on digital artifact parts and **(b)** an annotation providing context for a real artifact.

Another concept we mentioned in section 1.1 which we required within our system was the use of decision cards made by Gutierrez Lopez et al. These cards allow designers to highlight the decisions that were made for specific artifact versions. Figure 40c shows how we visualize these decision cards in AR. In general, every field that is mentioned in section 1.1 is included, except for the priority field. We choose to not include this field in the decision cards since the research conducted by Gutierrez Lopez et al. concluded that this field was vastly ignored by the test users and there does not seem to be an immediate reason for us to include it in our research either [20].

In general, decision cards can be used to log the design rationale for important decisions between versions. Since we already visualize all these versions and link them together, it becomes possible for us to include decision cards within the links. For example, Figure 40a shows two buttons, one underneath the link between the first two models and one underneath the link between the last two models. The numbers on top of the buttons represent the amount of decision cards that are present between the two versions. As shown in Figure 40b, when the user hovers with the cursor on top of the button, two small cards are shown with the titles of the decision cards that are present within the button.

When the user clicks on the button, the small cards become highlighted and the decision cards are shown instead of the normal version link (to avoid cluttering the visualization with too many lines). This is visualized on Figure 40c. When the user presses one of the two small cards underneath the button, it will no longer be highlighted and the decision card corresponding to the small card will be hidden. In other words, it allows users to filter the decision cards that are currently being shown. This can be useful in case the decision cards become too obtrusive when analyzing other decision cards, annotations or models.

The small cards underneath the button are used to give a quick overview of the decisions present within the artifact versions. When users decide they require more information about a particular decision, only then do they need to open and explore the full decision card.

Instead of annotations that can only be linked to one specific part of one artifact, in our use case, decision cards can be linked to a specific part on two artifacts. In case the decision mentions a change in a specific part present on both versions, the decision card can start the link from that part in the first model and then point towards that same part within the next model. This complies with the practice of “linking” that was mentioned in the guidelines of section 1.1, giving us reason to believe that we can provide better traceability means using this practice.

4.5 Real Object Integration

To overlay information about a real model in our system like mentioned in section 2.2, our system will first need to be able to recognize what model is currently being looked at, what the features of the model are and where the model is located. However, putting this into practice is already an ongoing problem in computer vision where no perfect solution exists for yet. An example research that attempts to provide such detection and alignment was conducted by Szemenyei et al. where they optimized the 3D object detection for tangible Augmented Reality. In their research they were able to detect the types of surfaces on primitive objects, classify them and link them together [36]. However, for our use case, we need to detect specific artifacts that are being looked at so information about these artifacts can be retrieved from the server and overlayed on the AR system. These artifacts can include both 2D applications or 3D objects.

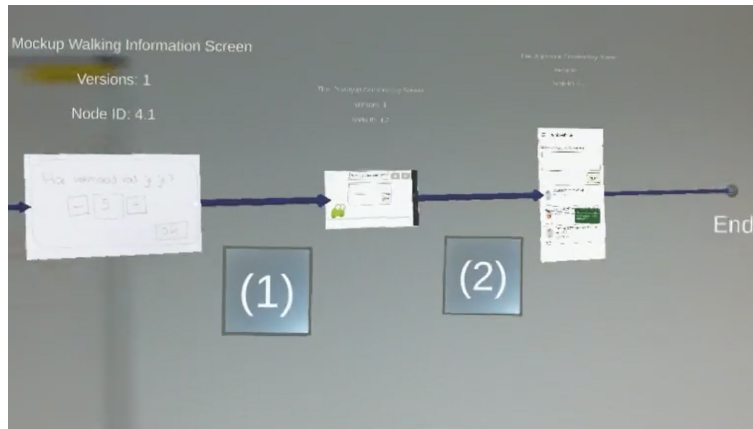
In our research we want to focus on the visualization and traceability aspects of using Augmented Reality, so instead of building our own type of object detection algorithm, we simulate the object detection feature. For this we make use of the Unity library called “Vuforia”¹⁹. Vuforia allows developers to store images and objects in their database and provide image tag and 3D object detection in their AR system for these stored objects. For the 3D object detection, users are required to scan all the different features of the object, while the image recognition only needs the user to look at a 2D tag somewhere in the environment. Compared to the image tag recognition, the object detection is a lot more unstable and does not provide the exact location of the object, so for research we make use of the image detection feature.

We use the image detection in two ways within our system: detecting 2D application screens directly and detecting specific 3D artifacts using image tags. To provide 2D application screens detection, we store screenshots of the application on the Vuforia database. When a user looks at one of the specifically stored application views using the AR system, the system will detect which model was recognized and visualize it in front of the user as a digital copy. This is visualized on Figure 41. We do not align the digital copy of the application with the smartphone, since users would have to move closer to the phone to explore the application view and read the text written on the views.

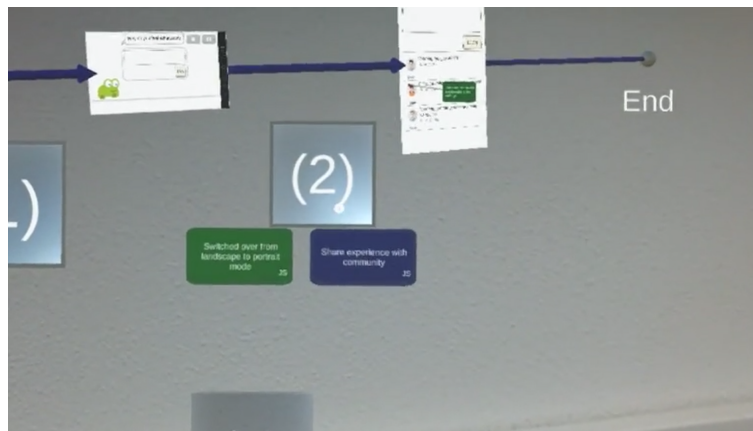
If we had chosen to not visualize the digital object, but only the annotations on top of it, the clipping plane mentioned in section 4.2 would have made the annotations disappear when the user moves too close to the model (shown on Figure 41a). For this reason, it is more efficient to show a digital copy for application views that can be freely manipulated like all of the other digital holograms.

However, for the architecture use case (section 2.2) this is not as prevalent. Here we want to specifically support alignment of the digital object to the real model, to provide more detailed information on top of the model or showing how the model is linked to different artifacts in the project. An example of this use case can be seen on Figure 42. The tag on top of the model is only an example image tag that can be used to align the object.

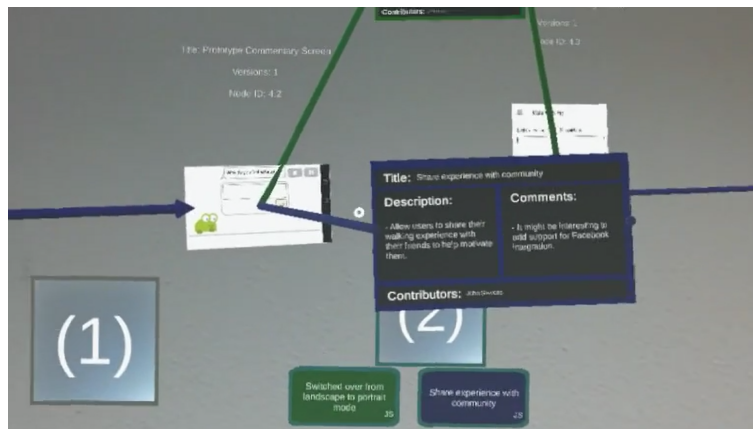
¹⁹Documentation of Vuforia for developers: <https://library.vuforia.com/getting-started/overview.html>



(a)



(b)



(c)

Figure 40: Decision cards visualization with (a) the overview of buttons underneath links, (b) overview of available decision cards for button and (c) decision cards being shown between the models.



Figure 41: Application view detected by the system using Vuforia with (a) the clipping plane not showing the full screen close-by and (b) the detected screen fully shown in front of the smartphone.

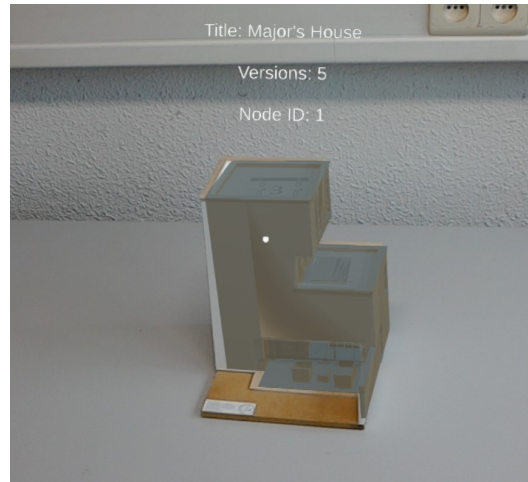


Figure 42: Digital object aligned with a real model

To do this alignment, the image tag is placed on a predefined location on top or near the physical model and the project configuration on the server stores the precise location of the object relative to the image tag. By using this method, we can ensure that the digital object will be positioned correctly on the real model. As soon as the digital object has been aligned correctly, the tag can be removed from the object since it is no longer required to track the real model. We use this approach since for some models, the tag can occlude the view of the real model, which we want to avoid.

However this does mean that when the physical object is moved or rotated, the digital object will not move or rotate along with it. In the architecture use case mentioned in section 2.2, this does not pose much of a problem, since maquettes are generally not allowed to be touched to avoid spots appearing on the model and to avoid breaking parts of it²⁰. To ensure the digital model remains aligned with the real model, we did not make the digital model manipulatable like the other digital objects in our system.

Because the outside of the model is fully closed off and since the model should not be touched, it becomes difficult for stakeholders to explore the insides of the model. For this reason, we have

²⁰Maquette practices can be found on: <https://www.rjmodels.com.hk/architectural-models-guide/>

included the ability for users of the system to look inside of the object. However, while we say we “included the ability”, we did not make this implementation ourselves. In section 4.2, we mentioned that in the hardware we use, there is a clipping plane problem that makes parts of objects disappear as soon as they are too close to the user. We make use of this bug and turn it into a feature. As soon as users come closer to the model, the walls will disappear, revealing the inside of the model. This visualization is shown on Figure 43. The only thing we had to provide for this to work, was to use the entire design model of the maquette instead of only the outside appearance of the model.

Using these methods, it becomes possible for stakeholders to find out more about the different parts of the model, both inside and outside. The digital model can sometimes even include parts that were not incorporated in the real maquette due to time, resource or feasibility constraints. In Figure 42 this can be seen by the couches that are visualized on the digital model, but are not included in the real maquette.

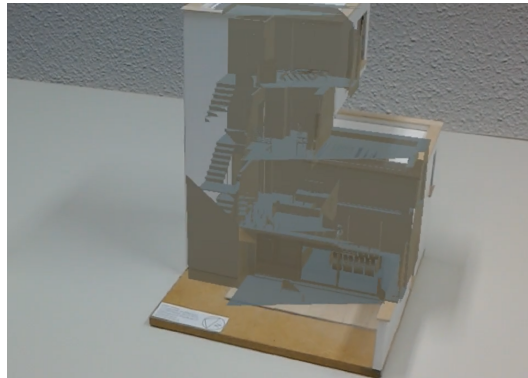
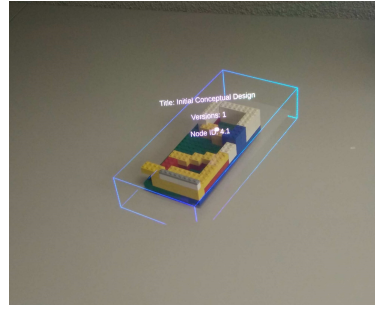


Figure 43: Inside of the model reveals itself when users approach the model.

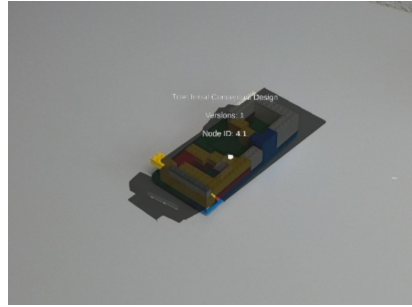
In section 2.2, we have also mentioned the required ability to consider a real model as part of the system, similar to how a digital model is supported. This use case can be seen on Figure 44a, where only the information about the model can be seen and on Figure 44b where is shown that model can even be selected by users. This functionality works like the previous object alignment practices, since an image tag is also used to align a digital copy of the object with the real model. However, we have made the aligned digital model invisible for the user by giving it a black texture color (like seen in Figure 44c). This works because in Microsoft Hololens, a black color will be rendered as semi-transparent, making it so users cannot see the model when the brightness of the Hololens remains low. When users move the cursor over the object, it will also move along the edges of the digital copy, which allows us to let the model be physically part of the system without having to analyze the actual meshes of the object in the environment.



(a)



(b)



(c)

Figure 44: A real model as part of the system with (a) the information about a real object being shown, (b) a real object having been selected and (c) the visualization of how the model is interpreted by the system.

5 Computer Science Study

We have previously mentioned in the use cases of section 2 that we will cover artifact traceability for the design rationale of software GUI designs. To evaluate the use of our system for this use case, we need to carry out a study where the system is used to trace the design decisions in a project using User-Centered design practices. For this reason, our testing demographic for this study consists of master students in computer science, since they have already experienced the aspects of software design and development in past courses.

For this study, we will make use of the “Walk with Me” project mentioned in section 2.1, since it applied the required User-Centered design practices. This means that for this study, the project models will consist of the artifacts seen in section 2.1. These artifacts are the application views consisting of low-fidelity prototypes, high-fidelity prototypes and a final view as versions.

There are several elements that we need to evaluate within this study, before we can decide our system to be practical for the covered use case. We first discuss these elements and what hypotheses we make for them, so we can highlight later whether our assumptions about the system were correct. Afterwards we discuss what procedures we used to carry out and evaluate the study and finally, we mention what we have observed during these tests and what we can conclude from these findings.

5.1 What to evaluate

We specifically want to discover the practicability of artifact traceability practices in an Augmented Reality environment. Within this study, these practices will be applied for the field of computer science and more specifically, the design rationale of software GUI designs (see section 2). We have highlighted in previous sections, that we make use of decision cards, links and annotations to provide information on the different artifacts, their versions and their relations. One of the first things for us to test, is whether our implementation for these techniques are able support the participants in tracing the design rationale of a project. For this reason, we make several hypotheses about our expectations for the evaluation of the implemented techniques.

For the decision card format in our implementation, there are a lot of hypotheses that we could make since this element consists of several components. However, we limit ourselves to one general hypothesis, since we only want to test the effectiveness of the artifact traceability practices as a whole. This general hypothesis is the following:

H1 The decision cards provided support for tracing the design rationale.

We have selected this hypothesis, since it provides the most context on traceability practices and naturally includes the extensions we implemented compared to the original decision cards format. In other words, this hypothesis allow us to evaluate the effectiveness of our decision cards implementation. Other hypotheses could have included the amount of times participants opened decision cards and whether they start with reading the description or the title to make assumptions on the changes. However, these findings include the ability of decision cards to support tracing the design rationale, so we will not focus on them specifically but rather keep them contained as the general hypothesis on providing support for tracing the design rationale.

A second component that we want to evaluate for our system is the use of branching for the versions visualization of the artifacts. This visualization gives participants the ability to find out what versions were created for an artifact and in what order they were created. In this study, these versions will specifically consist of low-fidelity prototypes, high-fidelity prototypes and a final version, meaning we can focus our hypotheses on this demographic too. This brings us to the following hypotheses for this topic:

H2 The links between the versions provided support for understanding the different phases of the application views.

H3 The semantic zoom provided support for understanding the models that are portrayed in the versions.

We only covered these two hypotheses, since they are the most important components of the branching visualization. They allow us to evaluate whether the branching technique is useful for the visualization of User-Centered design process phases. If the findings do support these hypotheses, they can bring interesting design implications for future traceability systems for this use case. Especially since to our knowledge, no prior research has covered the use of branching for this use case before.

One final component to evaluate for our system is on the observed benefits of applying Augmented Reality for the artifact traceability of a User-Centered design process. Currently it still remains unclear whether AR technology is more practical compared to traditional systems for supporting the design rationale tracing in the sector of computer science. Based on the findings for this component, regardless of the outcome, we can make several implications that can be of relevance for future research, since the use AR is only becoming more widespread over time [4]. This brings us to the following final hypothesis to evaluate for this study:

H4 Augmented Reality provided support for tracing the design rationale of software GUI designs.

5.2 Study Setup

In Figure 45 you can see the full setup of the computer science study. To effectively test the hypotheses, there are several approaches that need to be taken here. The general procedure of the study is as follows: Upon entry in the testing room, the participant is presented with two consent forms (Figure 45, "Papers of relevance") that they need to read and fill in (Appendix D). One copy of the consent form is for the participant and one is for the facilitator. After filling in the consent form, the participants are prompted verbally with four questions. The first question is on whether the participant knows the concept of AR and how they are familiar with it. This question is asked to estimate the participant's experience level and potential bias towards the use of AR. The second question is whether the participant has any prior experience using an AR system like Hololens. This question allows us to ensure the right conclusions are derived about the interactions, since participants with more experience might get adjusted to the system faster. The third question prompts the participants whether they are sensitive to epileptic visuals, which is asked to ensure no accidents will occur during the usability test. The final question consists of the facilitator asking the participant on whether they know the approaches used in User-Centered design processes. For this question they are provided with a sheet of paper explaining the MuiCSer framework shortly (also part of papers of relevance on Figure 45). This is done to ensure all participants are aware of the concepts and goals of low-fidelity prototypes, high-fidelity prototypes and final version (see Appendix E).

After these steps, the test works as follows: The participant is introduced with the procedures of the usability test and are asked to apply the "Think Aloud" protocol, which implies that the participant should try to word out as much as possible what they are thinking during the usability test [24]. After the introduction, participants are requested to stand up to which they are provided with a sheet of paper ("Testing Instructions" on Figure 45) containing the steps that will be taken during the study (Appendix F and G). Both the facilitator and the participant have a copy of this sheet, so the facilitator can go over the steps verbally while the participant can double check in case they forget what they must do. The facilitator also has a separate document containing more detailed descriptions about the study, including the answers to the questions that participants will need to solve (Appendix H).

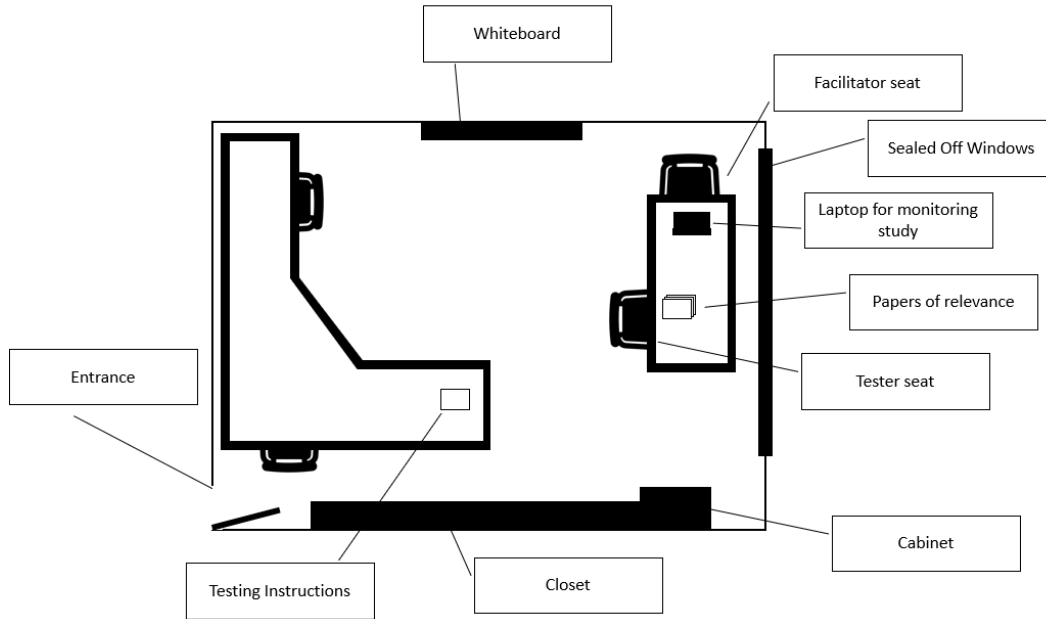


Figure 45: Computer Science Study setup

While the participant and the facilitator are discussing the different application scenarios and the system, the facilitator will continuously be keeping notes on any of the important behaviors of the participant and what discussion points are mentioned by them. However, the footage of the test will also be recorded by the Hololens to ensure no points are missed in the post processing phase of the study. The notes taken by the facilitator will only serve as a guideline for locating important mentions done by the participant, that can be re-watched using the video footage.

Because the system requires the use of the front camera and because of limitations in the Hololens application regarding the use of the front camera, the facilitator is not able to see the live footage of what is happening in the system. For this reason, to help support the note taking and guidance of the facilitator, a logging component was added to the server and the system that sends any event happening in the system. The facilitator always has a separate window open that visualizes the output of the server. This way the facilitator can know at any point, what objects have been pressed, manipulated and what other actions have been carried out.

To introduce participants to the concept of decision cards, annotations and the versions visualization, participants are first provided with a training phase (Figure 46). This way we can also avoid that a learning effect occurs for the AR interaction, which could create biased opinions when participants are analyzing the scenarios of the actual project. The steps of the training phase are deliberately chosen so participants can practice the use of the gestures, discover that they can manipulate artifacts, annotations and decision cards and so they know that they can manually filter and visualize the decision cards.

When participants are done with the training phase, they are requested to deliver the Hololens back to the facilitator so they can start the program with the actual project scenarios integrated. When the facilitator gives the Hololens back to the participant, they are introduced to background information about the “Walk with me” project and how the study will proceed. The project background information only covers the goals of the application (rehabilitating patients suffering from stroke) to avoid mentioning anything on the important decisions that were made. The study

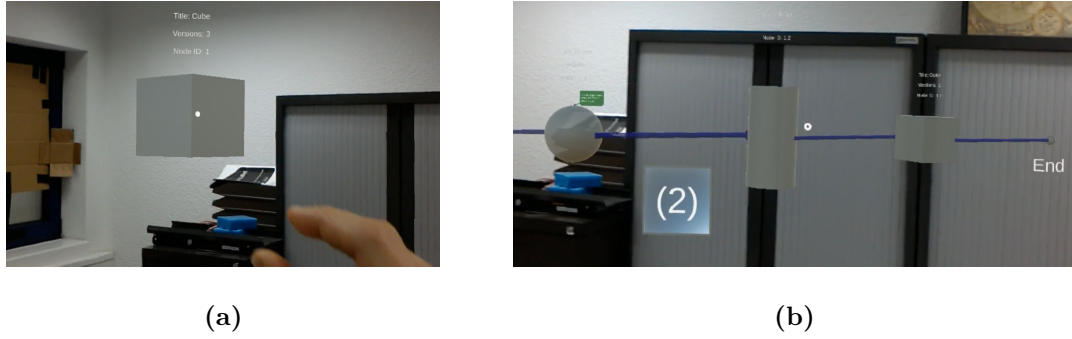


Figure 46: Training phase with (a) a cube as the final model and (b) the versions of the model.

procedure is mentioned to the participant to go as follows: The participant is requested to analyze the different application views that will be presented before them. The application views will have versions that correspond to the practices of low-fidelity prototypes, high-fidelity prototypes and final versions that are mentioned on the MuiCSer page presented to them previously. The analysis of these views needs to cover three important parts. One, the participant needs to explain what changes they notice between the different application views. Two, the participant is requested to mention the important decisions that seem to be present in the application views. And three, the participant needs to mention what the reasons for these decisions could be. These steps are also presented on the sheet of paper that both the facilitator and the participant possess (seen in Appendix G)

To load the different scenarios, participants are requested to look at the direction of the whiteboard on Figure 45 and are presented with an application view visualized on a smartphone that is being held by the facilitator. When the participant gets close enough to the smartphone for the system to recognize the application view, a digital copy of the current view will be visualized in front of the participant. From this point onward, participants are free to manipulate and take any approach in finding the different changes and decisions between the application versions. We mention they are free to use any approach, so interactions are left as open as possible and no biased results will occur due to enforced commands.

To evaluate the usability of the system for developers, after the test participants are prompted with the question “Do you think a system like this can be useful for developers?”. Asking this question to the participants after they have experienced using the system for themselves, might provide us interesting use cases or critiques that should be considered when fully developing such a system for production environments.

However, to allow us to fully evaluate the uses of the decision cards, versions visualizations and AR, the scenarios need to include special cases that can enforce the decision making of the participants and empower possible discussions involving the system. We describe these different scenarios as follows.

5.2.1 Application scenarios

Like we have mentioned previously, during this study we will use the “Walk with Me” application covered section 2.1 (see Appendix A for full screenshots). The application design process of “Walk with Me” consisted in total of 23 screenshots of application views and prototypes, but to avoid spending too much time in the usability study we only let participants explore half of them which created four different scenarios (the ones seen in section 2.1). Spending too much time in the

	Scene	Scene	Scene	Scene
User 1	1	2	3	4
User 2	2	4	1	3
User 3	4	3	2	1
User 4	3	1	4	2

Figure 47: Latin Square that matches the participant with the order that they will cover the scenes

usability study could cause exhaustion and frustration for the end participants, which we wanted to avoid at all costs.

Since there was no real use case for semantic versions (versions within other versions) in the application, participants only had to explore the versions view of the main application. In this application, there are three important factors that play a role in the evaluation of our systems. First off, Scene 2 is the only scene in the application where two versions are provided for the high-fidelity prototype. This is the case because both versions were used in unison to explain the concept of a character moving in the application view and a statistic being incremented meanwhile. In our system, the two high-fidelity prototype versions are visualized as parallel branches, meaning it will be interesting to see whether participants understand what the parallel branching visualization implies. To ensure we make no mistakes in assuming whether participants truly understood the visualization, after they are done exploring Scene 2, we prompt them on what they thought that the two branches meant. We do this by asking the question verbally and noting it down in the study notes. A second important factor that plays a role in the evaluation of our system, is that

in Scene 3, we purposefully left out a decision card between the last two versions, while there were important changes between these two versions. These changes are even consistent with the other changes throughout the application. We applied this practice since it allows us to explore how participants respond to the absence of a decision card and whether they can still find the important changes between the models. To ensure we do not make the wrong conclusions based on the results, we will prompt the participants after they are done exploring Scene 3, whether they think there should have been a decision card between the final two versions. Like in Scene 2, this will also be done verbally and will be noted down in the study notes.

A final important factor to consider is the versions of Scene 4. The three versions that are presented for the application view in this scene look quite different in functionality, but each of them arose after analysis of its previous version. Compared to the other scenes and application views, the changes of this scene are more discrete and it will be interesting to see whether participants can understand the link between these versions using the decision cards and the branching visualization.

During the usability test, the order of the scenarios that the participant had to cover were based on a 4x4 Latin Square ²¹. This is a method used often in statistics to ensure participants cover tests in a randomized unbiased format, which is generally used to prevent a learning effect. However in our case, we use the Latin Square to ensure traceability is correctly provided for all scenes (aside from those purposefully altered). Some decisions in the application occur multiple times and participants should be able to find the changes within a scene, regardless of which scenes they covered prior to it. The Latin Square for our system can be seen on Figure 47. In case there are more than four participants tested in the system, the Latin Square will be repeated from top to down (participant 5 will have the same sequence as participant 1).

²¹Website explaining Latin Square: <https://www.statisticshowto.datasciencecentral.com/latin-square-design/>

5.2.2 Questionnaire

Aside from asking participants questions during the usability test itself, after the test we ask them to fill in a questionnaire (part of “Papers of relevance” in Figure 45), where they can rate the system using a Likert scale²² for different types of categories (see Appendix I). This questionnaire is equal to the System Usability Scale (SUS)²³, which is often used in the fields of computer science to provide a reliable measure for a system’s usability. In this scale, participants are presented with ten questions that keep shifting between positive and negative wordings about the system experience. We choose to use this scale, since it is a light weight format that produces reliable results even for small sample sizes and allows us to calculate a score that can be compared to the averages of other tools and technologies. To calculate a score for this, every Likert scale result for odd numbered questions need to be done minus one and every even numbered question needs to be done five minus the score of the Likert scale. In the end all newly calculated scores are added together and multiplied by 2.5 to get a total score on 100. A score above 68 is generally considered to be above average and anything below 68 is considered to be below average.

Aside from the Likert scale questionnaire, the participants are presented with the following three follow up questions: Whether the participant still has any remarks for any of the given scores in the Likert scale, whether they have any recommendations on how the system can be improved and whether there are any other final remarks they would like to give. Asking these questions attempt to fill in any gaps in information that were not provided during the discussions of the actual test, which supports us in the evaluation of the system afterwards.

5.3 Observations and Analysis

In total we had eight Computer Science students that participated in the usability tests. The findings for the field of Computer Science can be categorized into three different parts: artifact traceability user methodology, practicability of artifact traceability in AR and the system usability scale evaluation. We discuss them here as follows.

5.3.1 Artifact Traceability User Methodology

This category consists of how participants interacted with the system to solve the questions in the usability tests. This means more specifically, what methodology did they use to find the changes within the application. A first topic to consider is on how participants used the different versions and decision cards in the system to find solutions to the questions. We never specified to the participant on how to find changes in the application views or how to use the decision cards specifically, only what information is provided on it. This means that they were open to take any approach on finding the changes.

While we never asked the participants in what order they should discuss the artifacts, six participants chose to discuss the artifacts in chronological order from first to last (on Figure 49a from first view to middle to last). When going through the artifacts chronologically, they would always discuss what has changed in the current view compared to the previous one. While the other two participants did not look at the versions in chronological order, one of them did select each of the versions separately to analyze as well.

This leads us to believe that when trying to find the changes in the different iterations, participants have a tendency to look in order from start to finish and tackle every individual change first before looking at the version changes globally. This is also backed by the fact that not a single participant has attempted to first find the changes by tracing a singular feature from start to finish

²²Likert scale explained by SimplyPsychology: <https://www.simplypsychology.org/likert-scale.html>

²³System Usability Scale explained on usability.gov: <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>

and then moving on to a next feature. It would be interesting to consider if participants would behave similarly, when a visualization was provided showing the different features separately from start to finish. However, right now we can only speculate on this behavior until this has been tested.

In the first iteration, six participants attempted to find the changes themselves before reading the decision cards. As shown in Figure 48, four of these participants did not notice all the changes in the application views and required the decision cards to find some of the changes. As a result, in the following iterations more participants started using the decision cards instantly to find the changes without having to look for them themselves. Five participants even made the following their methodology to discuss changes in the system: Look at a version separately, look at the decision card between this version and the next (shown on Figure 49a), look at the next version and repeat (like in Figure 49b).

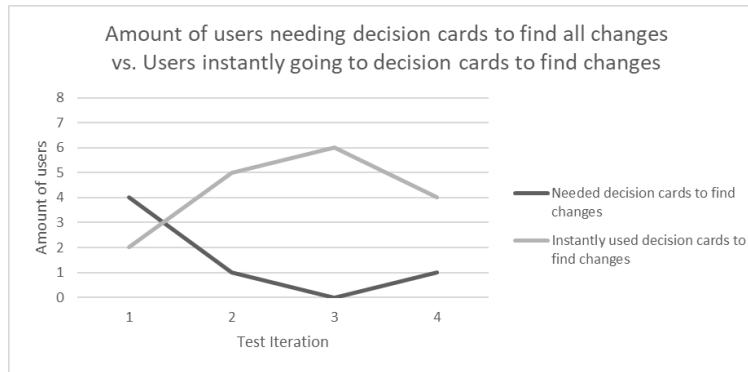


Figure 48: Comparison graph of needing decision cards vs. instantly using decision cards

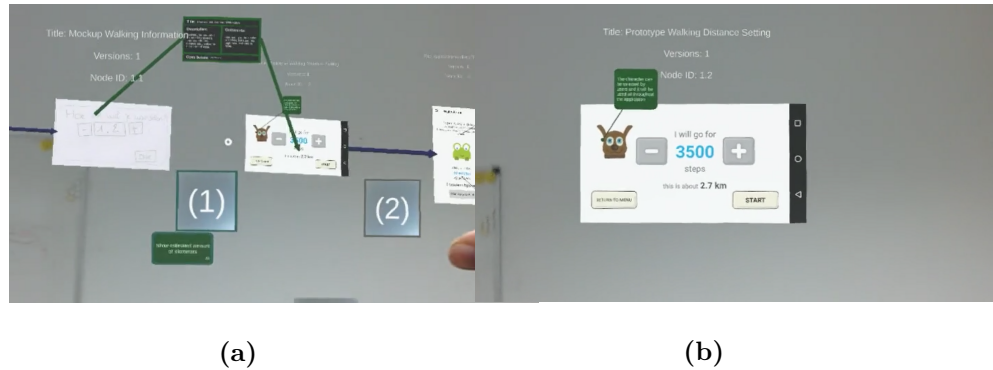


Figure 49: (a) User looking at the decision card and versions of Scene 1, (b) User looking at the second version of Scene 1.

This implies that participants found the decision cards to be useful for logging the decisions and were naturally drawn towards using them to find and explain the changes between models. To further support this theory, during the test all participants eventually understood the changes and design rationale mentioned on the decision cards (including the discrete decisions of Scene 4 mentioned in section 5.2.1). However, when a decision card was not present between last two versions in Scene 3 (shown in Figure 50), three participants had failed to find the changes between

these versions in which two of them were supposed to tackle this scene first in the list of scenes (due to Latin Square). These two participants even had a harder time understanding what they were supposed to be doing in this scenario when prompted to explain the changes between the models.

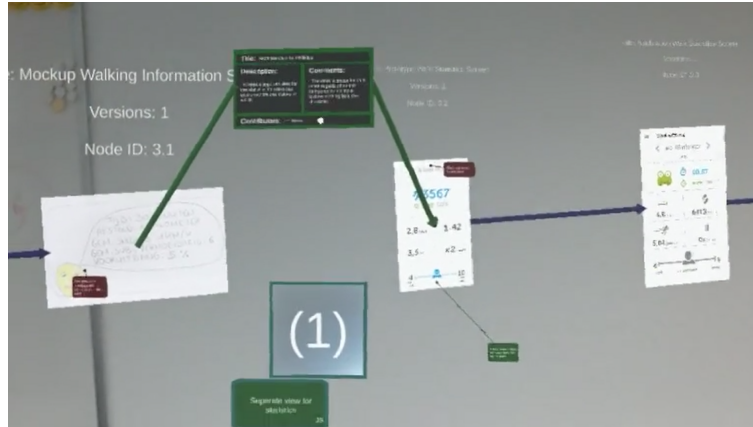


Figure 50: User looking at the decision card and versions of Scene 3

While this may have been a participant dependent coincidence, all eight participants initially expected there to be only small changes between the last two versions in Scene 3. Three of these participants (different participants from the ones that could not find the changes) were also met with confusion and even started questioning themselves whether they found all the changes in the versions. Like mentioned in section 5.2.1, we asked the participants whether they thought there should have been a decision card between the models of Scene 3. When prompted with this question, four participants mentioned they found it necessary that there was a decision card. This was either because the changes were in their opinion significant or because they found it easier to find changes when there was a decision card.

These findings seem to indicate that participants become reliant on decision cards to help them find the changes between the versions. A lack of decision cards also seems to be met with confusion and failure to understand all the changes both when starting off without them and after already being reliant upon them. Based on these findings, we accept hypothesis **H1** stating “*The decision cards provided support for tracing the design rationale.*”. However, provided with these findings, for hypothesis **H2** stating “*The links between the versions provided support for understanding the different phases of the application views.*” no solid conclusion can be derived. Participants displayed the ability to understand the meaning of the links and they were successful in understanding the changes between the phases of the application views, however these findings do not indicate anything about the use of the links specifically. For this reason, we will neither accept nor reject hypothesis **H2**, meaning more research will need to be conducted on the uses of the version links.

The findings do allow us to make speculations about the semantic zoom. Since the semantic zoom was naturally used by participants for exploring the versions more in depth, we can accept hypothesis **H3** stating “*The semantic zoom provided support for understanding the models that are portrayed in the versions.*”. However, this behavior might have been caused by the inability to scale, move and rotate the versions within the versions visualization, meaning objects could only be scaled by selecting them. Two participants have mentioned they would have liked the ability to manipulate the versions inside the versions visualization to make comparing the models easier. This indicates there might be value in implementing this as well. For this use case, it would be interesting to observe whether participants would prefer manipulating the models in the versions

visualization or would choose to explore them separately like they did now. However, we can only speculate on this behavior until this is tested in future research.

5.3.2 Practicability of Artifact Traceability In AR

To be able to analyze how participants evaluated the practicability of the system and the use of AR for developers, we prompted them with the question “Do you think a system like this can be useful for developers?” (see section 5). For this specific question, we received the following feedback from the participants.

All the participants found the annotations and decision cards to be useful for logging the changes between the different artifacts. Two participants mentioned that the system gives a good global overview on how the applications were made and three participants found the system to be specifically useful for when there is someone new joining the company, since they can easily be brought up to speed on how the applications were made. In total, three participants also mentioned the system to be useful for avoiding a clutter of information by being able to look at each screen individually and find more information about it. Two participants even saw a lot of potential for collaboration by allowing participants to work together with such a system in a shared environment.

These mentions already seem to imply that participants found the artifact traceability aspects to be useful for providing team communication within projects. These findings also conform with the guidelines mentioned in section 1.1. However, in total there was only one participant that found AR to be specifically useful for developers. This was because he liked to work on paper and the gestures used in AR to move objects around brought a similar sense of interaction freedom compared to paper interaction. Three other participants mentioned that an AR system brought overhead for viewing and interacting with 2D views, since they could be shown on a computer screen too where people tend to interact faster with. These participants saw more potential in using the AR system for 3D objects, since this helps with the spatial reasoning of the objects. One participant even specifically mentioned the use case of companies with more complex systems where an AR visualization would probably introduce a lot of information clutter.

Because the findings indicate that AR tends to bring overhead for the interaction and barely any positive remarks were mentioned for the use of AR within this sector, we reject hypothesis **H4** stating “*Augmented Reality provided support for tracing the design rationale of software GUI designs.*”.

5.3.3 AR Limitations

During the study, we have made several observations on how participants make use of the AR environment. Seven of the participants had no prior experience to using Hololens or any other AR headset, while the other participant had used and developed for Hololens before. Of these seven participants without experience, four participants had trouble using the gestures at first. Three of these participants had stopped experiencing trouble after they were done with the training phase. The fourth participant had decided to not scale or rotate any objects further since he did not feel the need to. This shows that while some participants had problems using the manipulation gesture at first, after becoming more adjusted with the system this posed less of an issue.

Another finding we made is that, during the usability test, four participants had accidentally bumped into a physical object (chairs next to them while looking at objects or the closet behind them when moving backwards). Three participants even had to physical move themselves to different parts of the room to look at specific information on a hologram. These cases all happened when versions of a hologram were projected beyond the walls of the room. This shows that, during interaction with the AR system, participants are sometimes forced to interact with their physical environment as well, whether it is intentional or by accident.

We have mentioned previously in section 4.2, that we use the first edition of the Microsoft Hololens to conduct our studies. However, there are some known limitations of this edition, which were also prevalent during our tests. One example of these issues is the field of view, which caused three participants to not realize that decision cards had appeared when clicking on the visualization button for the decision cards. Furthermore, three other participants experienced trouble finding the “Collapse” button of the versions since it was also outside of the field of view. Another issue that was prevalent in our studies is the distance of the clipping plane. Three participants mentioned that it was annoying for reading and exploring the objects, due to parts of objects disappearing when coming too close to the Hololens. For these reasons, it is important to notify users on the locations of specific objects and to provide ways for scaling and rotating the objects to allow users to see and read the parts of interest.

Finally, during the usability test none of the participants mentioned exhaustion in the arms from using the gestures. However, three users did mention the Hololens to be exhausting to wear and mentioned that they would not want to wear it for a long time. One participant even mentioned that his eyes were starting to hurt and that he was getting a minor headache from looking at the holograms. This participant compared the sensation to the feeling of looking at a bright display from a short distance for a prolonged amount of time.

All of these findings indicate that there are still limits to the current AR system and improvements still need to be made to make the system more user friendly and less exhausting to wear.

5.3.4 System Usability Scale Evaluation

During the usability tests, we asked the participants to give feedback on the system by asking questions directly and by letting them fill in a questionnaire (Appendix I) where they could rate our system using the System Usability Scale (SUS)²⁴.

Knowing the participants were in favor of the system, but were critical on the use of AR, we now take a look at the ratings they gave in the questionnaire (full results found in Appendix J). In Figure 51, you can see the different results for the System Usability Scale that we base our deductions on. The total mean score of the system ended up on a 70.94, which is a little above the average mean SUS score of 68. This means that the system was perceived to be generally adequate, but needs to have some changes.

On Figure 51b, we see that the questions “I think that I would like to use this system frequently.” and “I felt very confident using the system.” were scored the lowest on. These findings might imply that participants are not comfortable yet in using AR frequently. This implication is backed by the findings of the previous topic where participants mentioned the AR interaction to cause overhead, but mentioned the traceability part of the system to be well integrated (which is backed by the high score on the question “I found the various functions in this system were well integrated.”).

For this reason, the changes that we would have to apply to our system for developers, would most likely have to be either removing the AR component of the system entirely or improving the interactions with the objects significantly.

²⁴Documented website about the System Usability Scale: <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>

User	SUS
1	72.5
2	92.5
3	90
4	47.5
5	67.5
6	62.5
7	72.5
8	62.5
MEAN	70.94

(a)

Question	Mean
I think that I would like to use this system frequently.	5.625
I found the system unnecessarily complex.	7.5
I thought the system was easy to use.	7.1875
I think that I would need the support of a technical person to be able to use this system.	7.8125
I found the various functions in this system were well integrated.	8.125
I thought there was too much inconsistency in this system.	8.125
I would imagine that most people would learn to use this system very quickly.	7.1875
I found the system very cumbersome to use.	6.25
I felt very confident using the system.	5.9375
I needed to learn a lot of things before I could get going with this system.	7.1875

(b)

Figure 51: (a) SUS results for every participant and the mean SUS score, (b) Mean SUS score for every question.

6 Architecture Study

We have previously mentioned in the use cases of section 2 that we will cover artifact traceability for architectural building projects. To evaluate the use of our system for this use case, we are required to cover both the tracing of maquette versions and providing an understanding of architectural project structure. Since both scenarios are relevant for the sector of architecture, we carry out one study that covers both together, where our testing demographic are students and researchers in architecture. We choose this demographic, since they are experienced in building maquettes for stakeholder communication and are often required to analyze and keep track of project structures.

To perform this study, we will make use of the scenarios mentioned in sections 2.2.1 and 2.2.2, since they already provide the different models and scenarios required to test the system. Similar to the study in 5, we first discuss the hypotheses that we want to test for the system before discussing the procedures we will use to carry out and evaluate the study. Afterwards will discuss the observations and conclusions we made for this study.

6.1 What to evaluate

We will generally use the same approach as the one in section 5.1 for defining what we will evaluate within the study. However, for the architectural study, we need to evaluate some other topics than the computer science study in section 5. While the main concept remains the same, the use cases for architecture are vastly different from the ones seen in computer science. For example, within architecture the real models play a more vital role since they are essential for exploring the applicability of a project. For this reason, one of the first questions we can ask is regarding the understandability of the integrated real models in the system. This brings us to the following hypotheses:

- H1** The annotations provided the required amount of information to understanding the model components.
- H2** The digital models overlayed on top of the real models provided additional information about the model components.
- H3** The versions visualization provided the information required for the participants to understand the structure and history of the maquette versions.
- H4** The colors of the changes visualization provided the information required for the participants to understand the changes between the maquette versions.
- H5** The links provided clarifications on the relations between the digital models and the real models.

We have selected these hypotheses, since they provide us the most context on the use of the system for understanding the components, relations and versions of the real models. Since these hypotheses are on the integration of the real model within the system, they are automatically also evaluations on the use of AR for the field of architecture. This is because AR is the technique we employed to be able to include the real models in the system in the first place. Consequently, accepting or rejecting these hypotheses will allow us to evaluate the feasibility of the core AR aspects of our system for the architecture use case, which allows us to make a conclusion for the following general hypothesis:

- H6** Augmented Reality provided support for tracing the changes and relevant information of real objects.

Like seen in section 2.2, the decision cards in our system are also part of covering traceability for the artifacts of architecture. This is not the use case that the decision cards were intended or focused to be used for, but we still attempt to apply them to the architecture use case. This way we can evaluate whether they are usable for this field too. For this reason, we also need to explore the general hypothesis on the feasibility of decision cards for the architecture use case., which gives us the following hypothesis:

H7 The decision cards provided support for tracing the decisions and changes in the artifacts.

Depending on how the decision cards are used by the participants of the architecture fields, some parts may be considered useful and important while others might not. Based on the findings involving these hypothesis, we should be able to get a better understanding about the feasibility of the decision cards for the field of architecture and changes that might need to be applied to the format.

6.2 Study Setup

Figure 52 shows the study setup for the architecture usability test. Note that, compared to Figure 45 of section 2.1, here more models have been added to the setup and the seat of the participant has been moved to the left side of the room instead of the right side. We have chosen this setup, since it is best for the Lego replicas to be on separate tables to simulate the use case of real models lying around at any position in the room.

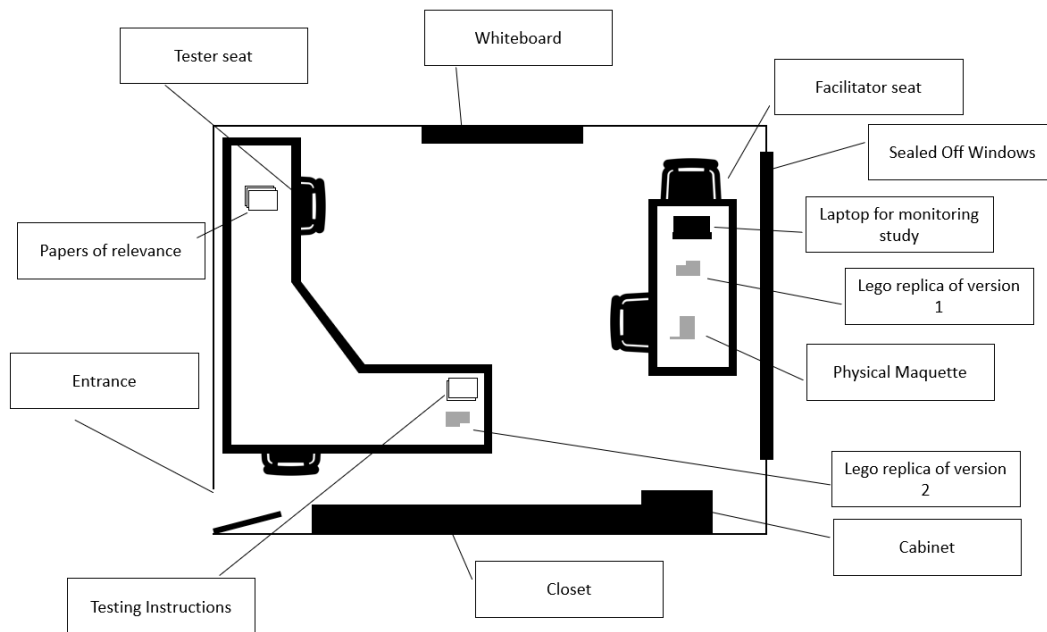


Figure 52: Architecture Study setup

In this study, two consent forms are also provided (Figure 52, “Papers of relevance”) upon entry in the room, where one copy is for the facilitator and one is for the participant. For this study we use the same consent form as in the computer science study (Appendix D), since the same elements apply here. After filling in the consent form, participants are also prompted with the questions of whether they know the concept of AR, whether they have any experience with AR and whether they

are sensitive to epilepsy inducing visuals. After gathering information involving these questions, the participants are introduced to the proceedings of the test and are also requested to apply the “Think Aloud” protocol for discussing the system and the changes in models.

After the introduction, participants are requested to stand up and are presented with two sheets of paper (see Figure 52, “Testing Instructions”) that include the instructions of the training phase (Appendix F) and the actual system (Appendix K). Again, both the facilitator and the participant contain a copy of these sheets so the facilitator can go over the steps verbally and the participant can double check in case they forget what they need to do. The facilitator also has a separate document containing more detailed descriptions about the study procedures, including the project structures and the answers to the questions that participants will need to solve (Appendix L).

All the other procedures including the discussion, note taking, logging and recording are the same as for the computer science study. We also provide the same training phase, since this still serves as a good introduction to the manipulation gestures and the aspects of the links and the decision cards. participants will carry out this training phase after receiving the introduction to the testing procedures.

After carrying out the training phase, participants are requested to hand the Hololens back to the facilitator to which he will load the next system. In total we carry out three different programs during the architecture study. The first program is the training phase, the second program provides the maquette versions traceability scenario mentioned in section 2.2.1 and the third program includes the project structure analysis scenario mentioned in section 2.2.2. The second and third programs both include their own questionnaire with slight adjustments compared to the other, to analyze the system for each scenario in full. After each of these two programs, we also prompt the participants with the question “Do you think a system like this can be useful for architects?”. This way, we can analyze how the participants interpreted the uses of each system. We describe these two programs or scenarios here separately, to discuss what we will carry out and analyze for each of them.

6.2.1 Maquette Versions Traceability

Like previously mentioned, for the architecture study we use the same projects that we mentioned in 2.2. For this scenario, this project is the major’s house model design mentioned in 2.2.1. Since the project only consisted of a physical model and its versions, only these maquette versions are present in this system.

Like seen in section 4.5, we allow the alignment of digital models with a real object. Within this scenario, the digital copies of the model are directly presented in front of the participant and are aligned with the real model. This alignment is done beforehand by the facilitator before giving the Hololens back to the participant.

We have also mentioned in section 4.5 that we allow participants to see the insides of the digital and real model when they approach closer to the model. To test the usability of this feature, one of the first tasks of the participant is to count the number of stairs inside the building. These stairs are not visible from outside the real model and can only be viewed by using the aforementioned feature. It will be interesting to receive feedback based on this feature on whether the architects find it useful for providing information on the models or not.

The next steps for the participant include navigating to the different versions of the model. Like mentioned in section 4.3.2, navigating to the different versions of a real model is done by using a swiping gesture in the direction you want the models to move in. For this scenario, it will be interesting to consider whether the participants understand the different versions in the visualization and are able to adjust to the swiping gesture without problems.

Another feature that is included in this scenario, is the ability to directly see the changes between the models in different colors. After participants are finished navigating to the different versions, they are requested to enable the changes visualization. When the participants are presented with the changes in colors, they are prompted on what they expect the meaning to be of the red and green colors. Like mentioned in section 4.3.2, a red color means that this component is removed in the next version and a green colored component means the component was added compared to the previous version. It will be interesting to see whether the participants are able to deduce the meaning of the colors and the changes based on these colors, which would imply that there could be merit in providing such a changes visualization in more detail.

After the participants are requested to visualize the changes in the different colors, they are required to analyze the different changes between the models and attempt to explain them to the facilitator. Decision cards between the models are also provided to help participants in understanding the changes and the reasons for the changes more clearly. It will be interesting to see whether the architect participants require the decision cards for finding the changes and whether there is merit in providing such a format for them in such a use case.

Like mentioned previously, we provide a separate questionnaire for this scenario (Appendix M). This questionnaire is based on the practices of the SUS questionnaire mentioned in section 5.2.2 but is made to be more specific for this use case. The practices inherited from the SUS questionnaire are the continuous switching between positive and negative oriented questions and the use of a Likert scale with values ranging between one and five. Some questions or statements of the SUS questionnaire have been removed in this modified version since they provide little to no extra information for this use case. These statements are “I thought the system was easy to use”, “I thought there was too much inconsistency in this system.” and “I needed to learn a lot of things before I could get going with this system”. The first statement was replaced with the more specific statement “I thought the gestures were easy to carry out.”, since the gestures are the most important form of interaction within this system. The second statement was removed because the system provided for this use case is small in scope and consequently there are not a lot of elements that can be compared with each other based on consistency. The third statement was removed since we are letting the participants use a relatively new type of technology with interaction methods that they are not yet accustomed to, making it so they had to learn how to use the system in the first place. Finally, one statement was also added to the questionnaire to help deduce whether the aligned digital copies in AR were clearly visible for the participants. This added statement is “I thought the models in the system were not clearly visible”, which is the final statement of our modified questionnaire.

For the calculation of the scores, we will still use the same practice as the evaluation of the SUS questionnaire but since this questionnaire is a modified version, it is best to not compare the total score calculation to the averages mentioned by the SUS questionnaire. Instead, there will most likely be more value found in focusing on the discussion of the question results separately. After filling in the Likert scale for the questions, participants are prompted with the question “Do you have any remarks to add about any of the previous statements?”, to ensure that the right conclusions are derived at based on the scores given to the Likert scales.

6.2.2 Project Structure Analysis

In the use case mentioned in section 2.2.2, there are already a lot of different models that are of relevance to understanding the full project structure. Because of this reason, we can re-use these models within our study. We have already discussed them in detail within section 2.2.2, meaning we can now focus on discussing the elements that are of interest for testing the system and answering our research questions.

First of all, it will be interesting to observe whether the participants are able to understand the full project structure based on the visualizations provided for these different models. This way

we can highlight whether the links with labels provide enough information for the participants to deduct how the different elements are related. We could even question whether the links or labels are even required, since participants might not need them in the first place for establishing the artifact relations.

Another element of relevance to our study, is the consideration of the physical model as part of the system. Since participants will not be able to see the digital model, however they will be able to see the real model. Consequently, all the links will also point towards these real models. For this reason, it will be interesting to analyze how the participants behave towards the real models being part of the system and other objects being linked to them. Example behaviors of this could be that the participants fail to recognize that the digital models are referring to the physical model or in contrast, are able to fully understand the implications made by the visualization on the real models. Based on these findings, we could deduce whether there is value in showing the digital objects, when referring to a real model in AR visualizations.

Like the previous scenario, participants are provided with specific steps used to analyze these different elements. In general, the steps of the study go as follows (see Appendix M): After receiving the Hololens from the facilitator, participants are requested to look at the visualization of the overview of Genk. They are asked to click on the model and press the “Show links” button to visualize the different links for the overview visualization. Only one link will be provided to the participant from this model, which goes towards the textured model of the conceptual design. Afterwards, the participants are asked to explain the link existing between the overview of Genk and the textured model using the decision cards. Furthermore, participants need to explain what the arrows of the link imply.

Asking these questions to the participants and having them discuss these relations, allows us to analyze how participants interpret the links and between the different models. However, using this method, there is a limitation to testing the participants interpretation of the links, since they will be enforced to read the decision card and to follow the link between the models. What this does not test, is whether the participants would intentionally make use of the links to trace the model relations. However, we are required to enforce this behavior, since we want to be able to analyze the interpretation of the links by the participants.

For this reason, we add an additional step in the study to analyze the use of links to trace the relations of the artifacts. In this step, participants are first required to follow the link from the textured model to the conceptual design filled with different colors. At this point they have already visualized the links for models twice, meaning they are most likely able to deduct at this point that they can visualize the important links for models using the “Show links” button. To test both this assumption and the assumption that the participant will intentionally make use of the links to trace the model relations, participants are requested to find the meaning of the colors of the conceptual designs. A link has been provided between the conceptual design and the side of the poster containing the color meanings with the label “Color codes are documented here”. This makes it so when participants look at the link of the model, they should be able to find the color codes rather easily. However, the question remains whether the participants will make use of this link to find the meanings.

To test the use case of the physical model being part of the visualization, after the color codes assignment, participants are required to first look at the versions of the poster and are then required to look at the versions of the conceptual design. Two of conceptual design versions are real models and contain links between both the final conceptual design model and the models in the versions view. We specifically request the participants to discuss and look at the different versions of the conceptual design (see Figure 53a), before we ask them to explore the versions of one of the real models (Figure 53b). This way we can first analyze whether the participants are able to successfully trace the digital links toward the real models and how they consider the models, before analyzing how the versions of real models are interpreted by the participants.

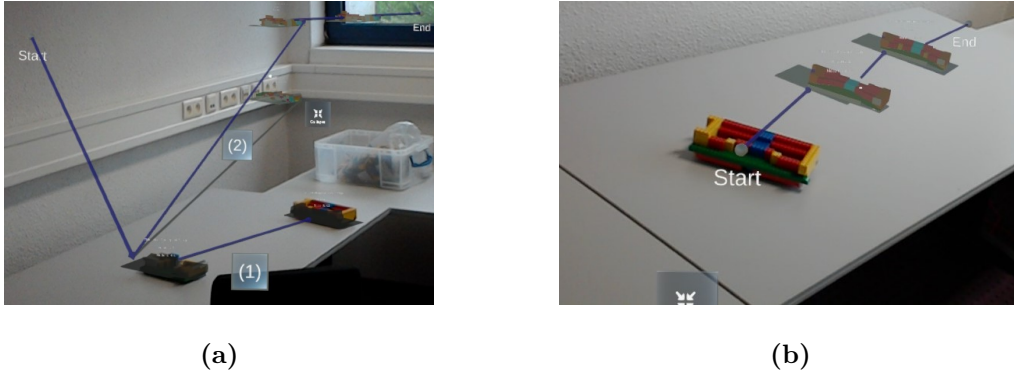


Figure 53: The conceptual design versions with **(a)** the full versions visualization and **(b)** the versions of the real model

For the versions of the real model, we provide the same visualization as the one for the digital models. We do this because it would be hard to provide a visualization like the one seen in section 6.2.1, when the digital model is not visible on top of the real model. We ask participants to explain the changes for these model versions and which of these versions is the one that is visualized by the physical model, to test whether participants can understand the changes made to the physical model and can make the comparison between a digital and a real model.

Finally, the participants have to explain the order in which the conceptual design versions appeared and what important decisions were made during the creation of these design versions (see Figure 53a). This is done to test whether the participants are able to understand the order in which the project versions appeared and are able to trace the changes, when a more complex example is provided to them.

For this scenario, a modified questionnaire based on SUS was also provided (see Appendix N). In this questionnaire, the same modifications are provided as the ones mentioned in section 6.2.1, except for the final added statement about the models being clearly visible. This statement is replaced within this questionnaire with the statement “I did not understand the structure of the project completely.”, since we want to test most how confident the participants felt in their understanding of the full project.

Finally, participants are provided with both the additional questions “Do you have any remarks to add about any of the previous statements?” and “Which system did you prefer and why?”. The first question was also present in the previous scenario of section 6.2.1 and is added here for similar reasons. The second question was added to make an evaluation on what type of visualization is preferred by the participants. This allows us to deduct what type of visualization participants have experienced to be more useful for providing traceability of artifacts with AR.

6.3 Observations and Analysis

In total we were only able to test two representatives of the architectural field. Due to the small sample size, it becomes hard to make solid statements about the system. However, we still discuss the findings of this study to indicate what we are expecting to result from a study with a larger sample size. We categorize our findings as follows: artifact traceability user methodology, real artifact interpretation and questionnaire evaluation. We do not discuss the scenarios of this study separately, since a lot of the findings within this study were derived from both scenarios.

6.3.1 Artifact Traceability User Methodology

In this category we discuss how participants experienced the artifact traceability elements within our system. These traceability elements include for the architectural field the use of annotations, decision cards, versions, changes visualization and artifact links. We will discuss for each of these elements what we have observed and concluded, to which we can compare these findings later in section 7 to the ones of the computer science study.

First of all, we have observed that both participants understood the meaning of all the annotations and decisions. They also used the annotations to understand the different components of the models for both scenarios. However, one observation we made was that both participants did not intentionally start using the decision cards. They did glance several times over the button that gives the decision cards overview, but they only opened the decision cards when this was requested by the facilitator. Both participants noted that within their field, architects most often do not read a lot of text, but instead tend to make visual comparisons between models and designs themselves. For this reason, they did not feel the need to visualize the full decision cards. Instead they relied on the annotations of the artifacts to describe the different model components and the titles of the decision cards that state specific changes. Both participants also mentioned the decision cards to be hard to understand when they are not able to make a visual comparison between the models. This was prevalent in the conceptual design versions use case seen in Figure 54, since the versions could not be moved closer together and the models were sometimes too far away or high up to be able to see the changes. From these findings we can deduct that in the field of architecture, decision cards need to play closer attention to the visual representation of the changes, instead of the textual like we did right now. To further highlight the importance of the visual aspects of the decisions. both participants also mentioned the arrows visualization to be useful for pinpointing what locations the decisions were referring to.

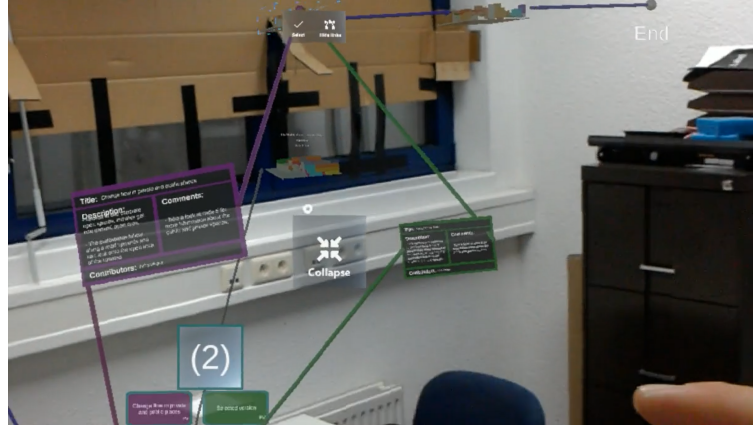


Figure 54: User looking at the decision cards but not being able to see the changes on the model above

Since participants understood the meaning of all the annotations and decisions and naturally used the annotations for deriving the meanings of different model components. we accept hypothesis **H1** stating “*The annotations provided the required amount of information to understanding the model components.*”. However, since only the arrows visualization and the titles preview were mentioned to be useful, and not the main constitution of the decision cards, we reject hypothesis **H7** stating “*The decision cards provided support for tracing the decisions and changes in the artifacts.*”.

Aside from the arrow links visualization between the decision cards and the artifacts, when prompted with the question “Do you think a system like this can be useful for architects?”, both participants mentioned the links between the artifacts to be a good and useful way to acquire information about other artifacts. They specifically mentioned these links to be of even more use for acquiring information about real artifacts in the environment and mentioned it to be useful for both team and stakeholder communication. They also mentioned the system to be useful in the sector of tourism, where visitors can explore the history and details of different artifacts.

In the use case where the participants had to find the meaning of the colors of the conceptual designs, one participant instantly chose to visualize the links for the conceptual design artifact. This participant mentioned that they did this since they expected there to be a link that would give them more information on where to find the meaning of the colors. The other participant did not take this approach since they mentioned that they saw a glimpse of the color codes when looking at the different models in the scenario and instantly went to this poster to discuss these colors. However, this participant also specifically mentioned the usefulness of the linking visualizations, both for the versions and the different artifacts. While we only observed one participant making explicit use of the links and labels between the models for tracing the artifact relations, we can deduce the linking visualization to be positively interpreted by participants. Based on these findings, we accept hypothesis **H5** stating “*The links provided clarifications on the relations between the digital models and the real models.*”.

When asked for the meaning of the colors in the versions visualization of the maquette, both participants were able to find the correct meaning of the colors (red for removed components, green for added components). Both participants were also able to find the changes easily and noted they found this to be an efficient way for visualizing the changes of the model versions, meaning we can accept hypothesis **H4** stating “*The colors of the changes visualization provided the information required for the participants to understand the changes between the maquette versions.*”.

Another observation we made, is that both participants mentioned that they would have liked the ability to move and scale the objects within a versions visualization. This was due to the fact that both participants wanted to make the visual comparisons between the different models and not being able to move the versions was impeding their ability to do so. This was again most prevalent when looking at the changes of the conceptual designs, since the participants wanted to move the models closer together to find the changes between these different models. One of the participants even dragged the infographics poster closer to the versions visualization of the conceptual designs, to be able to make the comparison between the different models (see Figure 55). The other participant also mentioned when discussing the changes of the conceptual designs, that they can look at the poster again to find the meaning of the colors when they feel like they need to. These observations imply that within the field of architecture, providing the ability in AR to move the digital artifacts closer together can improve the participant experience and efficiency for tracing the changes within artifacts.

One final observation for this category, is that when attempting to trace the versions of both the digital models and the real maquette, both participants experienced little trouble in understanding the project versions. They both mentioned the ability to see the versions of a model to be very useful for tracing the changes in a model. However, within the conceptual design versions visualization, one participant experienced more difficulty in understanding the sequence of the models. This was due to two of the models being spread around in the room as real models, making the versions visualization spread into different directions instead of being visualized at one static location in a specific order. This implies that, when a versions visualization is required in AR, it would be best to visualize it using a predefined format like we have provided for all the other models. Since participants understood the different project versions in the system, we will accept hypothesis **H3** which states “*The versions visualization provided the information required for the participants to understand the structure and history of the maquette versions.*”. However, more attention will still

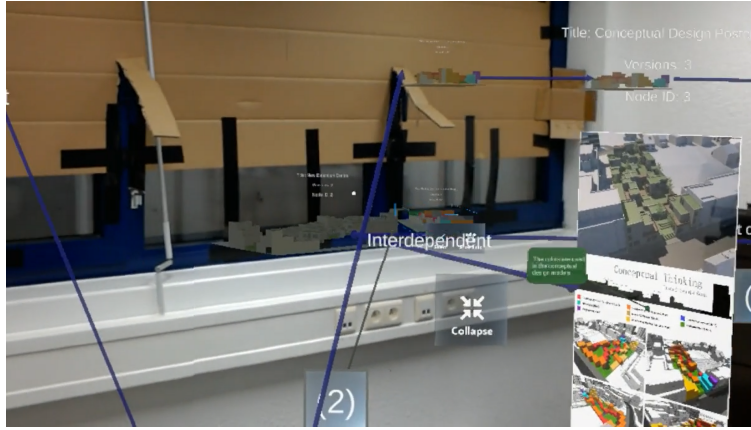


Figure 55: User placing the poster closer to the conceptual designs to help in tracing the changes

need to be spent for providing the versions visualization for spread out real artifacts.

6.3.2 Real Artifact Interpretation

Like mentioned in section 6.2, participants are presented with two types of real models: a real maquette with a digital copy visualized on top of it and Lego recreations of conceptual designs with no visible digital copy. In this category we discuss how participants interpreted these real models and what observations we made about the participants their behaviors involving these models.

For the first instruction of the real maquette use case, both participants were able to count the stairs within the model successfully. Both even mentioned afterwards that the ability to look within a real model using the system brings great benefits for communication with stakeholders about the insides of building structures. When looking at the real maquette, one participant even made the comparison between the digital model and the real model to understand what some parts of the digital model represented. However, when both participants were presented with real models without digitally aligned copy, they failed to notice its existence within the system. They even mentioned that artifact links that point towards these objects were “stopped abruptly in mid-air”. Consequently, both participants were surprised when asked to look at the versions of the biggest real model, since they had no idea it was part of the system as well.

Both participants also mentioned that they would have liked a digital model to be presented with the real model, where participants can use the digital model to interact with and use the real model to see elements in more detail. This is a useful feature to consider, regardless of whether participants are able to recognize the real models without digital copies to be part of the system. Overall, it seems that for providing artifact traceability for real models, the safest option would be to provide a visualization that shows a digital model for the object that is being explored. For this reason, we can safely accept hypothesis **H2** stating “*The digital models overlayed on top of the real models provided additional information about the model components.*”.

As seen by the hypotheses that have been applied for the real models (**H1**, **H2**, **H3**, **H4**, **H5**), a lot of additional information was successfully provided for the different artifacts by use of the linking visualizations and annotations in AR. For these reasons, we can also safely accept hypothesis **H6** stating “*Augmented Reality provided support for tracing the changes and relevant information of real objects.*”. We can do this due to AR being the technology that allows us to visualize these techniques on top of the real models.

6.3.3 AR Limitations

While there were some clear uses for AR within the sector of architecture, there are also some limitations that were present in the study. For example, both participants mentioned it to be bothersome to read and look at objects close-by, since parts of the objects would disappear due to the clipping plane of the Hololens. Both participants also mentioned that it was hard to notice things outside of the field of view like the collapse button of the versions or the decision cards and links appearing when requested. These findings indicate that there is still a need for increasing the effectiveness of the Hololens to fully allow users to explore the holograms like they would for real objects.

When it comes to using the gestures, both participants experienced trouble getting adjusted to both the manipulation gestures of the models and the swiping gestures for the maquette versions. However, once participants got adjusted to the gestures, they experienced little trouble in carrying them out repeatedly. They did mention that, for the swiping gestures of the maquette versions, they were confused in what direction they had to carry out the gestures to be able to select the models that were on the back and front edges of the visualization.

During the study, both users had to move to different physical location to explore some of the versions within the visualization. During this movement, they accidentally bumped into different objects within the environment including the chairs and tables. One user specifically mentioned this to be troublesome since they were too busy trying to interact with the digital models to look out for the real objects.

6.3.4 Questionnaire Evaluation

Due to the small sample size of participants, it is hard to make solid statements involving the questionnaire. For this reason, we will only discuss the general deductions from the questionnaire (full results in Appendix O). First, the second system was evaluated slightly better than the first system. This lies in line with the statement that the participants wrote for the final question “Which system did you prefer and why?”, where both participants mentioned the second system to be their preference due to it containing more detailed information about the project artifacts. Looking at the results from the previous section, the best case scenario for the system would be to apply the traceability linking of the artifacts from the second system and integrate it with the real model versions visualization from the first system.

7 Discussion

In this section we will discuss the findings of both studies, the limitations of our current research and the elements that are still left to explore in future work. We start with a discussion of the findings, where we provide main take-away messages on how to provide artifact traceability for the sectors of computer science and architecture.

7.1 Findings Interpretation

Within our research we saw that for the computer science study, the decision cards were very well received, but the links between the versions not as much. Compared to these findings, within the study for the architectural field the decision cards were interpreted as less useful and the links between the artifacts were considered as essential. However, annotations were received to be positive and useful within both studies. While the decision cards were initially created with the intent of providing design rationale for software GUI's, we expected the lightweight format to be just as useful for other uses and sectors as well. We already added some customizations to the use of decision cards ourselves, including the decision cards overview between models and the arrow links visualization (see section 4.4). These customizations were received to be positive and useful within the field of architecture, since they provide a quick overview of important matters and visualized the different components the decision cards were referring to.

Since it was mentioned that architects look for changes visually instead of textually, the original physical decision cards format (see section 1.1), would have been more suitable for this use case. This is due to the additional post-it notes in the comments section of the decision card being usable for drawing visuals for the decisions. However, the need is still present to keep these decisions in digital format so users can explore them using an artifact traceability system. For this reason, we suggest the inclusion of a new field in the digital format of the decision cards called the “transition field”. In this field, users are provided with two drawings or visualizations that represent the before and after states of the decision. This practice can also be served as an extension or guideline for the post-it notes present in the “comments, discussion and artifacts” field, to allow for easy integration with the current format.

The level of detail for the drawings can be entirely left up to the user, as long as the drawings represent what the state of the model was, or would have been, before the decision was made and what the state of the model is after the decision is made. Since the amount of detail can be left up to the users themselves, the decision cards format remains lightweight for the users. However, for the sector of architecture, it would be best to remove the description or enforce that only short statements should be present within the description to avoid occluding the card with too much information. In the case of software GUI designs, the transition field can also be used where the before and after states include the particular components that are changed within the decision. This makes the new field reusable for the initial target demographic as well.

When the decision cards are presented in a digital format, the “transition field” can instead include screenshots of the different parts of the model that are being discussed. To further enhance this, it could also be interesting to include a transition effect between the screenshots for the before and after states, to visualize how the components changed. However, since we were not able to test this new format of the decision cards yet, we can only make speculations on the use of it within the sectors. For this reason, we invite any researchers to attempt to test this new approach within future work.

Based on our previous findings, we establish guidelines for providing artifact traceability within the fields of computer science and architecture, which serve as the main take-away messages of this research. Since we have not tested the use of our new decision cards format yet, we will not include it in the current guidelines of the system. Some of the elements within the guidelines for the sectors

of architecture and computer science overlap with each other, meaning we will first establish general artifact traceability guidelines applicable for both fields.

7.1.1 General Artifact Traceability Guidelines

We establish the first general guideline for providing artifact traceability to be: **make use of annotations to highlight relevant points by connecting them to components with links or arrows**. We formed this guideline, since we have observed several times within the study that annotations helped in understanding the different model components, both for real and digital models. Participants have also mentioned the linking visualization of the annotations and decision cards to be very useful for highlighting what components are referred to by both formats. While the decision cards we integrated were positively evaluated within the computer science study, for the architecture study this was less apparent, so we specify the general guideline to only include annotations.

Our second general guideline consists of the following: **provide a visualization that gives an overview of the artifact versions based on their sequence relations**. This guideline was created from the findings that all participants understood the meaning of the versions visualization and were able to understand each separate version and the links between these different versions. Since the core component of artifact traceability consists of tracing the changes within an artifact, we can consider the versions visualization to be the main component that allows us to provide this traceability. For finding the changes within the different phases of the application views in the computer science study, participants made it their custom to visit each version chronologically. While during the architecture study, showing the versions in a scattered visualization proved to be problematic for the participants to find the sequences of the versions. Since no solid conclusions could be made about the links of the branching visualization, we have established the second guideline to include only the versions visualization and their sequence orders. This ensures users will be able to understand how the versions are related, while meanwhile supporting their methodology of finding changes chronologically.

For our third guideline, we give the following definition: **provide the option to apply semantic zoom for the different artifact versions within the versions visualization**. Since in both studies participants were exploring the different versions separately to be able to find the properties of a specific version, we established the semantic zoom to be useful for users within both sectors. Like mentioned by the definition of semantic zoom and branching in section 3.1.3, when it comes to exploring different artifact versions, users should be able to “check out” each version separately. This means our third guideline remains consistent with the methodology in section 3.1.3.

Finally, our fourth general guideline consists of: **provide the traceability practices within the same level of dimension as the main artifacts that are being explored**. We have selected this guideline since within the study of computer science, participants found the AR component of the system to bring overhead for the interaction. This was due to the different models of this sector consisting of 2D designs for software GUI's. Participants mentioned that for this use case, they would have liked to use these artifact traceability practices on a traditional 2D screen instead, since there is not much benefit to using AR for exploring these 2D models. Meanwhile for the study of architecture, some artifacts consist of 3D models where some models are already present in real-life as “maquettes”. Within this sector, architects build models in real-life to provide more information about the spatial properties of the components, which cannot be properly seen or interpreted using a 2D screen. For these reasons, within the study of architecture, there was a lot of benefit to be seen in using the AR system to visualize objects related to the real models. For example, participants mentioned the benefit of the AR system for being able to visualize the spatial properties of objects, in the same way how maquettes are used to provide this information. They also mentioned that the artifact traceability practices, combined with the real models, provided

them more detailed information on the object versions and components than what they would have known initially with only the real model. In other words, we have established our fourth guideline based on the finding that 2D designs were preferred to be visualized on a 2D screen and AR was proven to be useful for exploring the concepts of 3D designs and real models.

The fourth guideline also implies that one should only make use of practices like AR and VR, which visualize objects in a 3D environment, when the main models being explored are also consisting of 3D models. Like the example within the architecture study, the AR system can still include 2D models like infographic posters and manuals, since these are of relevance to the other artifacts. However, when only provided with 2D screens or only a select few 3D models of minor importance, it would be best to stick to a traditional 2D system for exploring these models. When AR and VR become the new traditional systems, 3D designs will most likely be made for the interactions with these systems, meaning that according to this final guideline, these 3D designs should also be explored in AR or VR.

Compared to the guidelines mentioned in section 1.1, these guidelines are more specific since they enforce the use of specific techniques in artifact traceability, rather than only the use of certain concepts. Our first guideline is for example, a more specific version of the guideline *“Provide a visual connection between artifacts and communication related to the artifact.”* mentioned in section 1.1. However, when these techniques are applied in context of an artifact traceability system exploring design changes, user experience and decision making should be optimized. To provide further enhancement of this process, we will now discuss additional guidelines for each sector individually.

7.1.2 Artifact Traceability Guidelines for software GUI’s

To provide artifact traceability for software GUI designs, we establish one specific guideline that states: **support the use of decision cards which highlight relevant points using arrows.** This guideline is created due to the finding that, within the computer science study (see section 5), participants naturally started relying on the decision cards to find the changes between different versions. The absence of decision cards was also met with confusion and failure to find changes between the models. Participants in both studies even mentioned the arrows visualization to be useful for pinpointing the locations that the decision cards refer to. However, within the study of architecture, participants mentioned the decision cards to be less useful, since architects tend to make visual comparisons between models instead of reading textual changes. For this reason, we have already established a new field for the decision cards that should prove to be useful within architecture as well. However, since this new decision cards format has not been tested yet, we can only establish the decision cards to be essential for understanding the changes between the different versions of software GUI’s.

7.1.3 Artifact Traceability Guidelines for Real models in Architecture

The first specific guideline to providing artifact traceability for real models within the sector of architecture consists of the following: **provide a digital copy of the real model that represent the model within the system.** Within the study for the sector of architecture, participants had to explore real models as artifacts (see section 6). In general, there were two types of visualizations used for the real models: a visualized digital copy aligned with the real model and the real model directly being considered as part of the system. The first specific guideline for this sector is based on the findings that, during the study, participants were unable to identify that the real models were part of the system for the second visualization type, even when arrows were pointing towards these models and it had a title floating above it. Aside from this finding, participants specifically mentioned they preferred a visualization where a digital copy is provided for the real models. This was due to the copy being usable for exploring the model in the system using the interactions, while at the same time making it possible for comparisons to be made with the physical counterpart. Hence why this first guideline was created.

We define our second guideline for this use case to be: **provide labeled arrow links between different objects to indicate their relationship**. During the study of the architecture sector, labeled arrow links were provided to indicate the relations between the different models. We have established the second guideline, since participants of the study mentioned these links to be very useful for explaining how the different objects are related and mentioned them to be specifically of use when exploring real models. Based on this information, we speculate that the links successfully provide clarifications for the relations between the digital models and the real models. The reason why this guideline was not included in the general guidelines, was because no analysis on the use of the relation links could be made in the computer science study. The computer science study only consisted of versions of an application view, where these versions equated to the different phases in the software design lifecycle. This means no other objects and relation links between these objects were included within the study and no conclusions could be derived for the use of relation links.

The third and final specific guideline for this use case consists of: **provide a color visualization to indicate between the versions, what parts have been removed and what parts have been added**. We establish this guideline, since our findings showcase that users found the colors visualization technique to be useful for finding the changes. Within our system we used the color red to indicate the parts that have been removed and green to indicate the parts that have been added. When interacting with the system, the users found the meaning of these colors instantly, indicating that there might even be value in specifically using the colors green and red for this visualization. However, we have not tested the visualization for other colors, meaning we cannot specify these colors to be required. The biggest limitation of the colors visualization technique, lies in the fact that it can only be provided for non-complex changes. We could have chosen to not include this guideline due to these limitations, however based on the scenarios of other projects, we believe this visualization can still prove to be useful for providing artifact traceability for some of the models. For this reason, we have still included it here.

7.2 Limitations

Within the studies of our research there were several limitations present. These limitations are either parts that we did not cover, work arounds that we had to provide or general problems that were not resolved in the system. We discuss these limitations here in several different parts.

7.2.1 Hardware Limitations

We have mentioned previously in section 4.2, that we use the first edition of the Microsoft Hololens to conduct the studies. However, there are some known limitations of this edition, which were also prevalent during our tests. One of these limitations was the clipping plane and field of view. Due to the clipping plane, when objects come too close to the participant, parts can disappear making it hard to view objects from close-by. Meanwhile due to the small field of view, objects are sometimes not visualized when users are looking into a different direction than the objects [6].

During our test, participants mentioned these issues several times, since it became difficult to read some text elements, look at some objects clearly and notice changes happening to objects that are outside of the field of view. However, these consequences were mostly mitigated due to the scaling option we provided for all the digital objects. New AR systems are also already being built that mitigate these problems within the hardware. An example of this is the second edition of the Hololens that will be released later in 2019 that has twice the field of view compared to the edition we used in our research ²⁵.

²⁵Microsoft Hololens 2 specifications on their website: <https://www.microsoft.com/en-us/hololens/buy>

7.2.2 Experimental Limitations

Within our experiments, there were several limitations present in how they were conducted. First off, within our current research the projects for the architecture and computer science studies were of smaller scale compared to projects used in actual business contexts. In the case of computer science, a lot more versions and application views would be present with probably more complex changes between the different designs. For the case of architecture, a lot more models, documents and measurements would be present as well with examples being construction documents, subsoil properties and required resources. In these complex cases, the visualizations become a lot more complex as well since links and versions would need to be provided for every model of relevance. Since we have only tested our system for providing artifact traceability visualizations for smaller scale projects, it remains unclear to what degree our findings remain applicable for such large scale projects. For example, we found that using the green and red colors for version changes visualization of the architecture study provided useful information to the participants about how the model has changed. However, these changes were very small and only consisted of the addition and removal of components. When model versions go through more complex changes, providing such a visualization becomes more difficult.

The same counts for the versions visualization itself. Real models may contain a lot of different versions throughout multiple branches. A question we could ask is whether our versions visualization will remain clear enough to understanding the exact changes that happened between the different models in a complex project. For now we can only speculate on this behavior until tests involving this topic have been conducted. For this reason, we invite other research to test the uses of artifact traceability practices for more complex and large scale projects.

Another limitation within our experiment, is that our study for the sector of architecture was severely limited by the sample size of participants. The reason for this small sample size was a lack of interest from within the sector during the recruitment phase of this study. For this reason, a lot of the speculations we made within this sector might not hold for a larger sample size. Sadly enough we can only assume the things that were mentioned to hold for all test cases until proven otherwise with a larger sample size.

7.2.3 Real Model Interpretation

During our study we have covered both the integration of real models using an aligned digital copy of the object and considering the real model to be directly present in the system. We have concluded within the study that participants were unable to derive that the real model was part of the system in the second scenario and preferred the use of the digital copy for providing information on the real model. However, one of the reasons that participants could not tell the real model to be part of the system, could be due to them having been presented with the digital copy scenario first. For this reason, they might have expected that a real model would only be part of the system when there is a digital object aligned with it. To properly test for this case, we should have used a Latin Square within the study to switch between the two scenarios, but we only discovered this need after the tests were already conducted.

This means the speculation about the real model not being able to be found by participants, should not be assumed to hold true for all cases. However, participants still mentioned to prefer a digital copy being present for the real model, which does not change our main finding about the real model integration.

7.3 Future Work

There are several different elements that we have covered within our research, from design rationale tracing to artifact relations to real model integration in AR. In section 7.2, we have already discussed

some of the limitations present in our current research for testing the use of these elements. Best case scenario, our research is repeated in the future without these limitations being present. However, aside from our research, these limitations should also be prevented within other future work as well. Aside from these limitations, there are some other components which we were unable to cover within our own research that should also be covered within future work. We discuss these components here as follows.

7.3.1 Collaboration Possibilities

In section 1.1, the second guideline mentioned the use of providing a shared workspace to help support in tracing the design rationale of artifacts made within teams. While we did have a server in our system containing the full project configurations that could support a collaboration environment (see section 4), we did not focus our research on the collaborative aspects of the system within team settings. Rather we focused on the stakeholder communication aspects of using such a system. However, during the computer science study, two participants mentioned they saw a lot of potential in the system for collaboration by allowing users to work together with such a system in a shared environment (see section 5.3). For this reason, there could be a lot of anticipated value in providing a system or research that explores the collaboration aspects of artifact traceability combined with AR.

To further emphasize on the concept of collaboration, in section 1.2 we mentioned one of the benefits of using AR includes the ability for remote users to guide local users in carrying out certain specific tasks. Within our own research, we have only focused on the communication with users that are currently present within the same environment. However, in real use cases, not all users will always be present in the same location and often the need for remote collaboration and communication will arise. This way, a lot of interesting scenarios can come about where artifact traceability needs to be provided for remote settings. While we have not explored these scenarios for ourselves, we will gladly invite any researchers interested in this topic to evaluate the collaboration aspects of local and remote use cases more in depth.

7.3.2 Creation of Traceability Means

During our study, we have analyzed the use of annotations, decision cards and links between artifacts and versions for providing artifact traceability for projects. However, during this analysis we did not cover the aspect of users creating these traceability means. We did not cover this, since our focus within this research lies on analyzing how effective these artifact traceability means can communicate design choices and artifact relations to stakeholders for different sectors. In the perfect scenario, we would have conducted two tests for the system. In the first test, participants would have to create the labeled links, decision cards and annotations for a given project, using an interactive system. Depending on the use cases that will be explored, this interactive system could either be in AR or on a more traditional system, since the project configurations are saved on a server regardless. The second test would be the same test that we have conducted in this research, but then participants would be presented with the decision cards, annotations and links created by other participants.

Currently the testing proceedings already consisted of an hour for every participant. If we had wanted to cover both these use cases, too much time would have to be spent during each test or less participants would be covered for each use case. This is due to each participant either having to cover both the creation of the traceability means and the analysis of other participants their creations or two separate tests would have to be conducted with other participants, where one test covers the creation and the other covers the analysis. Using this method, less conclusions would be derived for each individual use case, since less tests are conducted, or we would have to cover too many tests where it becomes hard to test the use of artifact traceability for each sector individually. For this reason, the most optimal method for deriving the use and creation of artifact traceability

means for both sectors of computer science and architecture, would be to carry out research for each sector individually where these use cases are explored more in depth. While we have not covered this within our own research, we invite any researchers to explore these use cases for both sectors further in future work.

7.3.3 Full Scale System Usability

We have already established the concept of providing collaboration within the system in section 7.3.1 and in section 7.3.2 we talked letting users create the traceability means that are currently used to derive the design changes. However, these two concepts can also be combined to build a new system where users can collaborate in an AR environment to create and explore annotations, decision cards and artifact links. Such a system would include all the components necessary to be fully applicable within project teams to both log and trace their design decisions for artifacts within AR.

To test this system, an approach similar to *Helaba* should be employed where project teams are requested to use the system actively for several months to log the design decisions and information for their projects. Using such an approach, more solid conclusions can be made on how an AR system can be used for providing traceability within active project teams for prolonged periods of time. However, based on the findings of this study, it might be best if such a system were applied for sectors like architecture that include real 3D models that need to be explored. We invite any future work to attempt and take this approach so we the full use of AR for providing artifact traceability can be analyzed.

7.3.4 Consideration of Physical Environments

Due to AR being visualized in physical space and users being able to walk around during the interaction, naturally the use of physical space also played a role in the interaction with the system. To exemplify this, participants had to move to specific locations in the environment to explore some of the models which made them accidentally bump into some of the physical objects. These cases all happened when versions of a hologram were projected beyond the walls of the room, since these versions could not be moved directly towards the user. Combined with the findings that participants did not initially notice the physical objects to be part of the system even when there were links pointing towards the models, we speculated that when interacting with the AR system, users tend to forget about objects in the physical environment. Aside from the architecture guideline mentioning that digital copies for the considered real artifacts need to be provided (see section 7.1.3), it is also important in future work to thoughtfully consider the placement of the holograms in real-life. This is especially relevant when the environment contains dangerous or important objects that should not be touched accidentally.

Another way to prevent accidents from happening, is to provide feedback to the users about the environment around them. This feedback can include warnings indicating there are important objects close-by or force pressure against the forehead of the user from within the hardware when users start approaching these objects in the environment. As one would expect, these types of feedback are of even more importance within VR systems where users cannot see anything about the environment, compared to AR where they can at least look around.

8 Conclusion

In this research, we introduce a new AR program that allows users to discover artifact traceability information about real objects in the environment. We build this program to analyze what practices need to be applied to the real objects to provide the required traceability needs. To also enable other developers to provide these needs, we establish several design guidelines based on the practices we used that can be followed during the development of such a system.

By providing the artifact traceability information for the real objects, users can now discover how these objects were made, what important decisions were taken for this object and what components are present and of relevance for the object. Furthermore, users are directly presented with this information without them having to search for the right physical papers or digital documents, across a working environment. This also ensures all the required information is communicated to the stakeholders for them to understand a project presented before them, meaning they can make more conscious decisions when it comes to the project.

We evaluate the use of artifact traceability using Augmented Reality for the sectors of computer science and architecture. For the sector of computer science, this study focuses on providing a system for tracing the design rationale of software projects. During the tests for this sector, participants look at a smartphone containing different views of an Android application where the AR system will show a digital copy of the view currently shown on the smartphone. Using this approach, users can discover for an application how each of the views were created. The applied traceability practices that we provided for these views include: visualizing the design phases of the application views using a technique called branching, showcasing annotations to provide clarifications for the application components and logging the design decisions by use of a format called decision cards.

During the evaluation, all participants were successfully able to find all the changes and decisions between the different design phases. This makes us conclude that the artifact traceability practices we used are indeed effective methods for providing the information required for users to find and understand the changes made during the application design phases. The participants even specifically mentioned the system to be useful for within team and stakeholder communication. During the evaluation, participants were also naturally drawn towards and became reliant on using the decision cards for finding the changes between the versions. This leads us to believe that decision cards are an essential component for providing traceability for the different design phases of software GUI's. However, evaluation has also shown that there was little to no merit found in the use of Augmented Reality for this sector. This is because participants mentioned the use of AR to only bring overhead for them when interacting with the models, since these models traditionally only consist of 2D screens. For this reason, we establish that for the exploration of application designs, participants prefer using a traditional 2D system like a computer or smartphone, instead of using AR technology.

Within the study of the architecture sector, a different approach is used compared to the computer science study. In the sector of architecture, stakeholder communication often includes the creation of physical replicas of 3D models which are called “maquettes”. For this reason, in our study, traceability information for the field of architecture is provided for two different use cases: maquette versions traceability and project structure understanding. To provide maquette versions traceability, a digital interactive hologram is aligned with a physical maquette in the AR environment. This digital hologram is used to provide additional traceability information about the real maquette, where similar practices are used compared to the ones within the computer science study. To provide project structure understanding, the different models of an architectural project are visualized in AR and are linked together to clarify their relation towards each other. These different models each also include their own decision cards, annotations and versions which can be explored.

After conducting tests with a small sample size, the participants mentioned they consider the

Augmented Reality practices to be useful for providing information on real models within the sector of architecture. While they still need time to get adjusted to the interaction with AR, they mention a lot of use cases that can benefit from such a visualization including stakeholder and team communication. However, since architects tend to compare models using visual comparisons, not much value was found towards the use of decision cards, since these cards include large amounts of textual information. To further emphasize on the tendency of architects to make visual comparisons, participants did find the arrows of the decision cards to be useful for highlighting what components more information is provided on. Because these findings highlight a problem with the original decision cards format for the sector of architecture, a new and adjusted format was discussed that can be explored within future research.

In the end, both results of our studies showcase that artifact traceability practices are considered to be useful for providing stakeholder communication within both fields computer science and architecture. However, only within the field of architecture was the use of AR considered to be positive and useful as well. Based on these findings, we established both general and specific guidelines on how to provide artifact traceability for both sectors. While there are many limitations present within our research like the hardware used, the scope of the considered projects and the limited sample size for the architecture study. We believe our findings and guidelines will still prove to be useful within future work for ensuring the right traceability practices are applied for the considered fields.

References

- [1] Application software companies; researchers submit patent application, "History as a branching visualization", for approval, 2015.
- [2] Ioan Adascalitei and Marius Baltoi. The influence of augmented reality in construction and integration into smart city. 22:55–67, 06 2018.
- [3] Matt Adcock and Chris Gunn. Using projected light for mobile remote guidance. *Computer Supported Cooperative Work (CSCW)*, 24(6):591–611, Dec 2015.
- [4] Charvi Agarwal and Narina Thakur. The evolution and future scope of augmented reality. *International Journal of Computer Science Issues (IJCSI)*, 11(6):59–66, 11 2014. Copyright - Copyright International Journal of Computer Science Issues (IJCSI) Nov 2014; Document feature - Diagrams; Graphs; ; Last updated - 2014-12-27.
- [5] Mujtaba Alshakhouri, Jim Buchan, and Stephen G. MacDonell. Synchronised visualisation of software process and product artefacts: Concept, design and prototype implementation. *Information and Software Technology*, 98:131 – 145, 2018.
- [6] Duncan Bell. MICROSOFT HOLOLENS, 2016.
- [7] Brian Berenbach and Ren-Yi Lo. An introduction to project and product traceability. *INSIGHT*, 14(4):38–42, June 2015.
- [8] Jyoti Bhagat. Google unveils 'ARCore' to bring ar to millions of android devices, 2017.
- [9] Ó. Blanco-Novoa, T. M. Fernández-Caramés, P. Fraga-Lamas, and M. A. Vilar-Montesinos. A practical evaluation of commercial industrial augmented reality systems in an industry 4.0 shipyard. *IEEE Access*, 6:8201–8218, 2018.
- [10] Marie Boran. Imagine your new home with Ikea's augmented reality app, 2018.
- [11] Rob Bracewell, Ken Wallace, Michael Moss, and David Knott. Capturing design rationale. *Computer-Aided Design*, 41(3):173 – 186, 2009. Computer Support for Conceptual Design.
- [12] Jorge Camba, Manuel Contero, Michael Johnson, and Pedro Company. Extended 3D annotations as a new mechanism to explicitly communicate geometric design intent and increase CAD model reusability. *Computer-Aided Design*, 57:61 – 73, 2014.
- [13] Emma Collins and Liz Ditzel. Holograms enhance student learning. *Kai Tiaki : Nursing New Zealand*, 24(10):26–26, 2018.
- [14] Peter Demian and Renate Fruchter. Effective visualisation of design versions: visual storytelling for design reuse. *Research in Engineering Design*, 19(4):193–204, Jan 2009.
- [15] Juri Di Rocco, Davide Di Ruscio, Ludovico Iovino, and Alfonso Pierantonio. Traceability visualization in metamodel change impact detection. In *Proceedings of the Second Workshop on Graphical Modeling Language Development, GMLD '13*, pages 51–62, New York, NY, USA, 2013. ACM.
- [16] Nader El-Bizri. Imagination and architectural representations. *From models to drawings: Imagination and representation in architecture*, pages 34–42, 2007.
- [17] Christian Gerth, Jochen M. Küster, Markus Luckey, and Gregor Engels. Detection and resolution of conflicting change operations in version management of process models. *Software & Systems Modeling*, 12(3):517–535, Jul 2013.

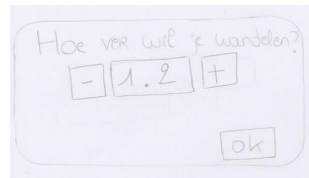
- [18] Jefferson Graham. Get ready for thousands of augmented reality Apple apps, 2017.
- [19] Ravi Kumar Gupta, Farouk Belkadi, Christian Buerge, Frank Bitte, Catherine Da Cunha, Jens Buerge, Gisela Lanza, and Alain Bernard. Gathering, evaluating and managing customer feedback during aircraft production. *Computers & Industrial Engineering*, 115:559 – 572, 2018.
- [20] Marisela Gutierrez Lopez, Gustavo Rovelo Ruiz, Mieke Haesen, Kris Luyten, and Karin Coninx. Capturing design decision rationale with decision cards. Springer International Publishing, 2017.
- [21] Marisela Gutierrez Lopez, Gustavo Rovelo Ruiz, Kris Luyten, Mieke Haesen, and Karin Coninx. Re-thinking traceability: A prototype to record and revisit the evolution of design artefacts. In *Proceedings of the 2018 ACM Conference on Supporting Groupwork*, GROUP '18, pages 196–208, New York, NY, USA, 2018. ACM.
- [22] Mieke Haesen, Karin CONINX, Jan Van den Bergh, and Kris Luyten. Muicser: A process framework for multi-disciplinary user-centred software engineering processes. SPRINGER-VERLAG BERLIN, 2008.
- [23] SRI International. Augmented reality solutions for construction inspection, 2017.
- [24] Riitta Jääskeläinen. Think-aloud protocol. *Handbook of translation studies*, 1:371–374, 2010.
- [25] Kevin Karsch, Mani Golparvar-Fard, and David Forsyth. Constructaide: Analyzing and visualizing construction sites through photographs and building models. *ACM Trans. Graph.*, 33(6):176:1–176:11, nov 2014.
- [26] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. Virtual object manipulation on a table-top ar environment. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*, pages 111–119, Oct 2000.
- [27] Yoichi Kobayashi and Jiro Abe. Realtime dynamic hologram of a 3d object with fast photochromic molecules. *Advanced Optical Materials*, 4(9):1354–1357, 2016.
- [28] Simone Kriglstein and Stefanie Rinderle-Ma. A visualization concept for high-level comparison of process model versions. In Marcello La Rosa and Pnina Soffer, editors, *Business Process Management Workshops*, pages 465–476, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [29] H.Y.K. Lau, K.L. Mak, and M.T.H. Lu. A virtual design platform for interactive product design and visualization. *Journal of Materials Processing Technology*, 139(1):402 – 407, 2003. IMCC2000 Vol. 2 S.I.
- [30] Wen-Tin Lee, Whan-Yo Deng, Jonathan Lee, and Shin-Jie Lee. Change impact analysis with a goal-driven traceability-based approach. *International Journal of Intelligent Systems*, 25(8):878–908, 2010.
- [31] Ingrid Lunden. A new ARKit app from houzz brings 500,000 objects to moveable life, 2017.
- [32] P. Rovegard, L. Angelis, and C. Wohlin. An empirical study on views of importance of change impact analysis issues. *IEEE Transactions on Software Engineering*, 34(4):516–530, 2008.
- [33] Joachim Scholz and Katherine Duffy. We are at home: How augmented reality reshapes mobile marketing and consumer-brand relationships. *Journal of Retailing and Consumer Services*, 44:11 – 23, 2018.
- [34] Joon Gi Shin, Gary Ng, and Daniel Saakes. Couples designing their living room together: a study with collaborative handheld augmented reality. In *Proceedings of the 9th Augmented Human International Conference*, page 3. acm, 2018.

- [35] R. Sudarsan, S.J. Fenves, R.D. Sriram, and F. Wang. A product information modeling framework for product lifecycle management. *Computer-Aided Design*, 37(13):1399 – 1411, 2005.
- [36] Márton Szemenyei and Ferenc Vajda. 3D object detection and scene optimization for tangible augmented reality. *Periodica Polytechnica. Electrical Engineering and Computer Science*, 62(2):25–37, 2018.
- [37] Matthew Tait and Mark Billingham. The effect of view independence in a collaborative ar system. *Computer Supported Cooperative Work (CSCW)*, 24(6):563–589, Dec 2015.
- [38] Thomas Wolfenstetter, Mohammad R. Basirati, Markus Böhm, and Helmut Krcmar. Introducing trails: A tool supporting traceability, integration and visualisation of engineering knowledge for product service systems development. *Journal of Systems and Software*, 144:342 – 355, 2018.
- [39] Zhiqiang Yan, Remco Dijkman, and Paul Grefen. Business process model repositories framework and survey. *Information and Software Technology*, 54(4):380 – 395, 2012.
- [40] Xiaohui Zhao and Chengfei Liu. Version management for business process schema evolution. *Information Systems*, 38(8):1046 – 1069, 2013.

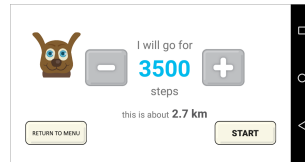
Appendices

A Walk With Me Screenshots

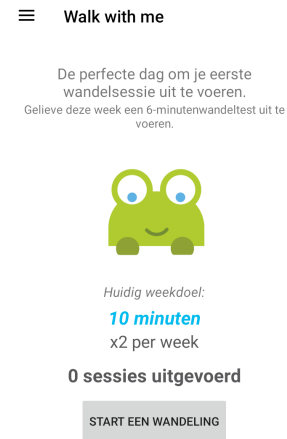
Abstract



(a)



(b)

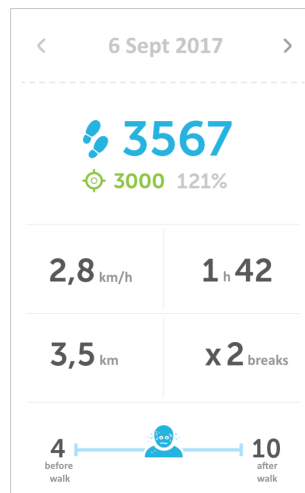


(c)

Figure 56: Scene 1, the option to start a walk with (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version.



(a)



(b)



(c)

Figure 57: Scene 3, the statistics visualized for walks with (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version.

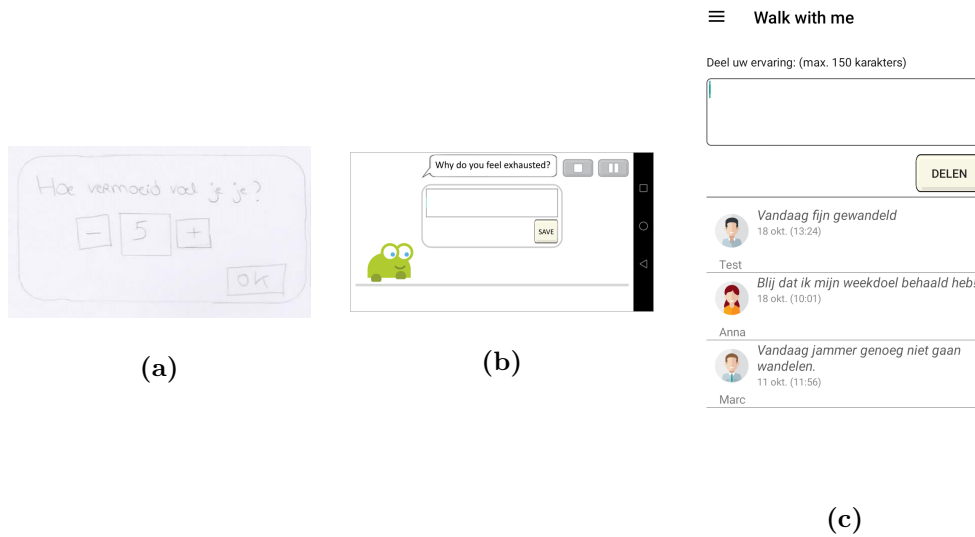


Figure 58: Scene 4, feedback on a walk and sharing the experience with a community (a) the low-fidelity prototype, (b) the high-fidelity prototype and (c) the final version.

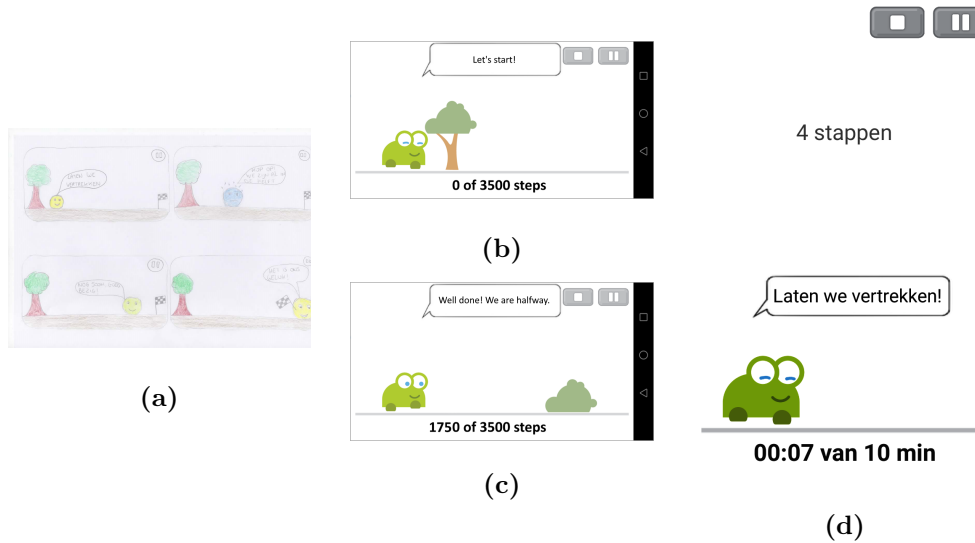
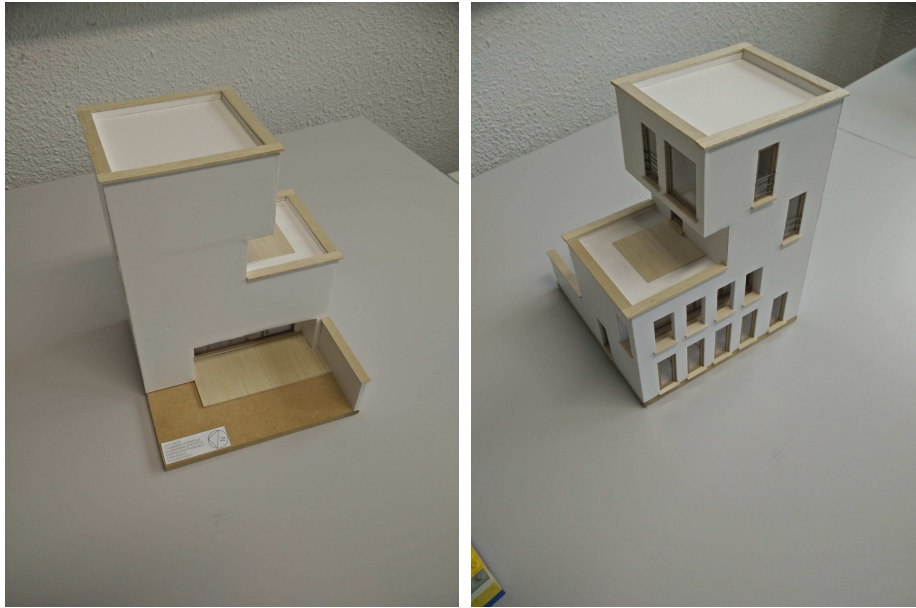


Figure 59: Scene 2, the progress shown during the walk with (a) the low-fidelity prototype, (b) the first high-fidelity prototype, (c) the second high-fidelity prototype and (d) the final version.

B Major's House Project

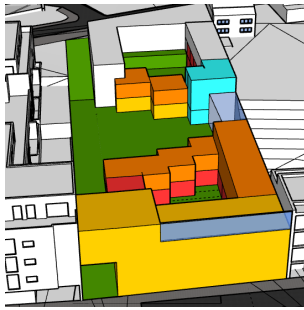


(a)

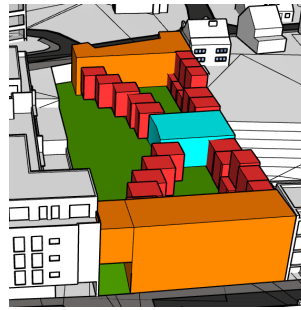
(b)

Figure 60: Physical maquette for the project of the major's house with (a) the front view and (b) the back view.

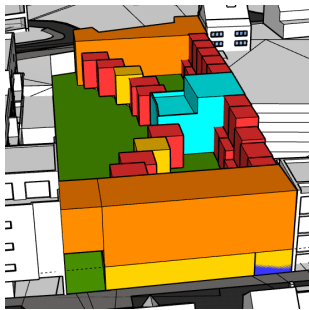
C Conceptual Design Detention Center Project



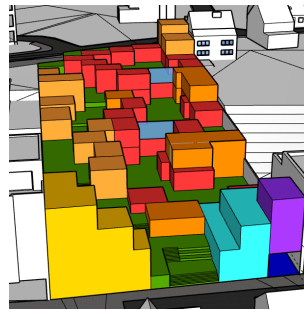
First version



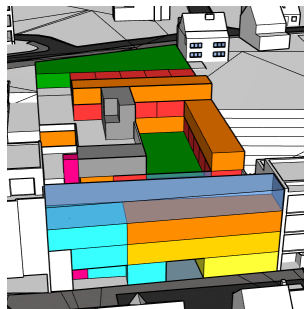
Second version



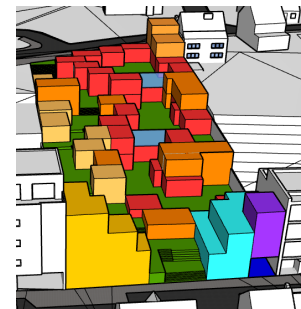
Third version



Fourth version



Fifth version



Sixth version

Figure 61: The conceptual designs for the detention center project

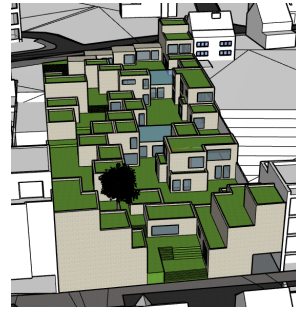
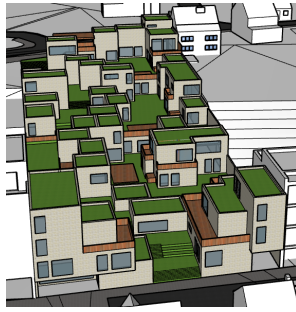


Figure 62: The textured designs for the detention center project

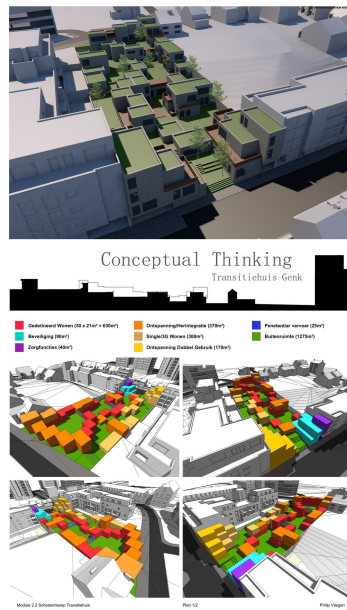


Figure 63: The infographic posters for the detention center project

D Consent Form

OVEREENKOMST VOOR HET VERLENEN VAN TOESTEMMING VOOR HET GEBRUIK VAN DE GEGEVENS AFGELEID UIT DE GEBRUIKERSTEST EN DE EVENTUEEL OPGENOMEN GEBRUIKERSTEST (ONDER DE VORM VAN EEN VIDEOFRAGMENT)

De overeenkomst tussen

1. De tester Mevr./Dhr.
In deze overeenkomst verder aangeduid als “de tester”
2. De facilitator Jeroen Ceyssens
In deze overeenkomst verder aangeduid als “de facilitator”.

Informatie dat wordt verzameld over de gebruiker:

- Kennis van en ervaring met AR systemen voorafgaand aan de studie .
- Kennis van en ervaring met Hololens voorafgaand aan de studie.

Art. 1:

De tester geeft toestemming aan de facilitator om de data, die wordt verworven uit de test, te gebruiken en publiceren in de masterproef van de facilitator. De identiteit van de gebruiker zal te alle tijde anoniem blijven en verzamelde data van de gebruiker zal enkel gelinkt worden aan een unieke deelnemers ID.

Art 2:

De tester geeft aan dat hij/zij

- ☐ opgenomen mag worden. Het bijhorende geluid mag worden gebruikt om data uit af te leiden voor de masterproef van de facilitator. De opgenomen fragmenten worden niet gepubliceerd door de facilitator.
- ☐ niet opgenomen mag worden.

Art 3:

De tester geeft aan dat hij/zij

- ☐ toestemming geeft om de systeembeelden tijdens zijn/haar test op video vast te leggen. De opgenomen beelden mogen worden gebruikt om data uit af te leiden voor de masterproef van de facilitator. De beelden worden niet gepubliceerd door de facilitator.
- ☐ geen toestemming geeft om de systeembeelden op video vast te leggen tijdens zijn/haar test.

Opgemaakt te Diepenbeek op _____

In 2 exemplaren (voor de tester en de facilitator)

De tester (handtekening)	De facilitator (handtekening)
--------------------------	-------------------------------

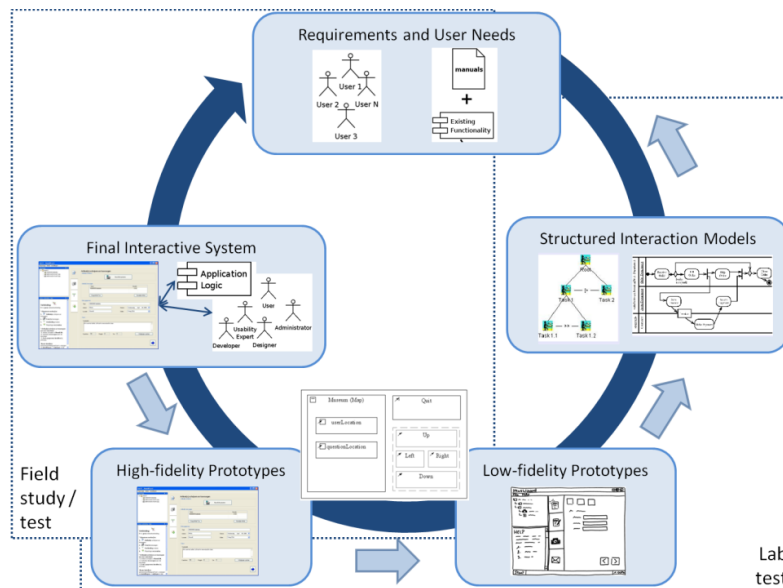
E MuiCSer Explanation

Uitleg: Design Proces van User Interfaces

Het design proces van user interfaces verloopt in het algemeen als volgt:

- Een gebruikersanalyse wordt uitgevoerd voor het bepalen van de requirements en de gebruiker noden.
- Aan de hand van de gebruikersanalyse wordt een taakanalyse gevormd aan de hand van taakmodellen en dialogmodellen.
- Aan de hand van de taakanalyse worden low-fidelity prototypes gemaakt (ook wel mockups genoemd). Dit zijn interface designs in de vorm van tekeningen op papier of via specifieke design tools en zijn al dan niet interactief.
- Aan de hand van de low-fidelity prototypes worden high-fidelity prototypes gemaakt. Dit zijn interactieve en meer uitgewerkte versies van de applicatie. De features lijken echt te werken, maar deze worden achterliggend slechts gesimuleerd.
- Aan de hand van de high-fidelity prototypes wordt een finale versie van het systeem gemaakt met alle features volledig werkend.
- Het hele proces herhaalt zich dan vervolgens bij het maken van een nieuw systeem of het uitbreiden van het huidige systeem met nieuwe features.

Bij iedere fase worden er testen uitgevoerd met eind-gebruikers en/of met stakeholders waarbij de feedback van de testen worden overgenomen naar de volgende fase.



Figuur: MuiCSer Raamwerk voor User-Centered Software Engineering

Tijdens deze test wordt voornamelijk de focus gelegd op de low-fidelity, high-fidelity en de finale fases.

F User Steps Training Phase

Stappenplan Trainingsfase

Gelieve tijdens deze stappen zo veel mogelijk het “Think-aloud” protocol te hanteren.

1. Ontvang de Hololens van de facilitator.
2. Pak de kubus vast met een hand en versleep deze naar een andere locatie.
3. Pak de kubus vast met beide handen en probeer deze te roteren en vergroten/verkleinen.
4. Tik op de kubus en klik op de “Versions” knop in het object menu.
5. Tik op de eerste versie en klik op de “Select” knop in het object menu.
6. Pak de bol vast met beide handen en probeer deze te roteren en vergroten/verkleinen.
7. Pak de annotatie vast en versleep deze naar een andere locatie.
8. Pak de annotatie vast met beide handen en probeer deze te vergroten/verkleinen.
9. Tik op de bol en klik op de “Go back” knop in het object menu.
10. Ga met de cursor over de knop met “(2)” erop en klik vervolgens op deze knop.
11. Klik op de kaartjes onder de “(2)” knop om de beslissingskaarten weg te filteren en weer terug te brengen.
12. Probeer de beslissingskaart te lezen en verklein/vergroot deze om de leesbaarheid te verbeteren.
13. Klik op de “(2)” knop om de beslissingskaart weergave te sluiten.
14. Klik op de “Collapse” knop onder de versies om de versieweergave te sluiten.
15. Geef de Hololens tijdelijk terug aan de facilitator en wacht tot je deze terugkrijgt.

G User Steps Computer Science

Stappenplan AR-Helaba

Gelieve tijdens deze stappen zo veel mogelijk het “Think-aloud” protocol te hanteren.

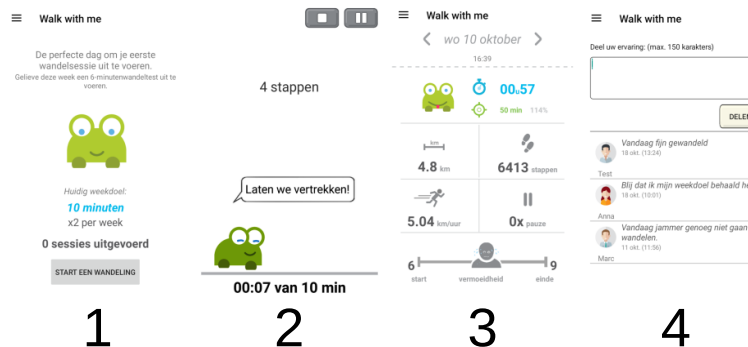
Herhaal de volgende stappen 4 keer:

1. Kijk met de Hololens naar de applicatie op de smartphone
 - a. Als niets weergegeven wordt ga dichterbij de smartphone met de Hololens.
2. Beantwoord voor de huidige applicatie weergave de volgende vragen:
 - a. Wat is er op het eerste zicht veranderd tussen de verschillende versies?
 - b. Welke belangrijke beslissingen zijn er gemaakt tijdens het ontwerpen van de huidige applicatie weergave?
 - c. Wat zijn de redenen dat deze beslissingen gemaakt zijn?
3. Geef de Hololens terug aan de facilitator en vul de vragenlijst in.

H Guidor Steps Computer Science

Computer Science

1. Save log of previous test and clear the current log.
2. Greet user and ask to fill in consent form twice (one copy for tester, one for facilitator).
3. Meanwhile start the servers 0 and 1.
4. Greet user and ask the following questions:
 - a. Do you know the concept of AR?
 - b. Have you ever used Hololens or anything of AR before?
 - c. Did you ever have an epileptic attack or are you sensitive to epilepsy inducing visuals?
 - d. Are you familiar with the design process of user interfaces in software? (If they aren't familiar, quickly give them an overview of the process).
5. Mention the following to the user:
 - a. How the testing will proceed (you will openly discuss matters with them as they are asked to solve some questions)
6. Start AR-Training and give hololens to user.
 - a. Guide the user through the mentioned steps of the training phase.
 - b. Explain to the user what each component in the step means (versions visualization for displaying versions, cards selection to filter, etc)
7. Ask Hololens back from user and start AR-Helaba1 application.
8. Mention the following to the user:
 - a. Goals and general overview of the application goals.
 - b. How the testing will proceed (users will explore application views + what questions do users need to solve for each view)
9. Place smartphone in front of user showing the the current application view.
 - a. Look at the Hololens of the user, you can see light appearing when the model has been detected.
 - b. If model detected but not visible, ask user to take one step backwards.
10. Ask user to fill in questionnaire, meanwhile collect recordings from Hololens.



Application View Combinations (Latin Square):

- 1-2-3-4
- 2-4-1-3
- 4-3-2-1
- 3-1-4-2

Answers for every view:

1. Before walk view:

- They switched from landscape to portrait mode.
- First it was all about setting the amount of km or the amount of steps to walk, but this was changed to the amount of time to walk.

2. Start walk view:

- They switched from landscape to portrait mode.
- They added the progress in numbers and changed the units of numbers later from steps to both steps and time.

3. Status of walk view:

- At first the character provided the statistics after the walk, but this got changed to an overview of the day which can easily be switched to another date.

4. Commentary of walk view:

- At first you could only point out how tired you were from the walk, but afterwards textual feedback was added to provide a more clear description of the walking experience.
- Finally a community was added to the system where walk experiences can be shared.

Questions for scenario's:

- After handling Scene 2: Do you understand what the split in the visualization implies?
- After handling Scene 3: Did you expect there to be a decision card between the final two versions? Do you think there should have been a decision card?
- After handling all scenario's: Do you think a system like this can be useful for developers?

I Computer Science Questionnaire

Please answer the following questions honestly, remember it's the system we are testing and not you.

1. Choose one option for each statement that you find is the most suitable for you.

		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
	Statements	1	2	3	4	5
1	I think that I would like to use this system frequently.					
2	I found the system unnecessarily complex.					
3	I thought the system was easy to use.					
4	I think that I would need the support of a technical person to be able to use this system.					
5	I found the various functions in this system were well integrated.					
6	I thought there was too much inconsistency in this system.					
7	I would imagine that most people would learn to use this system very quickly.					
8	I found the system very cumbersome to use.					
9	I felt very confident using the system.					
10	I needed to learn a lot of things before I could get going with this system.					

2. Do you have any remarks to add about any of the previous statements?

3. Do you have any recommendations on how we can improve the system?

4. Any other final remarks?

J Computer Science Study Questionnaire Results

User	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	2	3	3	4	3	3	3	3	2	3
2	3	4	3	4	4	4	3	4	4	4
3	4	4	3	3	4	4	4	3	3	4
4	1	3	2	2	3	3	2	1	1	1
5	2	3	3	3	3	3	2	3	3	2
6	2	3	3	2	3	3	3	2	1	3
7	2	3	3	4	3	3	3	2	3	3
8	2	1	3	3	3	3	3	2	2	3

Figure 64: SUS results for every question and user for the Computer Science study

K User Steps Architecture

Stappenplan Major's House

1. Ontvang de Hololens van de facilitator.
2. Kijk naar de maquette en tel de hoeveelheid trappen in het gebouw.
3. Bekijk de versies van het model en ontvang instructies van de facilitator omtrent hoe hierover te navigeren.
4. Navigeer naar de eerste versie van het model.
5. Navigeer naar de versie met de zetels op het balkon.
6. Navigeer naar de versie met de zetels op het terras.
7. Tik op het model en klik op de "Show changes" knop in het object menu.
8. Probeer uit te leggen welke veranderingen en beslissingen gemaakt zijn in het model.
9. Geef de Hololens terug aan de facilitator en wacht tot je deze terugkrijgt.
10. Geef de Hololens terug aan de facilitator en vul de vragenlijst in.

Stappenplan Conceptual Thinking

1. Ontvang de Hololens van de facilitator.
2. Kijk naar de weergave van Genk.
3. Tik op dit model en klik vervolgens op de “Show links” knop.
4. Volg de link en bekijk de beslissingskaart, leg uit hoe de objecten gelinkt zijn en wat de vermelde redenen zijn. Wat betekent de pijl?
5. Tik op het gerendered model en klik op de “Show links” knop.
6. Volg de link met het label “Follows from” tot het conceptueel design zichtbaar is.
7. Zoek en leg uit wat de betekenissen zijn van de kleuren van het conceptueel design (model 4).
8. Vermeld uit welke delen de achterkant van de poster bestaat.
9. Verplaats het conceptueel design (model 4) tot voor het middelste raam.
10. Open de versies van het conceptueel design.
11. Open de versies van het grootste fysieke model (model 4.2) en leg uit wat er veranderd is.
12. Sluit de versie weergave van het fysiek model door op de “Collapse” knop te drukken.
13. Leg uit welke iteraties van het conceptueel design gevolgd zijn naar het finale design toe en welke keuzes hierbij gemaakt zijn.
14. Geef de Hololens terug aan de facilitator en vul de vragenlijst in.

L Guidor Steps Architecture

Architecture

1. Save log of previous test and clear the current log.
2. Greet user and ask to fill in consent form twice (one copy for tester, one for facilitator).
3. Meanwhile start servers 0, 2 and 3.
4. Ask the following questions:
 - a. Do you know the concept of AR?
 - b. Have you ever used Hololens or anything of AR before?
 - c. Did you ever have an epileptic attack or are you sensitive to epilepsy inducing visuals?
5. Mention the following to the user:
 - a. How the testing will proceed (you will openly discuss matters with them as they are asked to solve some questions)
6. Start AR-Training and give Hololens to user.
 - a. Guide the user through the mentioned steps of the training phase.
 - b. Explain to the user what each component in the step means (versions visualization for displaying versions, cards selection to filter, etc)
7. Ask Hololens back from user and start AR-Helaba2 application, align the digital model with the real model and adjust models if necessary.
8. Mention the following to the user:
 - a. Goals and general overview of the Major's house project.
 - b. How we will openly discuss the changes.
9. During the versions visualization, explain to users how to navigate through the models.
 - a. Mention to carry out the gestures "nonchalantly".
10. Ask user to fill in questionnaire for system 1 and ask Hololens back.
11. While user is filling in questionnaire, start AR-Helaba3 and align the digital models with the real models, adjust models if necessary.
12. Ask user to fill in questionnaire, meanwhile collect recordings from Hololens.

Answers for maquette:

- Placeholder for garage/shed removed (would not be part of maquette in the end)
- The couches were not part of the maquette, since these are not part of the building itself and can be removed at any time.

Answers for conceptual designs:

- Reason for placing the detention centre in middle -> public routes connection
- Colors:

 Gedetineerd Wonen (30 x 21m ² = 630m ²)	 Ontspanning/Herintegratie (370m ²)	 Penetentiar vervoer (25m ²)
 Beveiliging (90m ²)	 Single/3G Wonen (300m ²)	 Buitenruimte (1275m ²)
 Zorgfuncties (40m ²)	 Ontspanning Dubbel Gebruik (170m ²)	

- Reason for iteration selection: Most amount of housing + separating public/private spacing + Still enough recreational buildings.

Questions for scenario's:

- After toggling the changes visualization: What do you think that the colors red and green imply for this visualization?
- After handling all scenario's: Do you think the system can be useful in the field of architecture?

M Architecture System 1 Questionnaire

Questionnaire System 1

Please answer the following questions honestly, remember it's the system we are testing and not you.

1. Choose one option for each statement that you find is the most suitable for you.

		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
	Statements	1	2	3	4	5
1	I think that I would like to use this system frequently.					
2	I found the visualizations unnecessarily complex.					
3	I thought the gestures were easy to carry out.					
4	I found the system very cumbersome to use.					
5	I would imagine that most people would learn to use this system very quickly.					
6	I did not feel very confident using the system.					
7	I found the various functions in this system were well integrated.					
8	I thought the models in the system were not clearly visible.					

2. Do you have any remarks to add about any of the previous statements?

N Architecture System 2 Questionnaire

Questionnaire System 2

Please answer the following questions honestly, remember it's the system we are testing and not you.

1. Choose one option for each statement that you find is the most suitable for you.

		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
	Statements	1	2	3	4	5
1	I think that I would like to use this system frequently.					
2	I found the visualizations unnecessarily complex.					
3	I thought the gestures were easy to carry out.					
4	I found the system very cumbersome to use.					
5	I would imagine that most people would learn to use this system very quickly.					
6	I did not feel very confident using the system.					
7	I found the various functions in this system were well integrated.					
8	I did not understand the structure of the project completely.					

2. Do you have any remarks to add about any of the previous statements?

3. Which system did you prefer and why?

O Architecture Study Questionnaire Results

User	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	3	3	3	3	3	3	3	2
2	3	3	2	2	4	2	2	3

Figure 65: Questionnaire results for every question and user for the Architecture Study System 1

User	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	3	3	3	3	3	3	3	3
2	3	2	2	3	4	2	3	3

Figure 66: Questionnaire results for every question and user for the Architecture Study System 2