BACHELOR THESIS

AUGMENTED REALITY IN THE INDUSTRY

TO PROVIDE A MACHINE OPERATOR WITH INSIGHTFUL PROCESS INFORMATION

BENJAMIN SUIJDENDORP / 1967118

CREATIVE TECHNOLOGY / FACULTY OF EEMCS

SUPERVISORS:

K. SLEURINK (MA-IT)

DR. J. ZWIERS (UNIVERSITY OF TWENTE)

DR. IR. D. REIDSMA (UNIVERSITY OF TWENTE)

UNIVERSITY OF TWENTE.

AUGUST 2022

Abstract

The automation industry produces a lot of data. In this thesis, augmented reality is used to investigate whether it can provide a level of insight to an operator that a regular Human machine interface can not. By building a software solution that provides augmentation on a live video feed, user tests could be conducted with experts in a testing facility. The outcomes of the user tests were that in general a device with augmented reality can provide extra insights. Problems like information overload, glitching and underdeveloped augmented reality are still evident and prevent the technology to be widely used. With more developed hardware and more integrated software, augmented reality can be an increasingly interesting technology to streamline communication between automated processes and humans.

Keywords: *Augmented reality, Automation industry, Information, Data, Cloud based, Hardware, Software , Engineering, Design, User experience, User centered design, Human machine interface, mixed reality*

Acknowledgments

First, I would like to thank both supervisors from the University of Twente, dr. Job Zwiers and dr. ir. D. Reidsma for the support, feedback and helpful tips during my graduation project. Secondly, I want to thank Koen Sleurink for providing me with the opportunity to perform my graduation project at MA-IT and his guidance, help and insights in the process. I want to thank Marc Waarle, Fabian Grootfaam and Rudy de Bock for participating in discussions and answering my many questions. Without them, I would not have been able to obtain the information I needed for this thesis. Finally, I would like to thank God, my family, friends and colleagues at MA-IT for their unwavering support, help and ideas during these months.

TABLE OF CONTENTS

Contents

List of abbreviations 6
1. Introduction 7
  1.1 Project 7
  1.2 Research questions 8
  1.3 Chapter overview 8
2. Background Research 9
  2.1 Introduction to Augmented Reality 9
  2.2 Deployment of AR in industries 10
  2.3 Hardware solutions 14
  2.4 Software solutions 16
  2.5 Challenges in AR 16
  2.6 Conclusions 17
3. Requirements caption and Ideation 18
  3.1 Stakeholder identification 18
  3.2 Designer scenarios 19
  3.3 Idea generation 23
  3.4 Augmented reality as an innovation 23
  3.5 Product idea 26
  3.6 Specification 28
4. Realization 30
  4.1 Methods and Techniques 30
  4.2 System architecture 32
  4.3 Iterations 37
5. summary of Evaluations 44
  5.1 Intermediary evaluations 44
  5.2 Final Prototype evaluation 45
6. Discussion 47
  6.1 Conclusion 47
  6.2 Recommendations 48
7. Bibliography 50
Appendix 1 : Code 54

FIGURE 1. THE LEMONADE FACTORY 7
FIGURE 2. MILGRAM'S MIXED REALITY CONTINUUM 9
FIGURE 3. 3D OVERLAY OF OPERATION SITE A MINIMALLY INVASIVE OPERATION 10
FIGURE 4. STRUCTURE OF THE ARCHITECTURE INSIDE OF A BUILDING 11
FIGURE 5. AUGMENTED VIEW OF DAISHUFA 11
FIGURE 6. AR IN A MUSEUM 12
FIGURE 7. WIKITUDE NAVIGATION APP 'DRIVE' 12
FIGURE 8. TES WITH AR ON A MILITARY VEHICLE 12
FIGURE 9. A SCREENSHOT OF THE TIP4SUPPORT APP 12
FIGURE 10. A DUCT CONSTRUCTION AUGMENTED REALITY APPLICATION. ON THE LEFT THE OBJECT BASED TRACKERS ARE VISIBLE. 13
FIGURE 11. A SOLDIER WEARING THE IVAS; ON THE BOTTOM RIGHT ARE CAMERAS THAT PROVIDE VISION FOR PASSENGERS THROUGH THEIR HMDS 14
FIGURE 12. EXPERIMENTAL BARS SETUP 14
FIGURE 13. REALWARE HMT-1 14
FIGURE 14. VUZIX M4000 14
FIGURE 15. CONCEPT FOR ARAM 15
FIGURE 16. THE LEMONADE FACTORY 24
FIGURE 17. EXAMPLE OF HOW TEXTBOXES CAN BE ANCHORED TO A REAL LIFE OBJECT 25
FIGURE 18. SCREENSHOT OF THE HMI OF THE LEMONADE FACTORY 25
FIGURE 19. INITIAL OVERVIEW 27
FIGURE 20. PRESSED THE TOP RIGHT POINTER 27
FIGURE 21. PRESSED ON THE BOTTOM POINTER WITH HIGHLIGHTED DUCT 28
FIGURE 22. WATERFALL MODEL 30
FIGURE 23. ITERATIVE MODEL 30
FIGURE 24. USE CASE DIAGRAM 32
FIGURE 25. SYSTEM OVERVIEW 33
FIGURE 26. SYSTEM 33
FIGURE 27. MOBILE DEVICE 34
FIGURE 28. AUGMENTED REALITY APPLICATION 34
FIGURE 29. AUGMENTED REALITY TRACKING 35
FIGURE 30. STATIC USER INTERFACE 36
FIGURE 31. DATA HANDLING 36
FIGURE 32. 3D OBJECT OF THE LEMONADE FACTORY IN A MODEL TARGET GENERATOR (VUFORIA) 38
FIGURE 34. A CUBE AUGMENTED ON AN IMAGE TARGET 39
FIGURE 33. PARALLAX EFFECT 39
FIGURE 35. APPLICATION REQUEST FIELD AND REPLY FROM DATABASE 39
FIGURE 36. AZURE CLOUD SERVICE PORTAL 39
FIGURE 37. FUNCTIONAL TESTING - AR, DATA, 3D VISUALISATION 41
FIGURE 38. FUNCTIONAL TESTING – DATA REPRESENTATION 41
FIGURE 39. TEST WITH AR AND STATIC TEXT 41
FIGURE 40. WARNING CTA 42
FIGURE 41. AR SCREEN 42
FIGURE 42. LOGIN SCREEN 42
FIGURE 43. ITERATIONS OF CTA'S 43
FIGURE 44. 3D OVERLAY 43
FIGURE 45. STATIC INFORMATION DISPLAY 43
FIGURE 46. 3D INFORMATION DISPLAY 43

LIST OF ABBREVIATIONS

Abbreviation	Description
AR	Augmented reality
VR	Virtual reality
HHD	Hand held device
HMD	Head mounted device
OEE	Overall equipment effectiveness
PACT	People, activities, context, technology
MRO	Maintenance, repair, overhaul
TES	Through-life Engineering Service

1. INTRODUCTION

1.1 PROJECT

Description and aim

In the automation industry it is paramount to provide an operator with insight in a machines operations. In Figure 1 such an automatic machine is visible. It is a small scale lemonade factory, which is operated by use of several sensors and actuators such as valves and motors. Although the machine should be able to catch of all cases – problems or not – sometimes human intervention is needed. A human machine interface is used to provide a way to intervene and visualize data for an operator. However, this interface is not always as intuitive and extensive and removes a lot of valuable data for the case of simplicity. A potential solution to keep an operator more connected to the data could be the use of Augmented reality. Augmented reality can be used to create overlays which help visualize data and connect it to the machine in a meaningful way.

Figure 1. the lemonade factory

In order to supply this, the data needs to be transported between the lemonade factory and the application. On request of the stakeholder MA-IT networking and databases were investigated and implemented.

Exactly how augmented reality could be leveraged was the main area of investigation. This required researching what the current state and sustainability of the technology was. The applications interface was user tested and evaluated following HCI design philosophies and principles as taught in the Create curriculum [5].

1.2 RESEARCH QUESTIONS

The situation description, problem statement, project requirements and challenges were condensed in smaller problem statements and subsequent sub questions, in an attempt to answer the main research question.

[AR+DATA+UI] RQ: *How can augmented reality be used in an automated lemonade factory to help specified personnel gain greater insight in the operational data of the machinery compared to static schematic representations?*

Four areas of interest will be examined. Firstly, the domain of augmented reality needs to be researched, in order to see what other companies are doing and what technologies have been developed and are currently being developed. This results in the first sub question:

[AR] SQ: *What augmented reality techniques are currently developed and used for the automation industry in the area of maintenance, repair and operations (MRO)?*

When the current techniques and technologies are extensively examined, provisional conclusions about usable technologies tailored to this specific case can be made. These conclusions are an answer to the following sub question:

[AR] SQ: *What available AR technologies are most suitable for the lemonade factory in order to create an augmented reality experience?*

The stakeholder has expressed their interest in overlaying operational information on the sensors of the lemonade factory on the machine. Since a mobile phone will be used, it has to connect to the server or cloud service where the data is stored. The stakeholder wants to use an IFM Edgecontroller to upload the data to a cloud service. There are multiple cloud options – think AWS, Google, Azure etc. – the following sub question has to be answered:

[DATA] SQ: *How can data be uploaded to and downloaded from a cloud service using an Edgecontroller and mobile phone?*

All the data had to be displayed in a way that is clear and concise, without being a distraction. This was designed and evaluated following the design principles of Create [5], in order to answer the final sub question:

[UI] SQ: *How can information be displayed in an AR application in order to enhance the insight in operational data?*

1.3 CHAPTER OVERVIEW

In this thesis several sections can be discerned. In the first section a literature research was conducted on the technology of augmented reality, as well as available hardware and software. Based on this research and requirements of the stakeholders, ideas were generated and the specifications for a prototype augmented reality solution were drawn up. After this, the realization of the prototype is described; this design phase was executed in an iterative manner, with intermediate user tests. Following, a summary of the evaluations and user testing is given, to be concluded by a final conclusion and recommendations for further development of augmented reality in the automated industry.

2. BACKGROUND RESEARCH

2.1 INTRODUCTION TO AUGMENTED REALITY

To research Augmented reality properly, it needs to be defined. In this paper, the definition given by Azuma et al. will be adhered to: *'AR is […] supplementing the real world with virtual (computer-generated) objects that appear to coexist in the same space as the real world'* [6]. In addition, Azuma et al. established three ground rules for a device to be considered Augmented reality:

1) combines real and virtual,
2) interactive in real time, and
3) registered in 3D [7].

These three principles will be used to examine whether a product or application is actually Augmented reality. Milgram et al. have proposed a continuum between reality and virtuality [8]. On this scale, an application can focus more towards the real world or a virtual world, in Figure 2 this scale is visible.

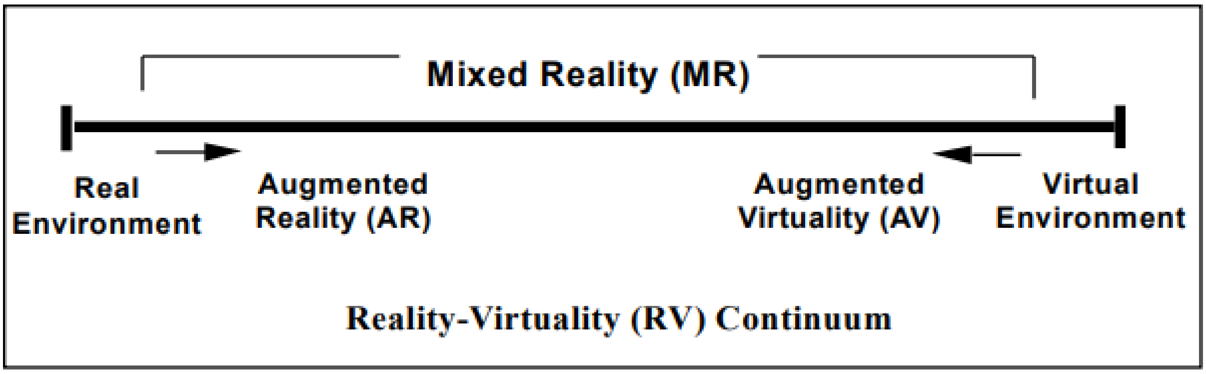


Figure 2. Milgram's Mixed reality continuum

The scale goes from a completely real environment without any digital attributes to a completely virtual environment, where there is no real objects present. Between these two extremes everything is categorized under Mixed Reality. On the left part of the scale, there are Heads-up displays. Heads-up displays as used in fighter jets or helicopters help to keep the pilots attention on the sky, and not down towards the instruments. These HUD's also connect to targets in the real world (first requirement of Azuma). Then there are assisted reality glasses (aR). They are technically not augmented reality, because they provide information about the real world but are not interactive with it. However, they are still included in this essay, because there is a great deal of overlap with this technology and augmented reality. It is placed on the left on Milgram's continuum. Augmented virtuality is the opposite of Augmented reality; there is a virtual environment which is augmented with real life objects – e.g. a webcam feed or a hand that is isolated from its real environment and placed in the virtual one.

The general consensus is that augmented reality is already and will grow to be an important part of industry, but there are still considerable challenges to overcome [9]. The objective of this background research therefore is to gain an insight in the use of augmented reality in the automated industry and to give an overview of its possibilities and limitations in installations. In order to gain this insight, related products will be examined in the second paragraph, and following that a state of the art of AR applications will be discussed. The conclusion will contain recommendations about and points of attention for the realization phase.

2.2 DEPLOYMENT OF AR IN INDUSTRIES

There are many deployments of AR in many different fields. While not all domains are directly relatable to this research, there can be technologies, findings, recommendations or warnings in them that are useful to consider in this application. These will be taken into account for the idea generation. Augmented reality deployed in healthcare, ACE, Guidance and closer related to this project Automotive, Maritime and Railway industries and the military will be discussed.

Healthcare

In healthcare augmented reality technologies have found their use. In 1994 a prototype was made for displaying ultrasound imaging [10]. By means of a HMD the ultrasound technician could see the 3D image of the fetus projected on the stomach of the expecting mother. Another application of AR in healthcare is the visualization of internal organs and the position and orientation of surgical instrumentation during the operation [11]. In modern medical systems a video feed of an endoscopic camera is displayed on a screen. This is an aid to visualize the operation site and position, orientation and movements of the instruments inside of the patient. This is a great challenge to surgeons as they have to integrate the visual information in the model of the operation site in their head. With an HMD the operation area can be visualized outside of the body, and give greater freedom to the surgeon by removing the challenge to map the video feed to his mental image of the operation site; an example is visible in Figure 3. This principle pinpoints a solution to the problem of unintuitive

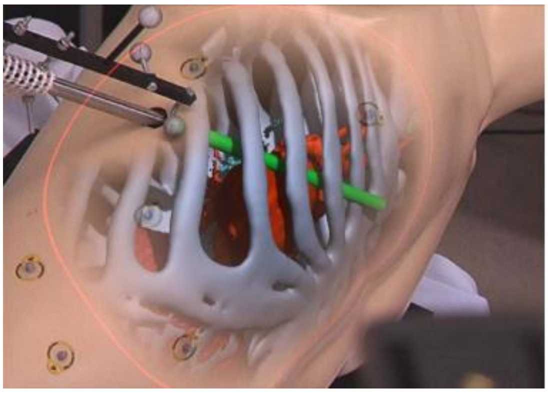


Figure 3. 3D overlay of operation site a minimally invasive operation

interfaces, which are found in automation industries too. The augmentation that is proposed here can be adapted and used in this project.

ACE (architecture, construction, engineering)

In the open air environment augmented reality is used in several fields: architecture, construction and engineering will be closely examined. Huang et al. [12] presented a way to augment the ruins of a historical site called Yuanmingyuan. In the Qing dynasty several emperors build this vast and magnificent royal garden. It was destroyed in 1860, and there have been discussions whether or not to restore it to its former glory. Huang et al. developed a static augmented reality device and presented it to the visitors of the site with the opportunity to get an idea of how the Dashuifa (Great

Waterworks) would have looked like, see Figure 5. Users gave feedback that implicated that this is a welcome way to interact with the history of the site. Answers to an interview indicated that especially ease-of-use of the device and 3D representation of the building are areas that needs to be improved.

Figure 5. Augmented view of Daishufa

Figure 4. Structure of the architecture inside of a building

Webster et al. [13] have created an AR solution to visualize structural data in architecture, which can be seen in Figure 4. Digital overlays present information considering positioning of re-bar – steel structures that greatly strengthen the concrete and a structural analysis, based of a CAD model of the building. AR is extensively suggested for architecture; [14] present multiple use cases for architecture. Examples are construction, inspection, renovation, facility management and maintenance, urban planning and locating pipelines under roads.

Guidance

There are two types of guidance: on one hand there is guidance in a museum, where people can walk around and scan objects. On the other hand there is pathfinding, like Google maps, Waze and Apple maps . Both types of guidance are discussed here. Miyashita et al. [15] have created a robust AR system in a museum, where users got a handheld device and instructions on the location of the AR waypoints. When a waypoint was reached, an AR animation would start playing automatically (see **Fout! Verwijzingsbron niet gevonden.**). The response of the users was in favor of AR, there were exclamations of surprise and excitement, which was strengthened by the interviews afterwards. A few problems were discovered nonetheless. A few AR points were skipped, due to there being not a clear sign of where the AR application could perform its actions. In the conclusion Miyashita et al. discuss future possibilities: using SLAM (simultaneous localization and mapping) could enable their wishes that AR is not limited to certain positions.

Another innovative use of AR in guidance is road navigation. Taking routes might be non-intuitive for some people. Wikitude has created an app that helps people navigate by showing the route overlayed on a video feed (see **Fout! Verwijzingsbron niet gevonden.**) [16]. They claim it eliminates the need to look at a chart and 'has the safety benefits of never having to take your eyes off were they're going'. Still, in reviews of this app, there are safety concerns. The field of view of the camera is too narrow to provide the same amount of information that your eyes would, and the augmented reality overlay has too much lag to provide any safety benefits [17].

Figure 6. AR in a museum

Figure 7. Wikitude navigation app 'Drive'

Automotive

AR is currently deployed in many fields of the automotive industry [9]. Volkswagen is a prevalent user of this technology; not only in their cars (during reverse driving), but also for training new employees in manufacturing or inspections, in the design of new buildings and assistance during repairs [18]. TIP4Support has created an augmented reality app where technicians have a live video communication, where a technician can remotely draw on his screen, which then shows up on the screen of the first technician [19]. The augmented reality aspect of this app makes it novel: the drawing is also anchored to the real world, which gives the technicians greater flexibility in their way of working. In Figure 9 the application is visible, with several functions being displayed, including drawn circles and made notes. In the evaluation the technicians were surprised to see how it improved their workflow and decreased the

Dini and Dalle Mura identified many uses of AR in the automotive industry [20]. Their main focus was on the role that AR can play in Through-life Engineering Services (TES). TES are procedures that are executed over the lifespan of a product, for example a car. These services can include periodical checks, which are often standardized by the manufacturer. The downside to these checks is that they require extensive information for the executing mechanic to perform the correct operations. The information is often accessible in big paper manuals, which are cumbersome to read and subject to wear and tear, especially in 'on site' conditions. AR can accelerate these procedures by providing

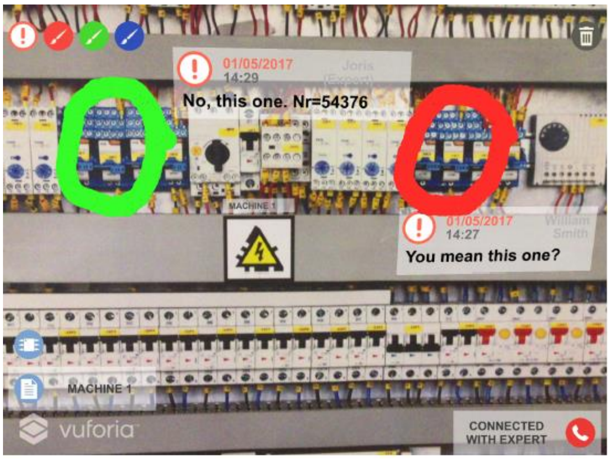


Figure 9. A screenshot of the Tip4Support app

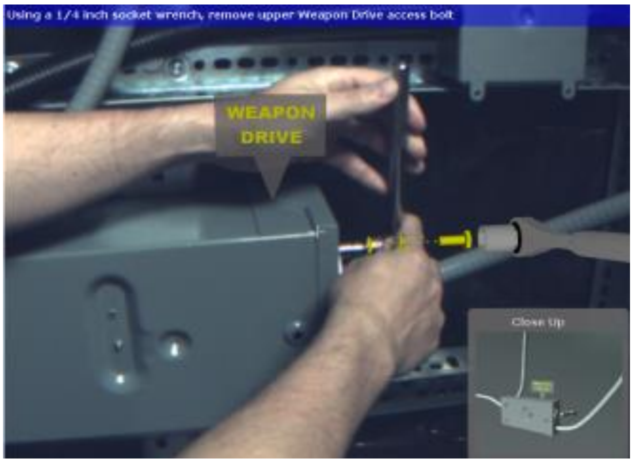


Figure 8. TES with AR on a military vehicle

tailored information, preventing information overload. Figure 8 shows how a TES performed through an AR device can look [21].

Maritime and Railway industries

Not only the car industry uses AR, Fraga-Lamas et al. [22] lists the uses of AR in a shipbuilding industry: it is being used as quality control, assistance in manufacturing process, visualization of location of products and tools, management of warehouses, predictive maintenance by datamining (cloud services), augmented coms, visualizing installations in hidden areas, remote operation of IIoT (Industrial Internet of Things) and smart connected devices. As an example of quality control and assistance in manufacturing, Olbrich et al. [23] used AR to determine discrepancies between CAD models of pipes in shipbuilding and the real life installation. The software also takes the CAD models and alters them based on real life pipe situations, using a calibration method and image recognition. In Figure 10 this application is shown. Similar to automotive and maritime industries, rail logistic companies can also benefit from AR, for the same reasons as other industries: the complex machines require more paperwork to describe the procedures, which can be done more efficiently by AR, in a

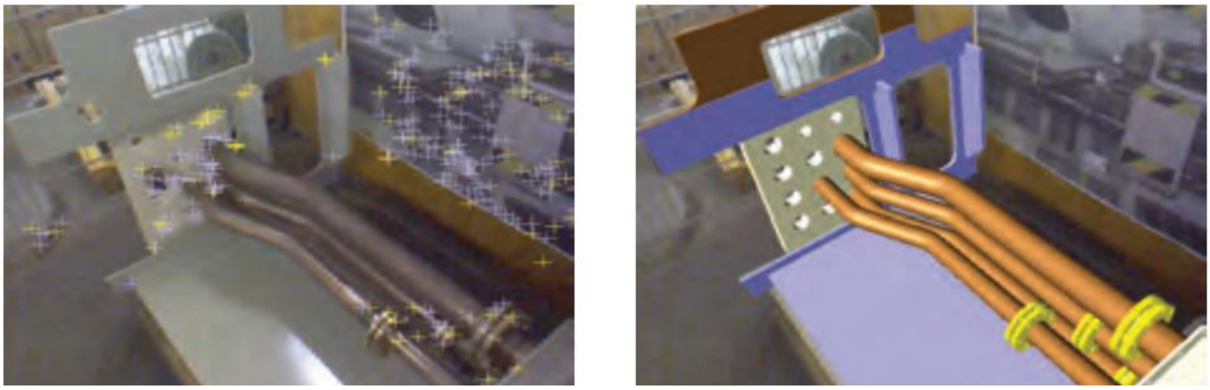


Figure 10. A duct construction augmented reality application. On the left the object based trackers are visible.

personalized way. Together with the healthcare visualization, the overlay of pipes and 3D models can be used to gain a greater insight in the system.

Military

The military is potentially at the forefront of development of AR. Livingston et al. [24] have identified four potential problem areas where AR can encompass a potential solution, including but not limited to situational awareness, information overload, training and quick-reaction forces. They created the Battlefield Augmented reality System, visible in Figure 12**Fout! Verwijzingsbron niet gevonden.** [25]. This was a land warrior in development in the early 2000's. Together with Microsoft the US military is developing IVAS: Integrated Visual Augmentation System [26]. It is a HoloLens that is specifically designed and adapted towards the military, in order to increase efficiency in pathfinding, night vision, target acquisition and terrain awareness during transit. While still in the testing phase, there are great improvements in time required for pathfinding. In Figure 11 a soldier is wearing an IVAS.

Figure 12. Experimental BARS setup

Figure 11. A soldier wearing the IVAS; on the bottom right are cameras that provide vision for passengers through their HMDs

2.3 HARDWARE SOLUTIONS

Head mounted displays (assisted Reality and Augmented reality)

Potentially the most famous example of AR devices are the failed google glasses. They were introduced in 2012, but failed to live up to the expectations. There were elements of fear what this technology could mean in terms of privacy of the surrounding people. Besides, Google failed to address the added value of the glasses; it assumed that the hype over the innovative technology was enough to sell the glasses. When this turned out not to be the case, the production and sales of Google glasses were halted, following the release of the beta version in 2015 [27]. Shortly after, Google announced the production of Google Glass Enterprise edition and more recently Google Glasses Enterprise edition 2, which are, as the name suggests, not available for private ownership [28].

The Microsoft HoloLens 2 [29] has grown to a well known device in AR field. Where the Google Glass focused more on providing information, the HoloLens is well suited for more complex design operations. It excels in group efforts, for example designing a part for an airplane engine or a surgery. This versatility and 3D modelling in the real space – meaning you can 'walk around' the virtual object in the real world – gives it possibilities that other HMD's do not have.

Head mounted displays can be further specified and categorized in two streams. Google Glasses are more focused towards assisted reality. Being very similar to a HUD (heads-up display), assisted reality is constantly in the field of view of the wearer, but it differs from augmented reality in the sense that

Figure 13. Realware HMT-1

Figure 14. Vuzix M4000

the information on the display is not interactive with the real world. Assisted reality devices are mostly attachments to or integrated in glasses [30]. An example is the head mounted Realware HMT-1, visible in Figure 13 [31]. Although similar to the Google Glasses, they have a screen instead of a see-through display. A few of the proposed use cases are remote help, digital aid and IoT connected engineer. More interactive with the real world is the Vuzix line up: in stead of a non-transparant screen, there are several models with Waveguide optics. This is a technology that reflect the light inside of the glasses. Examples are the Blade, M400, M400C and M4000 [32]. The latter is shown in Figure 14.

Hand held devices

Besides HMD's there are also HHD's, hand held devices. They are more widespread to the general audience and less specialistic: one such a device could be a mobile phone or tablet. Newer android or iOS devices with camera's and enough computing power can run apps such as Snapchat: a social media app which uses facial recognition and an AI point map to apply filters to a persons face [33]. Samsung devices have a similar built in application: AR Zone [34]. In the MR continuum that was presented earlier, these apps fulfill all 3 requirements for an AR application. They combine real and virtual, are interactive with the real world and make use of 3D.

The advantages of a handheld system are among other things the ease of use; many people have phone or tablet in personal use, so the learning curve to use this is smaller compared to a HMD. Moreover, it is less invasive and does not interfere with potential glasses. There are drawbacks to HHDs too: it requires hands to hold, especially with maintenance tasks this can be a problem. Having to put the HHD aside every time a task has to be performed takes precious time and concentration to remember the task to be done. A HMD keeps the hands free and uses voice commands to navigate. A HMD is often more immersive too, merging the real world and augmentations. This helps with navigating in the real environment, because there is a real depth perception. HHDs, which don't have these advantages, take concentration away from the real environment, and should therefore be handled with more care for the surroundings [35].

Spatial augmented reality

A third hardware application is spatial augmented reality. This technique does not require the user to wear or carry any type of equipment, but it projects the digital designs on the surface. Bode made such a system: Augmented reality assisted manufacturing (ARAM) projects the places where bolts have to be mounted [36]. This method increased the accuracy as well as the efficiency. In Figure 15 the concept setup of this system is shown. A similar research has been conducted by Doshi et al. [37].

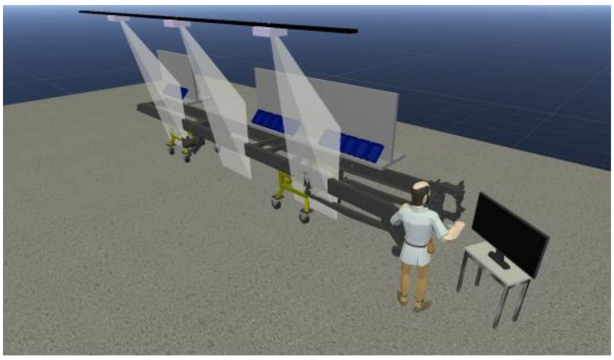


Figure 15. concept for ARAM

Their research provided a digital overlay on the surface that showed the mechanic where studs had to be welded on a frame. This system yielded an improvement of 50% in accuracy.

2.4 SOFTWARE SOLUTIONS

Considering the amount of hardware that has become available, there is also a lot of software available for the development of AR apps. Many of the HMDs work on Android. The options that this platforms gives are plenty: it is a robust framework that enables developers to create custom solutions.
Google has introduced AR core in android, which is a low level software library that can be addressed during the creation of a potential AR app. The difference with the apps already mentioned above is that AR Core is the software that Snapchat and AR zone use. iOS also have their own AR software library, called ARkit. This serves generally the same purpose as AR core. Other options are presented in Table 1 Overview of AR development tools. Not included is 3D modelling software.Table 1.

Name	**Platform**	**Category**
Vuforia	Android/ iOS	AR SDK
Wikitude SDK	Android/ iOS	AR SDK
Effect house	Android/ iOS	TikTok AR filter
LAYAR SDK	Android/ iOS	AR SDK
LENS studio	Android/ iOS	Snap AR filter creation tool
SPARK AR studio		AR tool developed for Instagram by META

Table 1 Overview of AR development tools. Not included is 3D modelling software. [1]

The two biggest competitors for the realization of the augmented reality app for the lemonade factory are PTC's Vuforia and Wikitude SDK. They are very similar in functionalities and can both produce great results. Vuforia has been active a lot longer in the industrial environment with around 30 years. Wikitude came around in 2008, rendering what they have achieved so far quite impressive. Considering the scope and target area of this project, the user base is important. Vuforia takes the crown in this occasion, with clients such as Infiniti, Lego, Lykan, Mercedes-Benz and Royal Delft [38]. This is with the reasoning that client support is a factor that has to be taken into consideration.

2.5 CHALLENGES IN AR

However, even with all these uses and possibilities, AR is still not fully developed. There are many technical challenges that need to be overcome. De Souza Cardoso [9] observed that object tracking was a recurring problem. Either the software was underdeveloped or the object to be recognized is easily obstructed by other items, causing the software not to see the target object correctly . Other problems, that Azuma et al. have pointed out, are discrepancies in overlaying due to measuring errors, excessive strain on the eyes due to the unusual viewing style, delays in the overlaying graphics due to inadequate hardware and excessive information being displayed on the screen [6].

Besides technical challenges, there are other domains where development of AR is halted. Firstly, application specific advancements are often kept a company secret, in order to keep a competitive

[1] This list is taken from: https://en.wikipedia.org/wiki/List_of_augmented_reality_software

edge [9]. Secondly, the vicious circle of not investing in innovative products makes innovating in AR hard. Managers need to make decisions that will further the company, and if one doesn't see AR in that picture, there will be no money available for the development of an AR application [39]. Thirdly the user-AR interaction has to be improved. Palmarini at al. [40] demonstrates that the authoring solutions, content management tools and visualization and ergonomics still need revamping in order to make AR a more widely accepted technology.

2.6 CONCLUSIONS

Augmented reality has found a place in many industries and is continuing to grow, following the forecasts. Fields such as automotive, maritime, manufacturing, maintenance, military, navigation and architecture increasingly often make use of AR at greater or lesser extent. Besides work field uses, it has also proven its worth in training of new engineers. By using AR instead of paper or digital manuals, time is saved and less mistakes are made, both in the field and during studying. This provides greater confidence to a user of AR when performing the task in reality.

Besides, the discoveries made by several researches will be taken into account as well. For example, the detection of objects can be difficult. This problem will be looked at in order to improve the accuracy of the app and experience of the user.

There is a spectrum of mixed reality, spanning from assisted reality to augmented virtuality. Different hardware devices are used to support the augmentation: HMDs, HHDs and Spatial devices. The choice has been made to use a mobile phone as platform to deploy the augmented reality app. Because of the ease of use, the current use of tablets and mobile phones combined with the expected familiarity with this device and extensive interaction possibilities it is considered a better applicable tool for this application. Although head mounted devices are a good option, they are more tiresome to wear and may provide less intricate interactions compared to a mobile device.

Software is needed to provide the augmentations. Different types of software have been examined, from which Vuforia and Wikitude have presented themselves as most applicable for the realization phase of this project.

3. REQUIREMENTS CAPTION AND IDEATION

The ideation phase is intended to obtain a project idea that is in line with the stakeholders needs and wants and based on the background research. When the technologies, requirements and ideas are combined, new solutions can be created. In order to keep as much creative freedom, no restrictions are imposed on this brainstorm session.

In this chapter the following sub question will be attempted to be answered: *What AR technologies are most suitable for the lemonade factory in order to enhance insight in operational data?*

3.1 STAKEHOLDER IDENTIFICATION

My Automation IT B.V. (MA-IT) is a company that employs over a 100 engineers that focus on providing knowledge in different disciplines: software, mechanical and electrical engineering, office automation, consultancy and product development. To keep providing services, it is of interest that the engineers continually focus on staying up to date with new technologies and techniques. Such a new technology is Augmented reality, where information is displayed digitally in the field of view of the user.

In automated processes there is a lot of data present and needed to run a machine. MA-IT has experience with this kind of data, but wants to make it more comprehensible for the operator. Currently, this is done by displaying information schematically on a display, but augmented reality might be a new technology that does this better. The process in which this can be tested is a fully automatic lemonade factory. The function of augmented reality will be to make the operational data more accessible and insightful.

In this light, the main focus of the 'Augmented reality in the industry' project is to provide a proof-of-concept. The proof-of-concept entails displaying the status of sensors, valves and other components on an augmented reality device.

In order to know who to address and keep informed, it is essential to know who the stakeholders are. In Table 2 the main stakeholders are presented.

	Name	**Expertise and responsibilities**
MA-IT	Koen Sleurink	AR expert and software engineer IA, *thesis mentor MA-IT*
	Marc Waarle	Software engineer and expert on lemonade factory
	Fabian Grootfaam	Recruiter
UT	Job Zwiers	Teacher on interactive media, *thesis mentor UT*
	Dennis Reidsma	Expert on HMI

Table 2. Stakeholders of the AR app

3.2 DESIGNER SCENARIOS

Scenarios

Scenario title
Monitoring the lemonade factory without an AR app

Scenario type
User story

Rationale
This scenario is written to give an insight in the way that operators work, as seen from the perspective of the head operator of a imaginary lemonade factory. The interview that was conducted with him is the basis of this scenario.

PACT analysis
People: Frank, a 23 year old junior operator, educated as mechanical operator, has been working for 2 weeks at the factory.

Activities: Co-monitoring the lemonade factory

Context: imaginary lemonade factory, example provided by M. Waarle, head of automated engineering in MA-IT.

Technology: HMI's, augmented reality

Scenario *without Augmented reality*

Operator Frank arrives at work quarter to 8. After drinking a cup of coffee while the computers and HMI's start up, he goes over the batches that have to be produced today. There are two types of juice that have to be made. There are a few ingredients that have to be manually added: he gives his two assistant-operators the task to make sure this happens. Between the two tasks a cleaning cycle has to take place. The assistant-operators are responsible for making sure there is enough bleach in the holding tanks.

The systems are started up, and the first half an hour Frank watches the values from the HMI's in his office, to make sure everything starts up correctly. He stands up to execute his first inspection round, where he has a routine path and devices to check. Because he is quite inexperienced with the machinery, it takes long for him to find out where he needs to look for sounds and noises, and what the correct values are. His boss gave him a tour in the first week, but after that he was left alone.

He walks around, and tries to listen for deviant sounds in the noisy production hall. He takes out his portable screen with graphs and schematics on it. He looks at an electrical motor, and since it looks a little bit dirty, he suspects it soon has to be maintained by the technicians. He tries to find the engine on the schematic, and after a few misses he finds the correct one by comparing labels for a few times. As the information shows, its maintenance is due in a week. He taps 'replace this part' and sets the date and time as suggested by the ERP.
He continues walking and hears a fluttering sound. There is a series of pipes close together, which all have valves in the same place. He takes out his iPad and tries to identify which valve could be making the sound. Due to the heat of the pipes and them being hard to follow, it is hard to detect which valve is fluttering and where the pipe leads.

He checks the iPad to see what the correct procedure is, and walks over to an area with several valve wheels. The operating images shows which valve to open. Gradually, the fluttering decays and the operating manual says the wheel valve can be shut.

Frank keeps listening for a few more minutes and when nothing changes, he continues his routine check. A few meters further down the route he feels a tank to see whether it is at temperature, but as he is inexperienced, this doesn't tell him much about the status of the process.
Back at his desk, he continues to monitor the process.

With Augmented reality

The problems encountered by Frank can potentially be solved by an augmented reality solution. This solution can provide insight when looking at operational data and comparing that to the real world positioning. He points it at the location or components he wants to inspect and gets a video with digital objects and information on his screen. In the example mentioned above, the fluttering in the pump gives a divergent sensor reading, which is visible in the AR app. With the warning, a guide to fix the problem is given. He gets directed to the valve unit and scans the area to let the app guide him to the correct valve. With turning the valve, the problems is solved and the sensor data gets updated. Via the app, Frank confirms through the sensor information that the problem is solved.

The tanks he encounters later on his route are scanned by Frank too. With doing so, he gets information about the range the tanks currently operate at and how the temperatures changed over the past half an hour. This information is important to keep track of, since the pasteurizing cycles are important to the safety of the consumers.

In this scenario the main problems for the operator are connecting information to the real life components, locating the faulty components and finding components that have to be changed in order to solve the problem. It stands without question that time spent looking for these components produces significant down time, which means losses in revenue. In a small scale lemonade factory this will not be a problem, but as pointed out in [19] when machines grow in complexity and size, the task of finding those components is quite a lot bigger.

Scenario title
AR used for showcasing Beer factory

Scenario type
User story

Rationale
This user story takes place in the Grolsch production plant in Enschede. It showcases how augmented reality can support the presentation of the tour guide and enhance the experience for the viewers.

PACT analysis
People: Pieter, a 35 year old public relations manager, a tour group

Activities: Pieter gives the group a tour through the factory.

Context: The Grolsch Fabriek in Enschede, during a group tour.

Technology: AR glasses, factory machines

Scenario
When doing the tour, Pieter guides the group of people to the main bottling and packaging area. This is a big room with several big machines which perform different operations needed to package the beer. The attendees have received AR glasses, which they wear while listening to the tour. Pieter also has the glasses and can highlight certain machines with his hand gestures. The group can also see this highlighting from their own perspective. Everyone has autonomous control over the augmented 3D model and animation with hand gestures. They can rotate, zoom in or out, put a question mark next to an object (which is visible for the tour guide) etc. Besides the 3D model the information about the object is shared as well. Examples are the batch size, amount of bottles filled per second and type of beer being bottled.

When the tour guide is done with the presentation about the machine, he removes it from everyone’s augmented reality display and continues to walk. When the next machine or item is discussed, the people in the group can interact with it the same way as described above. At the end, Pieter pulls up a birds eye view of the total factory ground, and the viewers can virtually fly and walk around it. When the tour is over, and people can hand in the headsets.

Scenario title
AR used for pathfinding: dislodging stuck cars in an automated sorting and packaging facility

---

Scenario type
User story

Rationale
This scenario is written to display a possible use of augmented reality in the automated industry. Augmented reality can help in time saving in several ways: guiding operators in the right direction, actions and actual data are directly accessible.

PACT analysis
People: Jan, a 38 year old senior operator , educated as electrical engineer

Activities: An operator needs to manually dislodge the stuck cars. Because the workspace is very large and enclosed (safety concerns for automated sorting and handling of packages), the operator needs to be guided to the location as fast as possible to reduce downtime.

Context: a sorting facility runs continuously during peak hours, but jams. The operators see that a package was not properly secured and has fallen in the path of a car that is now jammed.

Technology: AR, Automated sorting system, HMI, Video monitoring on the cars.

Scenario
John works as a senior operator in a big packaging and sorting company. All the sorting is done automatically, but the cars sometimes jam. The warehouse is large, and the automatic sorting system is behind a steel mesh. In order to locate the jam, the operator takes a tablet, starts up the Augmented reality application and confirms he wants to be guided to the jam. The application shows him the route, and guides him which door to go through. This works based on two technologies: beacons and SLAM (simultaneous location and mapping). The device roughly knows where the operator is by communicating with the beacons, and uses SLAM for the exact positioning and augmented reality displaying.
The operator enters the sorting facility through a door, and sees the jam augmented in the distance on the device. He follows the route that the device gives him, fixes the jam, presses a button on the AR device that lets the main operating system know that it can reset the car. When this procedure has happened, the device shows a green “OK” and the route back to the door that the operator entered through. When the door is securely closed, the operator has to confirm it in the application and the sorting machine starts running again.

With the help of AR the jam was easily located and solved, which might have taken longer if the same operator would not have had access to the Augmented reality technology.

3.3 IDEA GENERATION

These scenario's show current and future cases. Although these scenario's are not direct observations, they are taken from either experience from experts at MA-IT or derived from recommendations in the literature. The requirements of augmented reality are traceable in all of them: combining real and virtual, being interactive in real time and projected in 3D.

Training

The young operator could benefit a lot from an augmented reality app. The main problem faced by him is the time it takes to find the problems in the setup. When the installations get increasingly complex it takes more time to learn the system, and sometimes that is even impossible. As suggested by [36] the total efficiency of workers who have augmented reality at their side can increase up to 50%. This means half of the time the operator can perform other tasks. A junior operator would benefit in this case from a handheld augmented reality solution. The training can either be done on the handheld device, such as gaining information about an object or finding information about it in the ERP, but the tablet can also perform visual recognition and display real time process information. This will benefit the operator, as it turns the time the operator was struggling to find the right part into time spent learning about the part.

There are two aspects to the training: firstly to search for and localize important parts of the installation, and secondly applying the correct information to those parts.

Information overlay

An augmented reality program could function as a way to transfer information to a user. By using the augmented reality application, digital information can be connected directly to the object it concerns. This eliminates the need for a second display with information. The process of mentally connecting this information to the object, is mentally taxing. Augmented reality can take over this task. Derived from the scenario, all kinds of information can be displayed: a whole machine highlighted or lifted out of the factory hall, a part in the machine that gets highlighted or information about the machine, such as dimensions or operational data such as production volumes.

Pathfinder

As described in the last scenario, AR can be used as a pathfinder. In large installations or factories it can be hard to find the fastest way to complications and searching for a jam takes valuable time. AR can get rid of the unnecessary waste of time by calculating the fastest route, so the navigation based on memory can be omitted. A technology that is currently being researched at MA-IT is a system that communicates position with use of beacons, which could be implemented in this idea as a method to do localization.

3.4 AUGMENTED REALITY AS AN INNOVATION

Usually, augmented reality is considered a nuisance, not a necessity. However, as seen in the literature research, the technology is maturing. In this light, it is important to discuss why augmented reality is a step forward in this use case, instead of a nice technological feat which is not on the level of usable technology. In this paragraph the reasoning behind choosing augmented reality as a solution to this problem will be expanded.

Description of the situation

MA-IT has an experience center, where companies (including but not limited to IFM, Mitsubishi and ABB robots) have placed their devices to showcase their abilities and provide engineers with the opportunity to gain hands-on experience. As an example, students of the university of Applied sciences Windesheim have made a small scale but fully functioning lemonade factory, visible in Figure 16all to the specifications of the NWVA (HACCP) [1].

There are several steps the machine has to go through before lemonade can be dispensed. Tap water enters the factory on the left of Figure 16. The water is then ducted to the main mixing tank on the right, where lemonade is added. By means of circulation through several pipes and the mixing tank, the lemonade is mixed. When ready, a valve on the very right opens periodically to let out portions of lemonade.

Figure 16. the lemonade factory

After a batch of lemonade is made, the tanks and pipes have to be cleaned. In order to do this, one of the smaller tanks on the left is filled with water and heated, and the other is filled with a chlorine mixture. The hot water is pumped trough first, in order to pre-heat the pipes and the main mixing tank, and following this the chlorine mixture is pumped through. After all the tanks and ducts are cleaned, the waste water is discarded. Because this is a fully automated process, there are sensors to gather data about temperature and flowrates and actuators such as valves and pumps; they are connected to and operated by the main operating system (Aprol by B&R [2]). Some of this data is relevant for operators or technicians to monitor the lemonade factory. To visualize this operational data, a human-machine interface (HMI) is used. On this display – visible in figure 2 – information is shown in a schematic context of the lemonade factory [3].

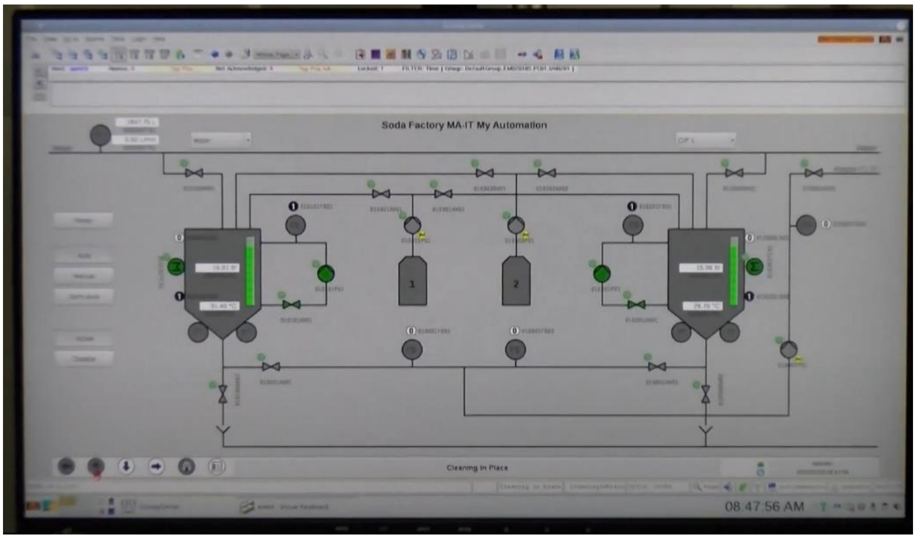


Figure 18. Screenshot of the HMI of the lemonade factory

This HMI is well accessible to the operator, due to the relatively small size of the production plant; one never has to walk a long way to the screen. But when this factory would be scaled up, all of these advantages fade the larger it becomes. Therefore it would be a great step towards operational insight if there is a way for an operator to access information for the part he wants to investigate. This can be done with a portable solution, such as a tablet. However, with a bigger machine and more components it can be hard to determine which schematic representation correlates to its real life component and where exactly it is located – this is especially the case with novice operators.

One way to tackle this problem to make use of augmented reality. Augmented reality is a technology that ‘augments’ the real world with digital objects, e.g. a textbox that appears to be anchored on an object in the real world, e.g. in Figure 17 [4]. When the augmented reality device is moved, the textbox stays connected to the object and subsequently moves around on the augmented reality screen and can potentially move out of sight. When the real world object comes back into view, so does the textbox that is connected with it.

Figure 17. Example of how textboxes can be anchored to a real life object

This technology can therefore be implemented as a means of HMI: the sensor data can be put into context and displayed as relevant information when it is connected to a certain object and shown on an augmented reality device. This makes the information directly relevant, and eliminates the need to walk all the way back to a stationary HMI or to compare the schematics on the HMI with the real life situation. The recent improvements in AR technology, such as advancements in hardware (capable mobile devices) and improved software have made it clear to MA-IT that AR can play an interesting role in this use case and in the general industry. The company has interest to explore the use of augmented reality to enhance the insight in operational information for operators.

The small amount of information displayed on the HMI is only the tip of the iceberg of the total amount of data gathered. All this data is not directly relevant information for an operator, but can give indications about the health of components. As a means of gathering this data, IFM has provided a device called Edgecontroller. The Edgecontroller is a HMI, but more importantly, it connects the lemonade factory to a cloud service. The data generated by the factory can be uploaded and accessed remotely. Then, a mobile application can be created that accesses data from the cloud, and the requested data can be put into context: the AR environment.

3.5 PRODUCT IDEA

To detect an element of the factory, the ideas went from detect 'a sensor', to detect so called Equipment Modules. In automation this means a group of components, including but not limited to sensors, actuators, tanks and pipes, which together perform a distinct operation. In this case, there are two mixing tanks with several sensors and a pump. In Figure 19 an equipment module is visualized, encased in a red quadrant (this is just for clarification purposes, it is not an element in the design). The sensors provide all kinds of data: storage levels in the tank, flow measurements such as speed, volume, direction and temperature

Each equipment module had at least one option to toggle information, in the form of a colored dot. This colored dot is a result from interviews with the stakeholders, where the need for an option to see whether something can be toggled was expressed. This dot will show up the moment the augmented reality software detects and recognizes an equipment module. When this dot is pressed, information is displayed in a box, which is connected to the location in the real world. This means that when the device is moved, the box moves respective of the screen, but not to the object in the real world and as such appears to be linked to it. These boxes are exemplified in Figure 21 and Figure 20 . The amount of information points that are displayed is subject to the amount of sensors in an equipment module.

Similar to the box connected to an equipment module, the client expressed wishes to display the equipment module in real space as a 3D overlay. It had to stay connected to the equipment module similar to the textbox. In Figure 21 an example is given how a duct can be highlighted. It also had to interact with the other machinery. Besides the toggles, textboxes and 3D overlays, the client expressed the wish for displaying warnings. Two classes of warnings were discerned based on urgency: warnings where machine operations are not directly impaired, and warnings where machine operations are immediately terminated.

The first set of warnings can be the result of data mining. This requires a place to store machine data which can be mined. This machine data should be ingested via a device called Edgecontroller. This device takes all the datapoints in that the sensors generate and upload it to a cloud service. This cloud service should then in turn be the storage of this data. The application should communicate with the database to read data. The app should take care of placing the data in the correct context, in order to make it meaningful information.

A useful feature could be the display of the historical data, so an operator could see the information on a specific part of the machine. This could be anything from temperature cycles, flowrates or motor loads. Another feature that is enabled by storing historic data could be the prediction of finishing time and batch sizes. This is not directly important for operators, but can be useful for planners and managers to manage precisely when new ingredients have to be manually added or other operations have to be executed.

When the features presented above were covered, there was a potential to add a synchronous communication, where commands can be send from the application that actuate mechanisms in the machine. This can be useful for an operator, because it would provide the option to change something in the machine and give it a command to run the change.

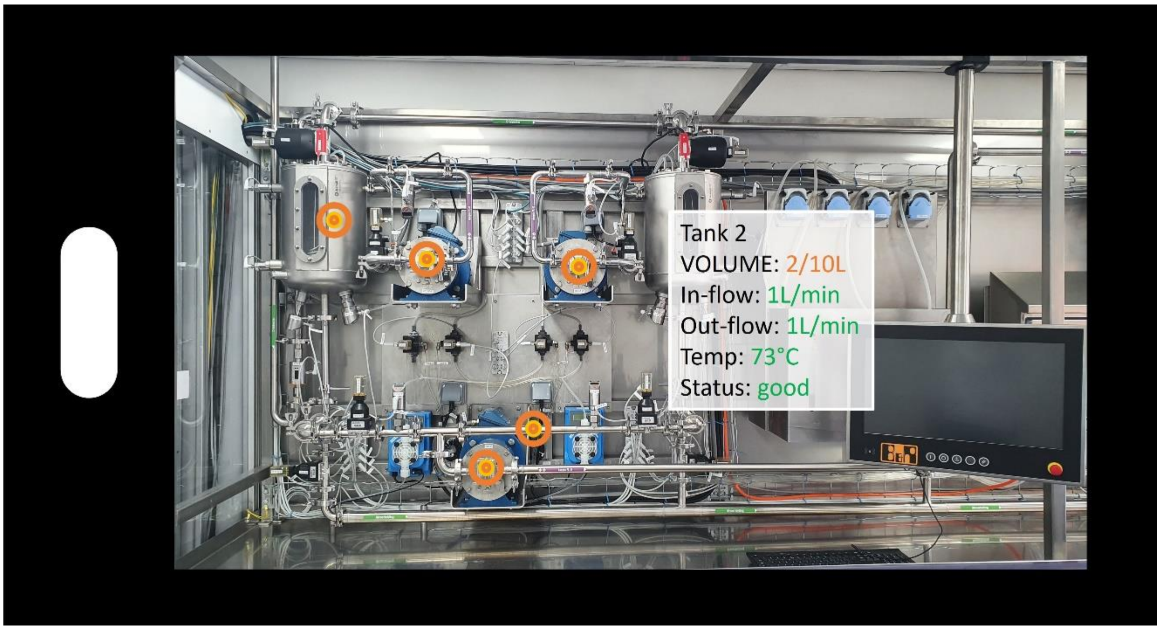


Figure 20. Pressed the top right pointer

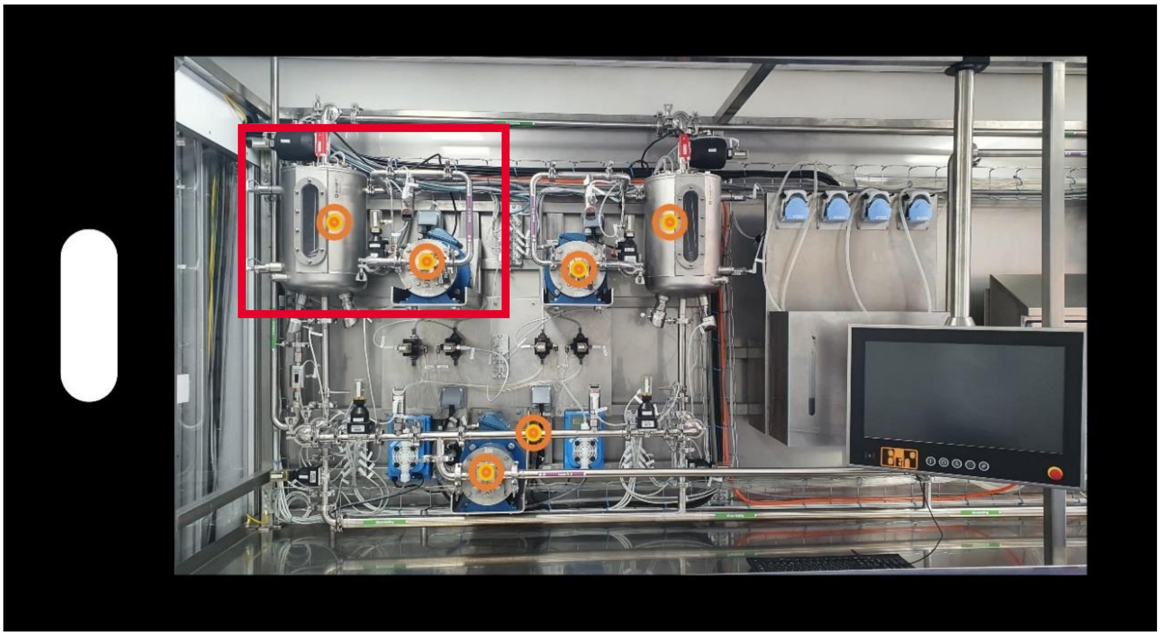


Figure 19. Initial overview

In the presentation of the initial ideas and designer scenarios MA-IT experts have pointed out the need for an intuitive user interface. They pleaded for less text and more visual cues, such as use of colors and buttons with icons. This is further expanded on in chapter 4.

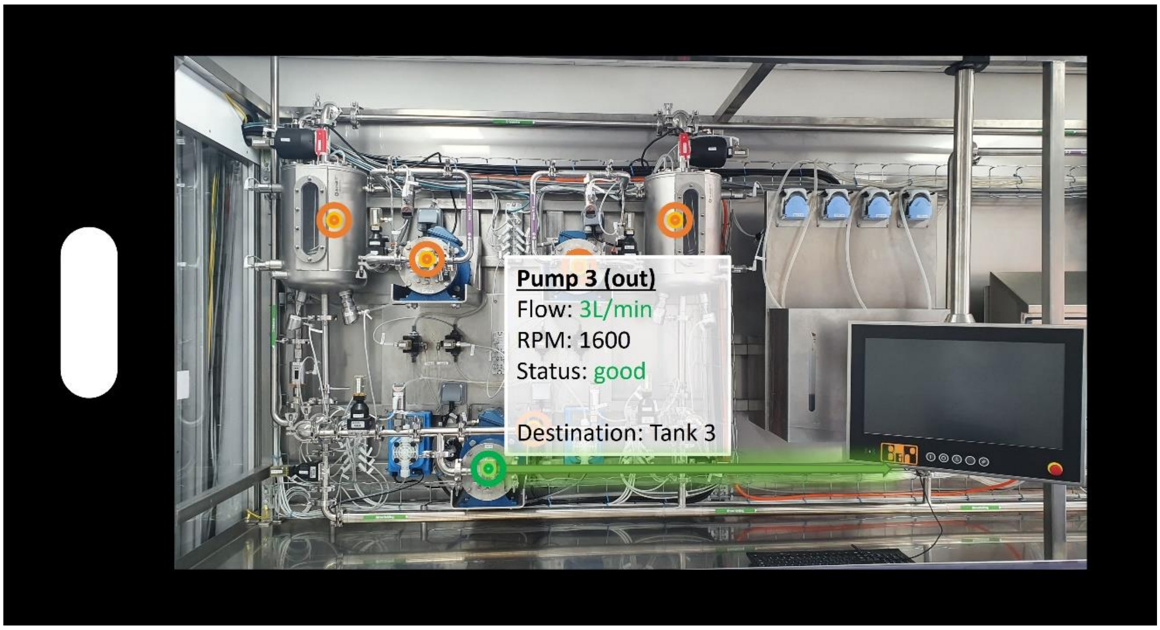


Figure 21. Pressed on the bottom pointer with highlighted duct

3.6 SPECIFICATION

To structure the wishes of the client, a MoSCoW prioritization [41] is used to prioritize what needs to be included in the design of the AR application.

Derived from user (expert) interviews, related work and feedback on sketches, mock ups and prototypes, functionalities are assigned to one of four categories. These categories are: ***M**ust have, **S**hould have, **C**ould have* and ***W**on't have this time*. A distinction is made in the categories, namely UI and Logic. This distinction is important to make clear where the focus is placed on.

Must have	
UI	• Recognize machine components • augmented reality displaying (textual, pointers, 3D overlays, warnings etc.) • visible sensor status • visible machine warnings • icons for navigation
Logic	• augmented reality tracking and processing • communication with cloud service (which is connected to the lemonade factory with the edge Edgecontroller) • user profile (technician) with user specific access to database information

Should have	
UI	• option to toggle visualizations (such as graphs) • visualization on the pipes (to see where pipes go)
Logic	

Could have

UI	• Visualize batch information (expected time of duration and time of completion) • Visualize availability of cleaning fluids (how much has been used, how many uses left?) • Selection of user profiles
Logic	• Multiple user profiles with different degrees of access (to information)

Won't have this time	
UI	• Hyperlink to replacement product • Installation/ repair guide • Manually actuate machine components
Back end	• Predictive maintenance algorithm (e.g. if motor has ran for 1000 hours, perform maintenance - not in app, should be server side) • Synchronous communication

Tabel 1. MoSCoW prioritization

4. REALIZATION

In this section the realization of the final prototype is documented. Firstly, the methods and techniques used to come to the final prototype are discussed. Secondly, a user analysis and system decomposition are given to elucidate the interactions with the application. Finally, iterations and design choices are explained.

4.1 METHODS AND TECHNIQUES

The realization of this prototype is done in an iterative way, see Figure 23. This method has a few advantages over a so called ‘waterfall’ model, as visible in Figure 22. One of the advantages of an iterative process is the smaller steps that can be taken in the design process, putting less pressure on the final prototype to be perfect on the first try. Secondly, it helps to keep in contact with the stakeholders. When plenty of smaller changes are made during the design process, the stakeholders are included more and can be made aware at an early stage of potential problems. With a waterfall model, this is only made known at a stage where it might be costly to change the design (and all components that rely on that part).

In several iteration sessions, new designs were reviewed, discussed and chosen together with client representatives from MA-IT. In these sessions key components such as database designs, User Interface designs and augmentations were discussed. This method is heavily inspired by Jesse Schell’s *‘The Art of Game Design’.*

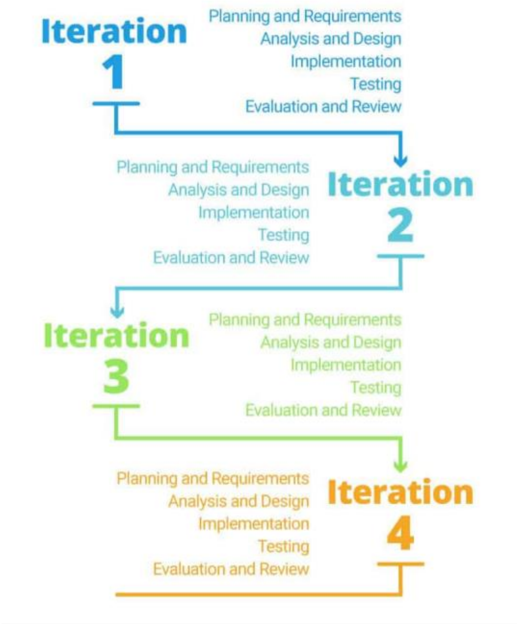


Figure 23. Iterative model

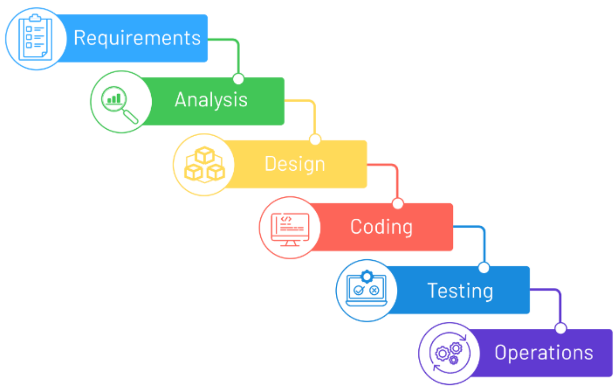


Figure 22. Waterfall model

Schell has introduced methods for game design, which can be applicable in many kinds of product development. In the process of designing the AR application, Schell’s “Eight tips for productive prototyping” were used [42]. To exemplify, in Chapter 3.3 (Idea generation) several quick designs of a potential app were made. This was done for several reasons. Firstly, it gave the opportunity to check whether or not the ideas MA-IT had with the AR app had come across as intended. Secondly, it provided a simple sketch that the client could make suggestions on. There might be cases where the client had envisioned something in his head, but when the idea is visualized unforeseen problems become visible. Therefore, this can be seen as the ground design, the ‘zeroth’ iteration.

It is imperative to keep these early sketches as simple as possible. Otherwise the pitfall of withholding critique due to fear of hurting the designers feelings will open. This occurs when a lot of effort has been put in these initial concept sketches. It might feel harsh to critique or question the designs. By the critical observer it can even be argued that the early sketches are too detailed already. The reason these sketches have been made in this level of detail are because there have been paper prototypes and ample discussion with the client before this design.

Similarly, there have been discussions about the backend part of the system. A paper prototype of the systems functions has been constructed. Discussions have been had on the database to be used, and how to best ingest and sort data in the database tables to improve readability, simplify querying and creating a structure.

A really important part of this method is the way prototypes are seen. A design should never be considered a final product. In the beginning a few questions have been asked that need to be answered. To quote Schell: “[…] whether you like it or not, the first version of your system is not going to be a finished product, but really a prototype that you will need to discard before you build the system the ‘right’ way.”(p. 87). With the prototyping this principle was used in the early stages, to get a starting point. It is easier to say that something turns out to be not good, rather than to design it perfectly on the first try.

4.2 SYSTEM ARCHITECTURE

User interaction

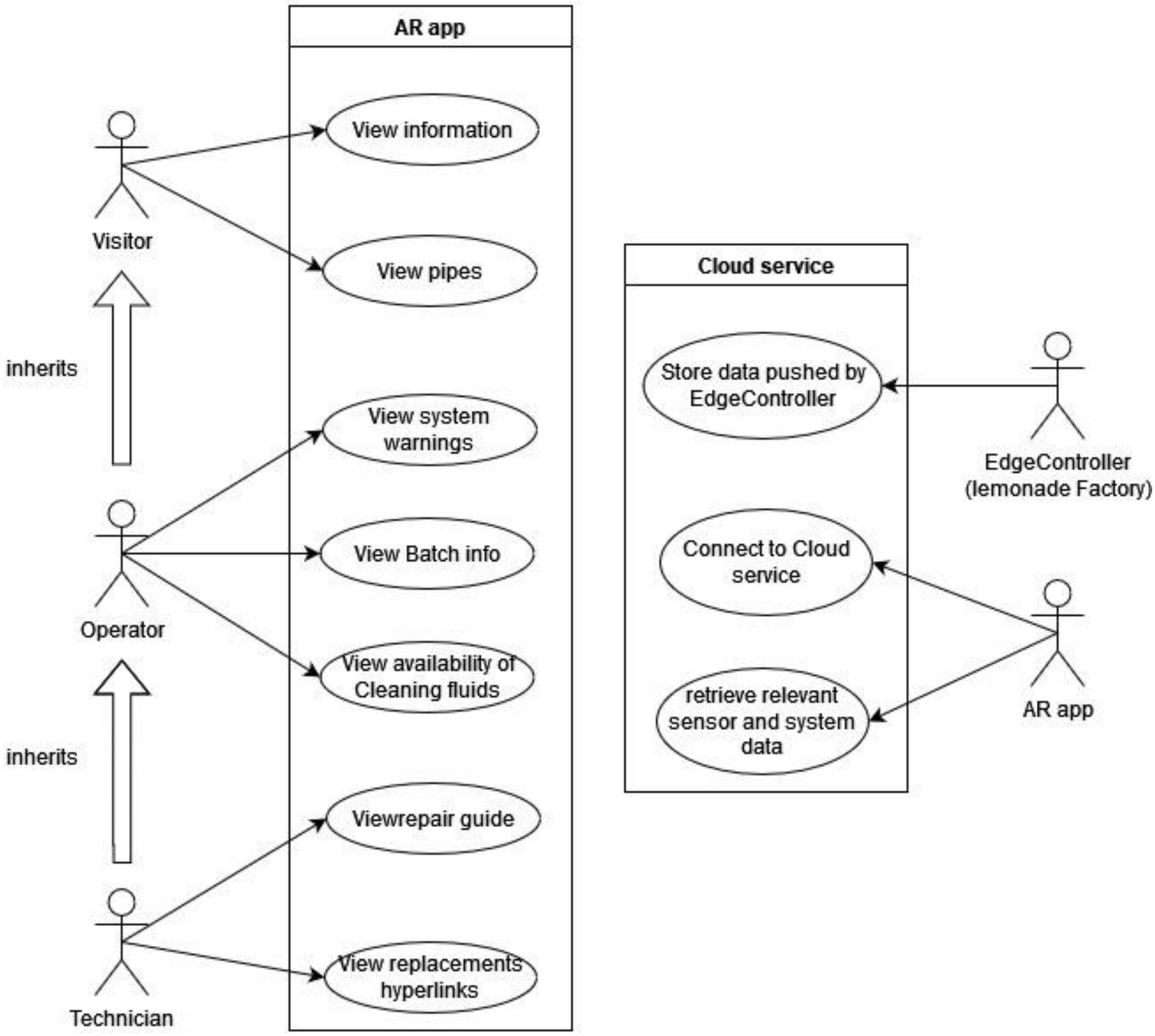


Figure 24. Use case diagram

The use case of the augmented reality app is visible in Figure 24. The use case is set up in such a way that the actors 'act' on an application. The stick figures denote an actor on the application. The operator inherits the functionalities of the visitor, but it doesn't have access to the information a technician has access to. On the right, the AR app is considered an actor on the cloud service, since the communication is initiated by the app.

System decomposition

A way to decompose a system is to display system components as black boxes. This is often done in a top down way, meaning that the black boxes become increasingly detailed. The boxes can have inputs and outputs. These are visualized by arrows.

Overview - System (inputs of both ends)

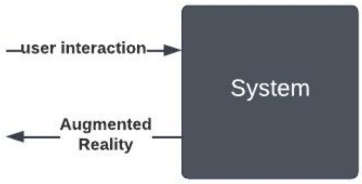


Figure 25. System overview

The system is a black box that only shows its in and outputs. In Figure 25 the inputs and outputs are shown. User inputs are entered into the system box, and Augmented reality is returned to the users. This diagram shows that the system has been attempted to be made easy to use for the user.

System

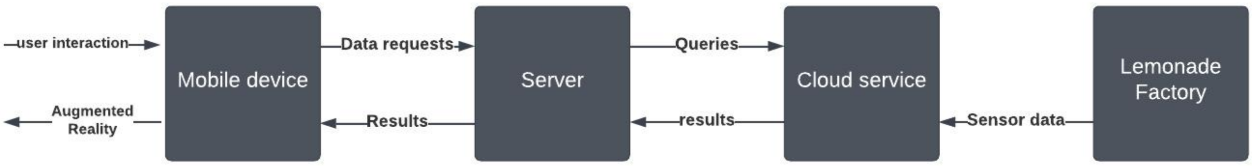


Figure 26. System

The system can be decomposed in four main parts. In Figure 26 these parts are visible. The user is on the left of the system (as seen in Figure 25). The mobile device connects to an intermediary server. In this prototype it is a server running on a laptop, which makes the requests to the cloud service. The cloud server replies with the query results, which are then relayed back to the mobile device. The data that is present in the Cloud service comes from the lemonade factory, pushed to the cloud with the IFM Edgecontroller.

The connection to the database is also carefully chosen. Usually, it is not recommended to directly connect to a database, but to an intermediary server. This is for several reasons, from which security is the first and arguably most important one. With specialized tools hackers can decompile the app and retrieve login information from it. When direct access is available, a user can potentially perform harmful actions on the data in the database. Besides the security risk, there is a problem with app maintenance. When the database is updated, all the apps have to be manually updated to be able to interact with the database. With an intermediary service only one application has to be updated: that of the service. There is also the possibility to push data directly to a mobile device. This, however, doesn’t create the flexibility that a centralized database has: data is be accessible on multiple locations and it can be data mined. Moreover, the security risks that this brings is also substantial. With a direct connection, the lemonade factory has to be made secure for every device that is connected. With an intermediary device, there is a dedicated security and request handler. The last disadvantage of a direct connection is that a direct connection would put a lot more strain on the application, since the data acquisition and handling is done in the application.

Mobile Device

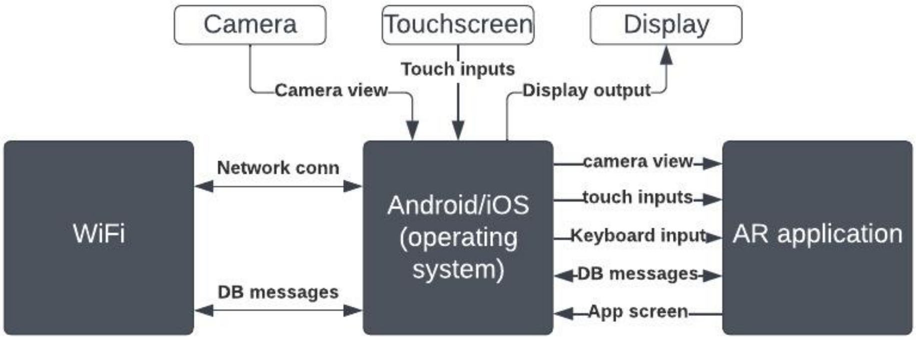


Figure 27. Mobile device

In Figure 27 the used systems on the mobile device are displayed. Except for the Augmented reality application there are no components specifically made for the application. The augmented reality Application relies on the operating system of the mobile device. Firstly it makes use of the camera. The camera feed enables the visual detection algorithms. The feed is then augmented and sent to the display. Secondly, the touchscreen provides inputs that the app uses to navigate. These consist of virtual keyboard inputs, Call to action interactions and virtual button presses. The augmented reality app uses the networking capabilities of the mobile device too; in this case Wi-Fi is used. A network connection is established and database messages are transmitted: both requests and results are transmitted via the Wi-Fi connection.

Augmented reality app (user interface, app core, networking, AR tracking, data requests)

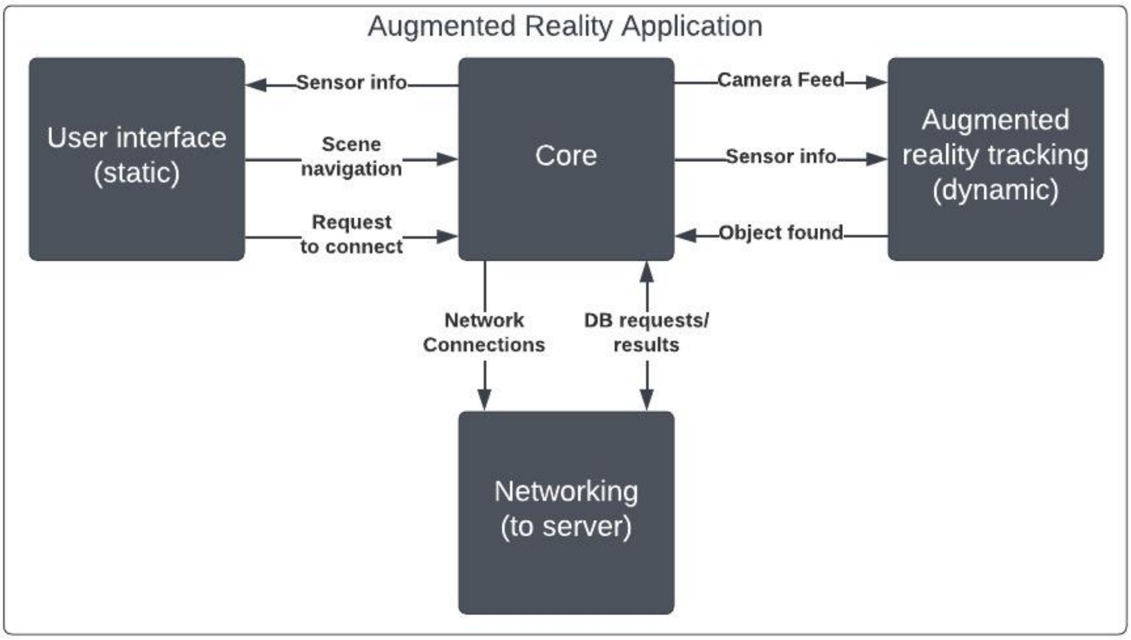


Figure 28. Augmented reality application

The augmented reality app is based around the core (the part that is connected to the operating system of the mobile device). It sends and receives information to and from three different subsystems. The dynamic augmented reality tracking system receives the camera feed and sensor information for dynamic displaying of the sensor info. Tests have been conducted to see whether data displayed in 3D is beneficial or should rather be in the static UI. The AR subsystem also returns whether an object is found. This prompt is then used to perform a database Query through the established network connection. At last, the returned data is then visualized in either the 3D space or the static GUI. The GUI is also responsible for scene navigation and handling inputs from the user, including but not limited to requesting to make a connection to the server.

AR tracking (show the recognized EMs (Equipment Modules))

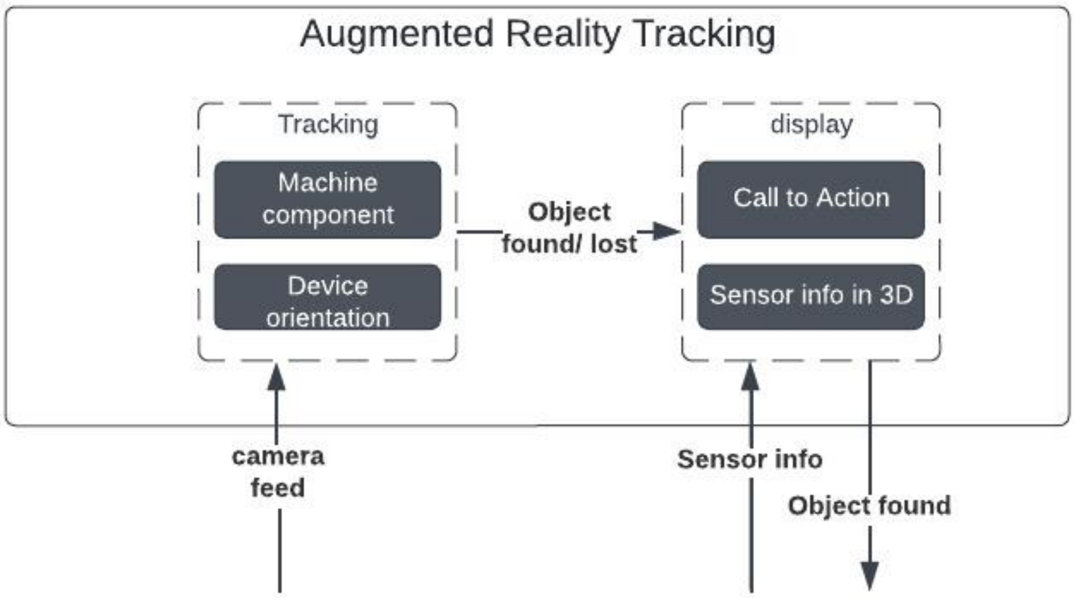


Figure 29. Augmented reality Tracking

Figure 29 shows that the Augmented reality system consists of two parts. Firstly the detection of objects, called tracking, and secondly displaying of objects. The tracking is done in two ways: the Vuforia library makes use of the orientation of the mobile device to help track an object. Secondly, it deploys computer vision algorithms to detect objects. Combined, computer vision and device orientation tracking create the possibility to track objects.

When an object is found or lost, this information is relayed to an algorithm that decides what to display. There are multiple possibilities, of which the first is to display a CTA. When this is displayed it can be pressed, and the AR tracking box will output *which* object is found. In another subsystem information is retrieved, and send back to the AR tracking system. Here it is connected to the found object, in 3D space.

User interface (Static)

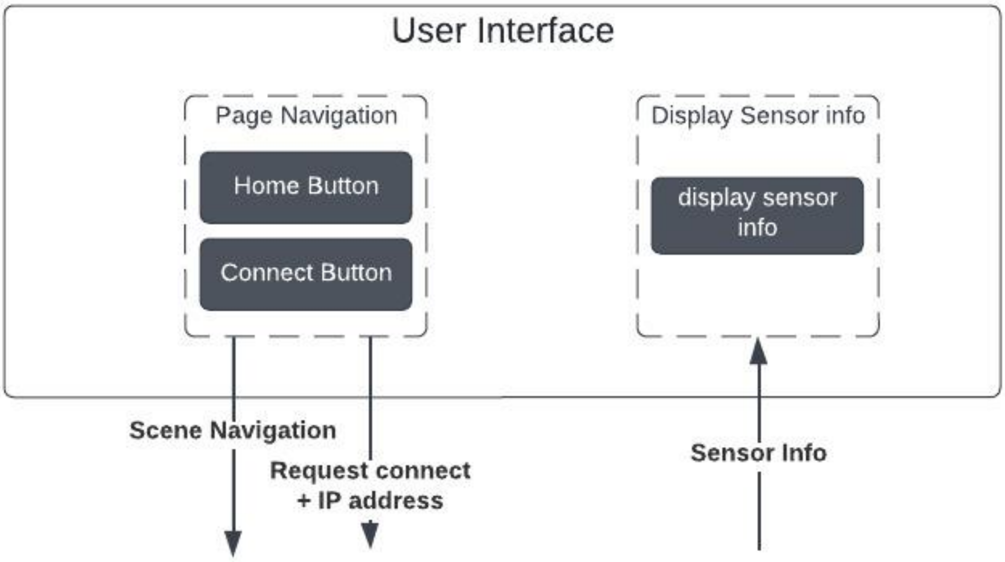


Figure 30. Static User Interface

On every page (or scene, in Unity) of the application there is a button that is linked to page navigation. The choice has been made to simplify the interaction that is possible with the UI. In the early development stages there was a special page for login, but on the home screen there was a possibility to go directly to the Augmented reality part of the application. This would cause problems since there was not a database connection established. In the newest prototype versions this problem is eliminated: from the login screen a user is only able to get to the AR functionality after it has connected to the database.

Data handling

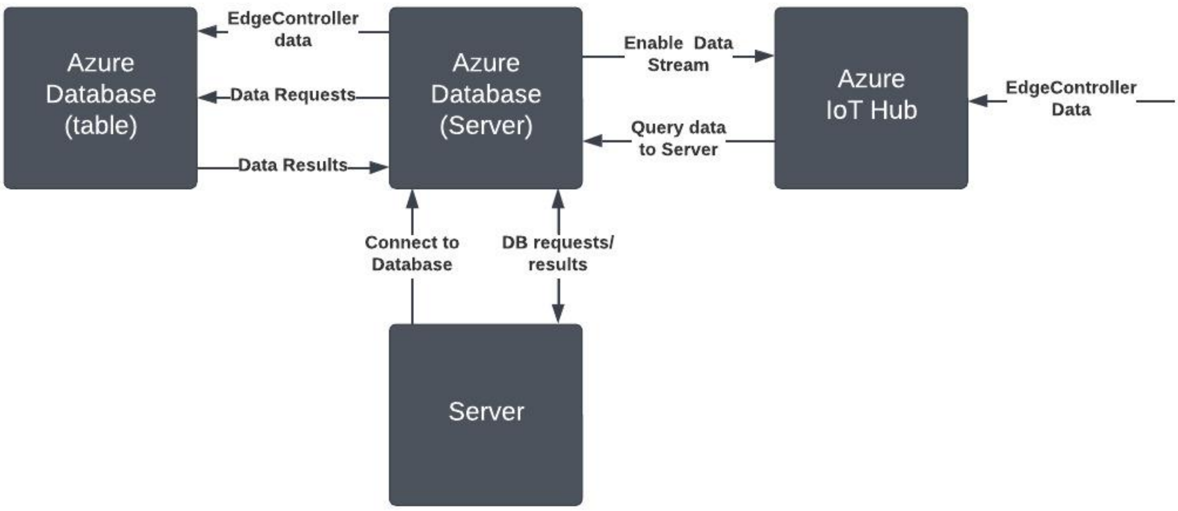


Figure 31. Data handling

During an interview with the engineers of IFM, the company that makes the Edgecontroller, it came to our attention that Google Cloud, AWS and Cumulocity IoT hubs require a user to sign up with a credit or debit card which wasn't available at the time. Azure however has a special discount for students. For this project it was not a big dealbreaker to figure out which cloud service to use, but it was pivotal for using the Edgecontroller. The capabilities that this device opens up towards augmented reality is more important than which cloud service is being used for the proof of concept.

Inside of Azure there are a few applications that need to be ran to acquire data from the Edgecontroller. Data ingestion is handled by an application called Azure IoT Hub. In the free student version it will take up to 8000 data files (the Edgecontroller sends data points of all selected sensors in one file once every second – or custom update frequency). In paid versions the amount of messages can be much greater. Due to the limitation of 8000 messages it is necessary to keep data stored somewhere to perform actions on the data. In order to host this database, a server needs to be setup first. In this server a user can open up multiple databases – all in in Azure Cloud service.

A smooth integration between the IoT Hub and the database is already suggested by Azure: an application called Stream Analytics Job. Simply put, it takes the data from an input (IoT Hub), performs an action or filtering, and forwards it into an output (Database). It is very quick to set up, and can run indefinitely.

4.3 ITERATIONS

As mentioned in the introduction of this chapter, the realization process was executed following Schell's method for game design. This iterative way of working works towards a prototype that can be fully tested and held against the lights of the functional and non-functional requirements and the scenario's from 3.2 (Designer Scenario's). In this paragraph several iterations building towards the final prototype are described.

Software used

The software solution that is used to be able to perform all these ideas is Vuforia, in the shape of a Unity library. Unity is a 3D game development engine, which enables scripting in C#. These scripts perform actions that are required in the application. It includes handling user inputs, application behavior, displaying of elements in the app and other logic. Moreover, Unity enables the application to be exported to different operating platforms, including Android, iOS, Windows and others. This makes Unity the perfect platform for this prototype, as the application could be extensively tested and manipulated if needed be.

Methods of object recognition

The first iteration that was done was choosing a method of object recognition. Based on the research literature that was found on existing augmented reality technologies, Vuforia has 4 different methods of object recognition: image targets, cylinder targets, model targets and 3D scanning. Several trials have been done, such as the use of a 3D CAD model, as seen in Figure 32.The problem with this method is that the provided 3D model is not entirely accurate, which resulted in no recognitions.

Secondly, a 3D area scan to compare to an existing scan was investigated, but this required a specific sensor which only certain mobile devices have: a Lidar sensor. This sensor creates a map, in which distance to points is measured. The tests that were done with this technology proved that it made the application substantially bigger, without an expected increase in recognitions. For the short time that the technology could be tested, it proofed not to provide the expected results and therefore was not eligible for the augmented reality application. The suspected problem is that the objects that needed to be recognized were too small to be detected.

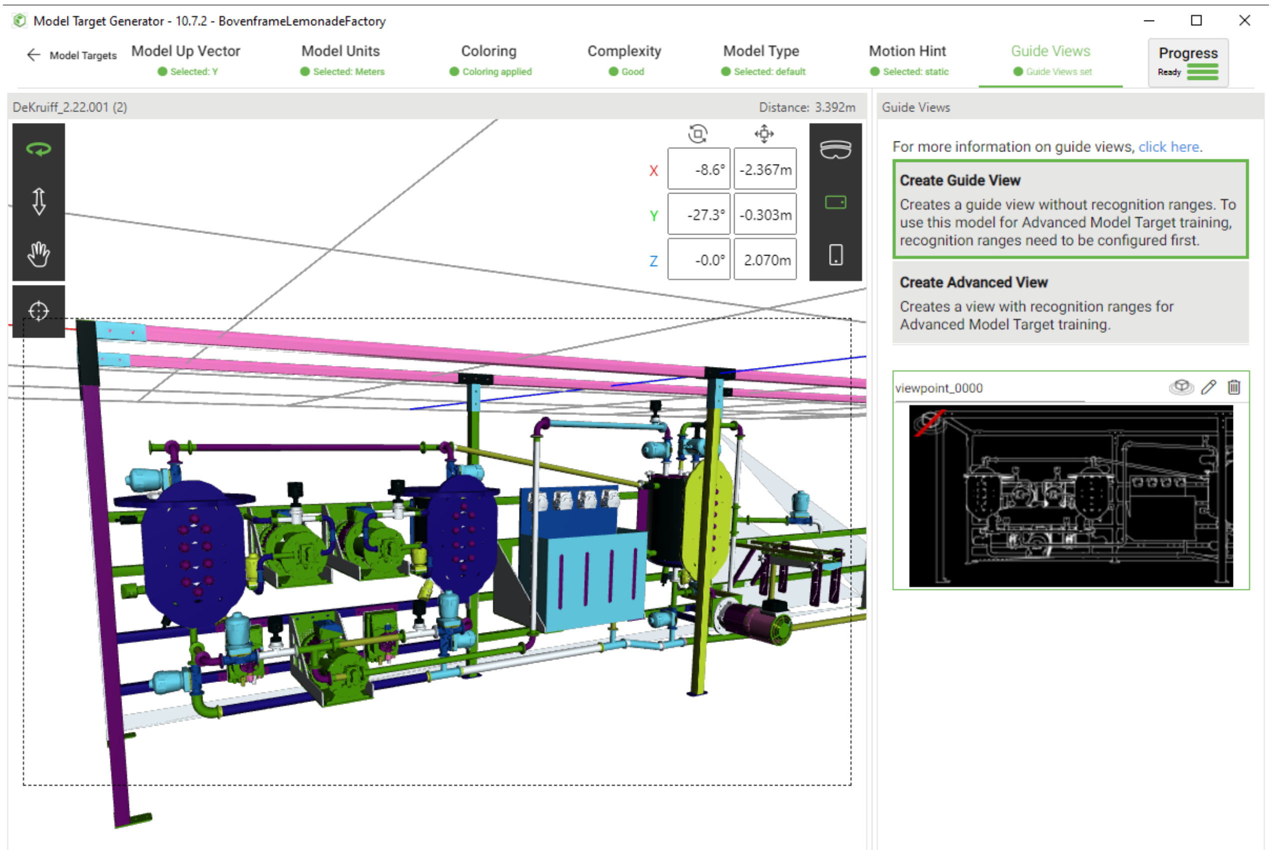


Figure 32. 3D object of the lemonade factory in a Model target generator (Vuforia)

The cylinder target is not applicable because the shape of the lemonade factory is too varied. Moreover, the algorithm tries to bend an image on a rounded plane, which is not needed. After testing, an image target was tested. This option is also not without caveats however, as the lemonade factory is in 3D. If the position of the phone changes, the perspective changes too, and as such changes the image. This phenomenon, called parallax, is exemplified in Figure 33. Parallax effectFigure 33. Taking these points into consideration, the image was still the most viable option and was therefore chosen to continue the work. In the recommendations these choices will be revisited, because during the realization phase new insights have been gained in how these different algorithms could be employed in a more suitable way. Figure 34 is an example how an image target works. An image is scanned and an augmentation is presented to the user of the application. This augmentation is fully customizable and subject to the choices of the designers and engineers.

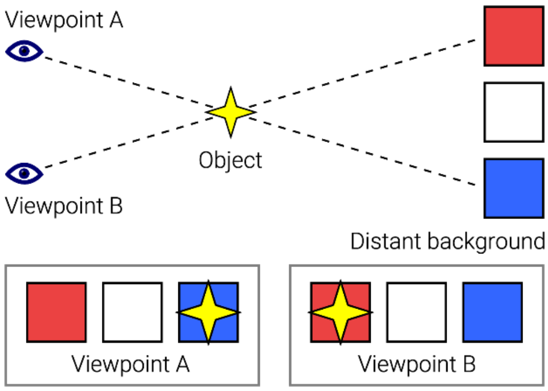


Figure 34. Parallax effect

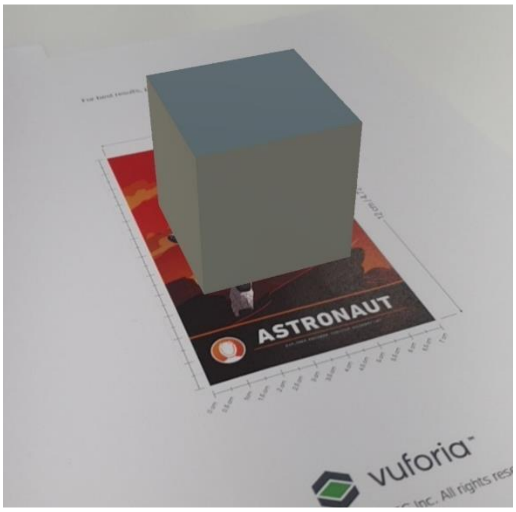


Figure 33. A cube augmented on an image target

Cloud service

The choice of cloud service has been subject to development as well. In an interview with a cloud specialist in MA-IT and several phone calls with the contact person at B&R, a discussion was had about the best way to implement the Edgecontroller and connect it to a suitable cloud service. An advice that was given was to make a custom database server, but due to impracticality and extra work that this solution would bring, this advice was not the most practical. In the following discussion the best possible implementation would be provided by Microsoft Azure cloud services. They have a special deal for students, which enabled me to make use of their services to a certain amount of messages routed and get work experience with cloud services in general. In Figure 36 the cloud service portal is displayed. Under 'Resources' a test group and four applications that are employed as the back-end of the augmented reality application are shown. In Appendix 1 : Code the connection between the app and the *unitytestdatabase* (SQL-server) can be found.

An intermediary storage, such as a database, offers the opportunity to perform big data analysis. With cloud based apps several interesting information points can be retrieved, such as predictive maintenance, planning and general performance measures. Secondly, historic data can be viewed with a data storage. When data is routed directly to the device, the only history that device can display is the data that it has received while requesting from the database. A direct connection to an

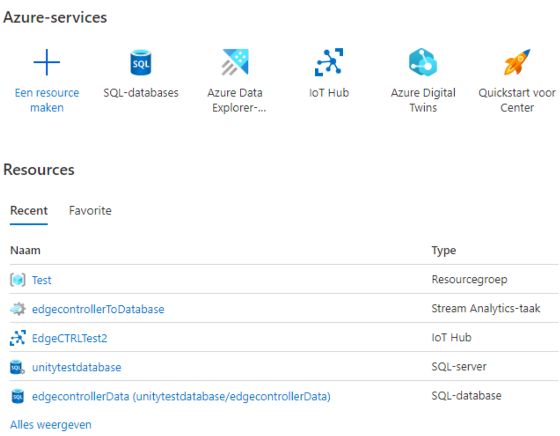


Figure 36. Azure cloud service portal

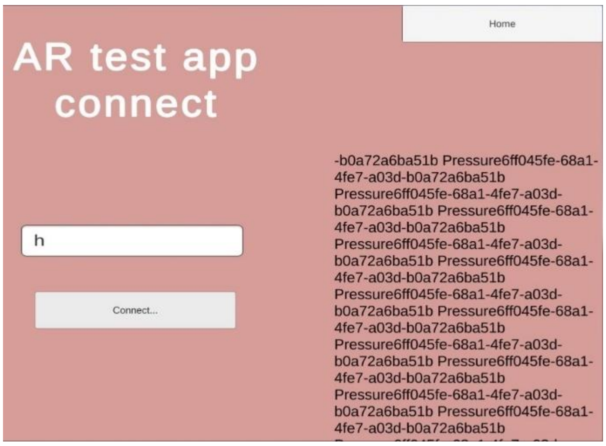


Figure 35. Application request field and reply from Database

app doesn’t provide these options. Finally, a direct connection puts a higher strain on the application side, which is highly unfavorable. A cloud database therefore is chosen.

The connection between the app and the database server was subject to several iteration as well. The first stage was a simple request and reply communication, visible in Figure 35. Any input would result in a similar reply from the database. When the communication was established, the communications part was moved to the augmented reality part of the app, and the first was changed into the connect page.

In Figure 39, Figure 38 and Figure 37 three iterations of the augmentations are visible. The main changes are how data is retrieved and augmentations are visualized. When an iteration works, it is implemented into the application. In Figure 40 a warning is visible, to signify anything in that Equipment Module is faulty. This can be a sensor measuring wrong values for example. When this occurs, instead of a normal CTA a warning is shown. The warning is accessible like the other CTA’s.

Figure 39. Test with AR and Static text

Figure 38. Functional testing – Data representation

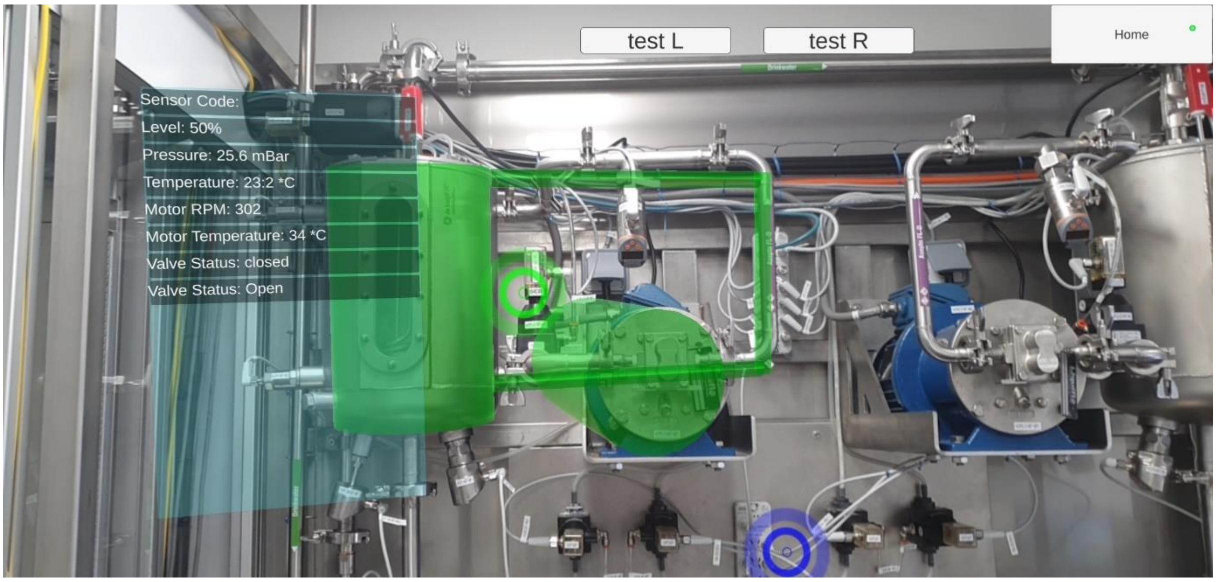


Figure 37. functional testing - AR, data, 3D visualisation

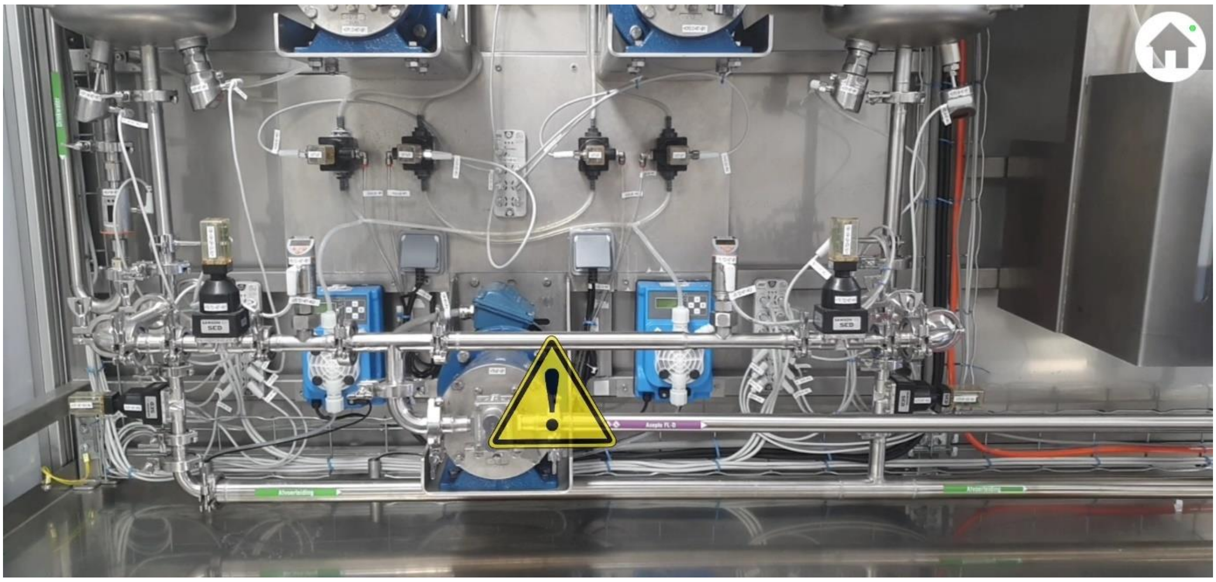


Figure 40. Warning CTA

User interface iteration 2

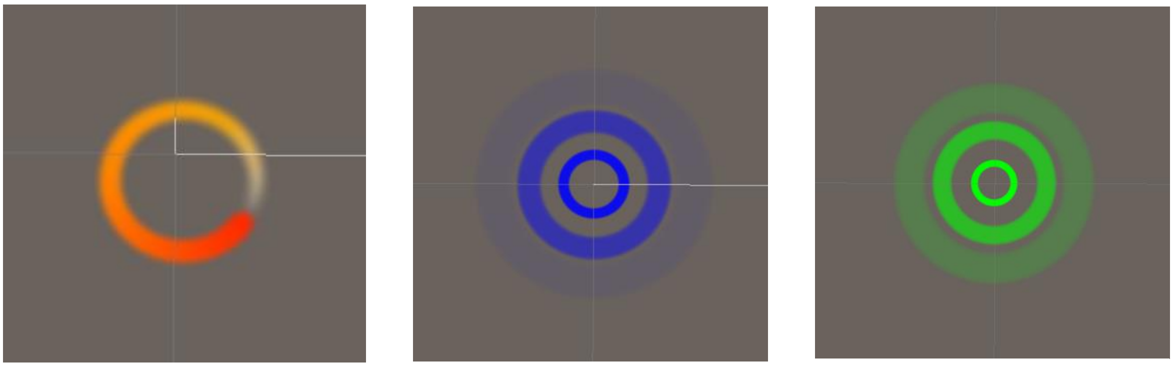
Figure 42. Login screen

Figure 41. AR screen

In the second user interface iteration, buttons were changed from text to icons. Operators can be of any nationality and machines are found all over the world. Therefore, the HMI's should be set up in such a way that it is easily understood by most. The use of internationally understood icons is a great way to avoid complex language and translation systems. In Figure 42 and Figure 41 the used icons are visible. On the login screen an exit pictogram is used, to signify that the app will be closed when pressed. On the AR page, only a Home button is present, to minimize information overload. Moreover, in the current state of the app, there is no need to provide any more navigation.

Call to action visualizations

The first version of the CTA (call to action) for when a subsystem is found is visible in Figure 43(a). Its original purpose was to be very noticeable. The color scheme did exactly that, but the animation gave it the appearance that something was loading. This became apparent during user tests and when discussed, the testers provided valuable feedback in the form of examples which they found to be inviting to press. The color was found to be misleading too. Red and orange are (subconsciously) perceived to point out danger, which was not the intention for the CTA's in this stage, as there were future possibilities of using distinct warning and danger signs. With this feedback in mind, another iteration was made, resulting in the CTA visible in Figure 43(b). In order to visualize that a certain Equipment module is active, its CTA gets a different color. In this iteration visible in the color green was chosen to stay close to the blue color theme while not having (negative) visual connotations, this iteration is shown in Figure 43(c).



a b c

Figure 43. Iterations of CTA's

Information display

One of the most important factors of the application is the visualization of information. Currently, there are two options of providing this information. Firstly, there is a panel in 3D space that is fixed in the same orientation as the lemonade factory. In user tests, this option, visible in Figure 46, was favored over a static panel, as seen in Figure 45. The reason given for this preference can be summarized in (one of) the principles of proximity: “[it] helps users understand and organize information faster and more efficiently” [43]. The second option of visualization is a static panel. Especially on smaller screens this form of displaying information has proven in user tests to be more accessible. However, in this case the proximity principle is traded for clarity. When the screen is moved, the panel with information moves irrespective of the background. This results in a greater readability which was favored over the proximity. In order to make clear where the information is connected to, a 3D model of the lemonade factory is displayed when an Equipment module is made active. This is visible in Figure 44.

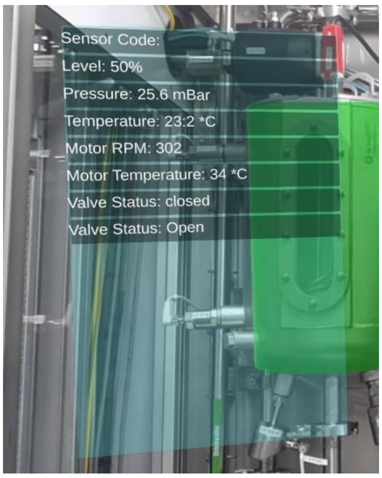


Figure 46. 3D information display

Figure 45. Static information Display

Figure 44. 3D overlay

5. SUMMARY OF EVALUATIONS

In this section the user evaluations are discussed. During the development of the application there have been sessions where the then current prototype was discussed and tested. This resulted in design changes that bring the product to the final prototype. This section consists of two parts: firstly the intermediary evaluations and design changes will be reviewed. Secondly, the final prototype user evaluation will be discussed.

5.1 INTERMEDIARY EVALUATIONS

Functional testing

During the first weeks of the project there were several problems with the Edgecontroller. When the device first arrived, it would not start up properly. With the device sent back to the factory for repairs, the choice was made to create a database with dummy data. The focus was more on the implementation of augmented reality, and data ingestion could be added afterwards as the structure was already tried and tested. When the Edgecontroller finally returned, the rest of the data infrastructure was all ready.

This infrastructure was mainly consisting of the Azure Cloud service. As mentioned in section 4.2 System Architecture the choice was made for Azure due to its advantageousness for students. Moreover, there was plethora of documentation and implementation between C# and Azure, both created by Microsoft. It had the added benefit that no MA-IT server space had to be used, and data injected into the cloud was readily available for testing, wherever the engineers are located.

When talking about data, it is important to mention the format in which it is being transported. First, a custom library was being constructed which would make data transport very easy, but was not easily scalable, generalized or widely used. So where this could make sense to implement in a solution that will not change, this is rarely a given situation.

As mentioned in the realization chapter, several iterations were made on augmented reality. Vuforia has four different types of recognition and augmentation algorithms. Image targets, cylinder targets, model targets and area targets. The first one recognizes images, the second one recognizes a cylindrical shape and can project on this shape correctly, the third one recognizes a structure on the basis of a 3D model and the fourth one uses a 3D scan of an object or area. Due to problems with the latter three options, images were taken of the Equipment Modules. The images were then implemented in an algorithm with could recognize the equipment modules in a video. Unity could then project on the found objects.

User testing

The background screen has also been subject to change. In user tests the background of the screen could use a more interesting color and be more personalized. This was done by changing it into a blurred image of the lemonade factory.

In the augmentation part of the application several iterations were made on the color, shape and movement of the toggles, or CTA's. Changes were made in changing color from orange to blue in order to clearly distinguish them from warning CTA's. Besides, the movement was adapted from a turning motion to a ripple effect, to eliminate confusion that was created by the resemblance to a video loading.

In the first iterations of the application, most of the buttons contained text. However, during the ideation the wish for easily readable application was made. When the buttons worked, a way to change them into icons was figured out. This was done by changing the layout from a classic button to replace it with an image, in this case a home icon was chosen for in the live application, the login screen got an exit icon.

5.2 FINAL PROTOTYPE EVALUATION

Discussion on the MoSCoW prioritization

To evaluate the prototype as it is standing in this case, the Moscow prioritization can function as a checklist. In Tablel 2. MoSCoW prioritization - graded a colored square show whether a requirement is met fully or in part (respectively green and orange). It becomes visible that the Must Have options have been met.

Must have	
UI	• Recognize machine components • augmented reality displaying (t.b.d. how; textual, pointers, other overlays) • visible sensor status • visible machine warnings
Logic	• augmented reality tracking and processing • communication with cloud service (Which is connected to the lemonade factory with the Edgecontroller) • user profile (technician) with user specific access to database information

Should have	
UI	• option to toggle visualizations (such as graphs) • visualization on the pipes (to see where pipes go)
Logic	

Could have	
UI	• Visualize batch information (expected time of duration and time of completion) • Visualize availability of cleaning fluids (how much has been used, how many uses left?) • Selection of user profiles
Logic	• Multiple user profiles with different degrees of access (to information)

Won't have this time	
UI	• Hyperlink to replacement product • Installation/ repair guide
Back end	• Predictive maintenance algorithm (e.g. if motor has ran for 1000 hours, perform maintenance - not in app, should be server side)

Tablel 2. MoSCoW prioritization - graded

The visualizations on the mechanical parts have been realized too. In the previous chapter, pictures of these are available. The options that are not met, have been largely due to time constraints. The options that are still white can be used to further improve the application.

6. DISCUSSION

This section gives a summary of the report, concludes the main findings and attempts to give a concise answer to the sub questions and finally the main research question. Secondly, recommendations on further work are given, in order to keep developing the technology of augmented reality and its implementation in the automated industry.

6.1 CONCLUSION

The main goal of this thesis is to find an answer to the main research question: *How can augmented reality be used in an automated lemonade factory to help specified personnel gain greater insight in the operational data of the machinery compared to static schematic representations?*

In an attempt to answer this question, 4 area's of interest were researched. Firstly, the current uses of augmented reality in several industries, including but not limited to healthcare, architecture, military and MRO's (maintenance, repair, overhaul) and industries such as car and ship manufacturing. Idea's that were taken from this literature research are 3D overlays on real life objects, information connected to the real, physical world and the advantages and disadvantages of different devices which can deploy augmented reality.

The second area of research is which technologies would fit this project best. That requires to know where the stakeholders are interested in. As the client has wishes and a general idea, the main point is to generate ideas which reflect these wishes but are novel too. This resulted in the first prototypes, which were well received and were used as the point of departure for the rest of the development. This first prototype was simply a drawing with certain aspects highlighted.

Thirdly, the wish of the client to use a specific device is also taken into account. Though not an integral part of the research thesis, it was still very related. This device took care of uploading the sensor data to the cloud service. As this was a question more geared towards implementation and not necessarily designing a novel use, less time was attributed to this request. A solution to this question was presented as a series of applications connected in Azure cloud service; data ingestion, routing and a virtual server containing a database.

The fourth question, that has been the main focus in this project, was how the data could be best displayed using augmented reality in the application. The data presentation starts with recognizing where data should be displayed, then how it should be displayed and finally which data should be displayed. Several pitfalls, such as information overload, lack of clarity or accuracy and miscommunication had to be avoided. In order to navigate around these pitfalls design choices were made in a iterative process, where designs were based on literature and user tests. The results of these tests were the basis of new design choices.

The final augmentations were pop-up Call to Actions, as well as warnings and immediate problem warnings. When these are pressed, a text box that was connected to the world in real life appears, as well as overlays on the factory components. In user tests this proved helpful as it connects the information directly to a certain part of the machinery, without having to make the mental connection yourself. The information was received from the cloud service and displayed in a textbox connected to the real world. All in all, the main conclusion of this project is that there is definitely a place where augmented reality shines above a normal static schematic representation, but the application in its current state is not yet ready to overtake its predecessors place.

That being said, augmented reality is a technology that is being developed, and can increase the insight as technology gets better. Therefore I recommend that this augmented reality can definitely have a place in coming projects, if it is continued to be developed. The biggest innovations will come from the direction of choosing what to augment and how to stabilize the augmentations. Making the app more refined and adding options like the ability to enlarge the boxes or to pin them to the screen of the user could be directions that will enhance the usability.

Concluding, augmented reality should be used in places where there is a small risk of collisions with other obstacles. As augmentations may differ from the real world, an operators perception might be incomplete or distracted from reality. In case of using augmentations with a robot setup, this might be dangerous, because the operator is focused on the augmentations while a robot might be in motion.

If MA-IT has found reasonable grounds in this project to continue research in this area I imagine they can provide a novel edge in their corner of the industry for clients that do not necessarily fit size wise in the row of clients that Vuforia already has. In Chapter 5.2 Final Prototype evaluation a list of potential other additions to the application are available.

6.2 RECOMMENDATIONS

The recommendations can be categorized in three different sections. Firstly, technology focused aspects, such as software or hardware related development. Secondly, recommendations on user interaction are given and finally recommendations on use cases.

Although the product was received well by the testers, there is not yet a statistical proof that this solution provides a superior way of visualizing data. Therefore, a series of tests should be conducted to see whether augmented reality actually does provide this. I recommend tests focused on finding parts and looking up information based on the time conventional programs take compared to augmented reality; this is to make it measurable.

Secondly, I recommend to improve on the application itself. There is a functioning framework, but there are plenty of possibilities for improvement. The recognition algorithm is chosen to work as a proof of concept, but is very crude. In order to potentially improve the usability and functionality of the application, the chosen recognition algorithm should either be improved in the current employment or expanded with extra algorithms to make this system more robust. As stated in Chapter 4.3 Iterations great improvements can be made by looking more carefully at the options that Vuforia provides. The possibility of adapting the 3D model so it more closely resembles the current factory could be further investigated. Moreover, the use of markers could help the application to determine its orientation in the room better, thus improving the tracking results and eliminate glitching.

Thirdly, the way data is routed from the factory to the app can be greatly improved. This can be done by figuring out a way to keep the data format as JSON in the database, instead of transforming it from JSON into a table and back to JSON, as it is automatically done right now. This development needs more extensive knowledge on how to achieve a better routing system in a cloud service, specifically Azure if the system is chosen to keep in place.

Finally, the hardware device chosen as platform for the augmented reality app was a mobile phone, but this can definitely be expanded to other platforms, such as tablets or head mounted devices.

Expanding the range of device will give greater reach for enterprises and can be integrated in order to cater to different preferences and job requirements.

7. BIBLIOGRAPHY

[1] "HACCP | NVWA," [Online]. Available: https://www.nvwa.nl/onderwerpen/haccp. [Accessed 24 2 2022].

[2] B&R, "Automation PC 910," B&R, [Online]. Available: https://www.br-automation.com/nl-nl/producten/industrial-pcs/automation-pc-910/. [Accessed 28 4 2022].

[3] My Automation - IT, "Soda Factory CIP en productie test," My automation IT, Deventer, 2020.

[4] T. Wójcicki, "Supporting the diagnostics and maintenance of technical devices with Augmented Reality," *Diagnostyka,* vol. 15, no. 1, pp. 43-47, 2014.

[5] A. Mader and W. Eggink, "A Design Process for Creative Technology," 2014.

[6] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier and B. MacIntyre, "Recent Advances in Augmented Reality," IEEE Xplore, 2001.

[7] R. Azuma, "A Survey of Augmented Reality," *Presence: Teleoperators and Virtual Environments,* vol. 4, no. 6, pp. 355-385, 1997.

[8] P. Milgram, H. Takemura, A. Utsumi and F. Kishino, "Augmented reality: A class of displays on the reality-virtuality continuum," *Telemanipulator and Telepresence Technologies,* vol. 2351, no. 1, 1994.

[9] L. F. de Souza Cardoso, F. C. Martins Queiroz Mariano and E. R. Zorza, "A survey of industrial augmented reality," *Elsevier,* vol. 139, p. 106159, 2020.

[10] M. Bajura, H. Fuchs and R. Ohbuchi, "Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery within the Patient," *ACM SIGGRAPH Computer Graphics,* vol. 26, no. 2, pp. 203-210, 1992.

[11] C. Bichlmeier, F. Wimmer, S. M. Heining and N. Navab, "Hybrid In-Situ Visualization Method for Improving Multi-Sensory Depth Perception in Medical Augmented Reality," in *IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nara, Japan, 2007.

[12] Y. Huang , Y. Liu and Y. Wang, "AR-View: An augmented reality device for digital reconstruction of Yuangmingyuan," in *IEEE International Symposium on Mixed and Augmented Reality - Arts, Media and Humanities*, Orlando, FL, 2009.

[13] A. Webster, S. K. Feiner, B. MacIntyre, W. Massie and T. Krueger, "Augmented reality in Architural Construction, Inspection and Renovation," *Computing in Civil Engineering,* pp. 913-919, 1996.

[14] M. Mekni and A. Lemieux, "Augmented Reality: Applications, Challenges and Future Trends," *Applied Computational Science,* vol. 1, pp. 205-214, 2014.

[15] T. Miyashita, P. Meier, T. Tachikawa, S. Orlic, T. Eble, V. Scholz, A. Gapel, O. Gerl, S. Arnaudov and S. Lieberknecht, "An Augmented Reality museum guide," *IEEE International symposium on Mixed and Augmented reality,* pp. 103-106, 2008.

[16] "Wikitude Navigation," Wikitude, 2010. [Online]. Available: https://www.wikitude.com/showcase/wikitude-navigation/. [Accessed 12 May 2022].

[17] R. Whitwam, "Wikitude Drive for Android review: does AR make driving safer?," ExtremeTech, 30 August 2011. [Online]. Available: https://www.extremetech.com/mobile/94326-wikitude-drive-for-android-review-its-about-the-journey-not-the-destination. [Accessed 12 May 2022].

[18] M. D. Vivo, "How Volkswagen is Using Augmented and Virtual Reality," RelayCars, 23 November 2020. [Online]. Available: https://blog.relaycars.com/volkswagen-augmented-virtual-reality/. [Accessed 14 April 2022].

[19] B. Gerats, "Increasing technicians efficiancy in executing proactive maintenance tasks," University of Twente, Enschede, 2017.

[20] G. Dini and M. Dalle Mura, "Application of Augmented Reality Techniques in Through-life Engineering Services," *Procedia CIRP,* vol. 38, pp. 14-23, 2015.

[21] S. J. Henderson and S. Feiner, "Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret," in *IEEE International Symposium on Mixed and Augmented Reality*, Orlando, FL, USA, 2009.

[22] P. Fraga-Lamas, T. Fernandez-Carames, O. Blanco-novoa and M. Vilar-Montesinos, "A Review on Industrial Augmented Reality," *IEEE Access,* vol. 6, no. DOI: 10.1109/ACCESS.2018.2808326, pp. 13358-13375, 2017.

[23] M. Olbrich, H. Wuest, P. Riess and U. Bockholt, "Augmented Reality Pipe Layout Planning in the Shipbuilding Industry," in *IEEE International Symposium on Mixed and Augmented Reality*, Basel, Switzerland, 2011.

[24] M. A. Livingston and e. al, Military Applications of Augmented Reality, New York, NY: Springer, 2011.

[25] S. Julier, Y. Baillot, M. Lanzagorta and D. G. Brown, "BARS: Battlefield Augmented Reality System," Naval Research Laboratory, Washington, 2004.

[26] C. Bacon, "IVAS goggle amplifies mounted capabilities," U.S. Army, 19 February 2021. [Online]. Available: army.mil/article/243505/ivas_goggle_amplifies_mounted_capabilities. [Accessed 12 May 2022].

[27] V. Luckerson, "Google Glass explorer program ends," Time, 15 January 2015. [Online]. Available: https://time.com/3669927/google-glass-explorer-program-ends/. [Accessed 2022 May 9].

[28] "Introducing Glass Enterprise Edition 2," Google Glass, [Online]. Available: https://www.google.com/glass/start/. [Accessed 10 May 2022].

[29] "Hololens," Microsoft, 2022. [Online]. Available: https://www.microsoft.com/en-us/hololens. [Accessed 5 5 2022].

[30] "5 Benefits of Assisted Reality," RealWear, 29 January 2021. [Online]. Available: https://www.realwear.com/blog/5-benefits-of-assisted-reality/. [Accessed 5 5 2022].

[31] Realware, [Online]. Available: https://www.realwear.com/. [Accessed 21 April 2022].

[32] "Vuzix M4000," Vuzix, [Online]. Available: https://www.vuzix.com/products/m4000-smart-glasses. [Accessed 13 May 2022].

[33] "How Does Snapchat's Facial Recognition Software Work?," PSafe, 19 February 2018. [Online]. Available: https://www.psafe.com/en/blog/snapchats-facial-recognition-software-work/#:~:text=To%20put%20it%20in%20layman's,the%20features%20of%20these%20filters.. [Accessed 5 May 2022].

[34] "Welke functies zijn er beschikbaar op AR-zone in de Galaxy Z Flip?," Samsung, 2022. [Online]. Available: https://www.samsung.com/nl/support/mobile-devices/welke-functies-zijn-er-beschikbaar-op-ar-zone-in-de-galaxy-z-flip/. [Accessed 6 May 2022].

[35] A. Rovira, A. F. g. Schieck, P. Blume and S. Julier, "Guidance and surroundings awareness in outdoor handheld augmented reality," *Plos One,* vol. 15, no. 3, pp. 1-14, 2020.

[36] M. Bode, "Evaluation of an Augmented Reality Assisted Manufacturing System for Assembly Guidance," University of Twente, Enschede, 2019.

[37] A. Doshi, R. T. Smith, B. H. Thomas and C. Bouras, "Use of projector based augmented reality to improve manual spot-welding precision and accuracy for automotive manufacturing," *The International Journal of Advanced Manufacturing Technology,* no. 89, pp. 1279-1293, 2017.

[38] "Vuforia Engine," PTC, 2022. [Online]. Available: https://www.ptc.com/en/products/vuforia/vuforia-engine. [Accessed 13 May 2022].

[39] H. Regenbrecht, G. Baratoff and W. Wilke, "Augmented reality projects in the automotive and aerospace industries," *IEEE Computer Graphics and Applications,* vol. 25, no. 6, doi: 10.1109/MCG.2005.124., pp. 48-56, 2005.

[40] R. Palmarini, J. A. Erkoyuncu , R. Roy and H. Torabmostaedi, "A systematic review of augmented reality applications in maintenance," *Elsevier,* vol. 49, pp. 215-228, 2018.

[41] "ProjectFramework 10 MoSCoW Prioritisation," Agile Business Consortium, 2022. [Online]. Available: https://www.agilebusiness.org/page/ProjectFramework_10_MoSCoWPrioritisation. [Accessed 24 March 2022].

[42] J. Schell, "Eight Tips for Productive Prototyping," in *The Art of Game Design*, Burlington, Morgan Kaufmann Publishers, 2008, pp. 86-95.

[43] "Law of proximity," [Online]. Available: https://lawsofux.com/law-of-proximity/.

[44] Editors, "Industrial Revolution," Brittanica, 13 March 2022. [Online]. Available: https://www.britannica.com/event/Industrial-Revolution. [Accessed 14 April 2022].

[45] K. Schwab, "The Fourth Industrial Revolution," Britannica, 25 May 2018. [Online]. Available: https://www.britannica.com/topic/The-Fourth-Industrial-Revolution-2119734#ref1255187. [Accessed 14 April 2022].

[46] G. Picard, "Ivan Sutherland, Father of AR," Atomicdigital.design, 29 May 2020. [Online]. Available: https://atomicdigital.design/blog/1965-ivan-sutherland-father-of-ar. [Accessed 14 April 2022].

[47] "Augmented Reality Market Size, Share & Trends Analysis," Grand View Research, San Francisco, 2021.

[48] "Augmented Reality for MRO (Maintenance, Repair and Overhaul) Market - Global Forecast to 2024," Infoholic Research LLP, [Online]. Available: https://www.infoholicresearch.com/report/augmented-reality-for-mro-market/. [Accessed 14 April 2022].

[49] "Varjo XR-3," Varjo, 2022. [Online]. Available: https://varjo.com/products/xr-3/. [Accessed 5 5 2022].

[50] R. Rosser, Y. Tan, H. J. Kennedy, J. Jeffers, D. DiCicco and X. Gong, "Image insertion in video streams using a combination of physical sensors and pattern recognition". USA Patent US6100925A, 8 August 2000.

[51] P. Milgram, H. Takemura, A. Utsumi and F. Kishino, "Augmented Reality: A class of displays on the reality-virtuality continuum," *Telemanipulator and Telepresence Technologies,* no. 2351, pp. 282-292, 1994.

APPENDIX 1 : CODE

AR app

Switch scene

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class switchScene : MonoBehaviour
{
    public void goHome()
    {
        SceneManager.LoadScene(0);
    }

    public void playGame()
    {
        SceneManager.LoadScene(1);
    }

    public void closeApp()
    {
        Application.Quit();
        TCPTestClient.TCPClient.StopThread();
        Debug.Log("Quit!");
    }
}
```

BoxAppearDisappear.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class BoxAppearDisappear : MonoBehaviour
{
    public GameObject Plane;
    public GameObject Pipes;

    private bool isActive = false;


    private void Start()
    {
        Plane.SetActive(false);
        Pipes.SetActive(false);
    }

    private void Update()
    {
        if (Input.touchCount > 0 && Input.touches[0].phase == TouchPhase.Began)
        {
            Ray ray = Camera.main.ScreenPointToRay(Input.touches[0].position);
            RaycastHit hit;

            if (Physics.Raycast(ray, out hit))
            {
                if (hit.collider != null) //if a hit is detected
                {
                    if (hit.collider.tag == "CleaningTankL" && !isActive)
                    {
                        Plane.SetActive(true);
                        Pipes.SetActive(true);

                    }

                    if (hit.collider.tag == "CleaningTankL" && isActive)
                    {
```

```
                        Plane.SetActive(false);
                        Pipes.SetActive(false);
                    }

                    isActive = !isActive;
                }
            }
        }
    }
}
```

Changing_Colors.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class changing_color : MonoBehaviour
{
    GameObject ripple;

    public void activeColor()
    {
        var rippleRenderer = ripple.GetComponent<Renderer>();
        rippleRenderer.material.SetColor("_Color", Color.red);
    }

    public void colorVisited()
    {
        var rippleRenderer = ripple.GetComponent<Renderer>();
        rippleRenderer.material.SetColor("_Color", Color.green);
    }

}
```

jsonDataClass.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[Serializable]
public class jsonDataClass
{
    public List<Sensor> Sensors;

}

[Serializable]
public class Sensor
{
    public string psid;
    public string value;

    //public list<...> psid;
    //public list<...> value;
}
```

TCPTestClient.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Net.Sockets;
using System.Text;
using System.Threading;
using UnityEngine;
using TMPro;
```

```
public class TCPTestClient : MonoBehaviour
{
    public static TCPTestClient TCPClient;

    #region private members
    private TcpClient socketConnection;
    private Thread clientReceiveThread;
    #endregion

    private void Awake()
    {
        if (TCPClient == null)
        {
            DontDestroyOnLoad(gameObject);
            TCPClient = this;
        }
        else if (TCPClient != null)
        {
            Destroy(gameObject);
        }
    }

    private string customIP;
    public string retrievedMessage = null;

    public void ConnectToTcpServer(string Somestring)
    {
        customIP = Somestring;

        clientReceiveThread = new Thread(new ThreadStart(ListenForData));
        clientReceiveThread.IsBackground = true;
        clientReceiveThread.Start();
    }

    private void ListenForData()
    {
        try
        {
            //socketConnection = new TcpClient("192.168.1.127" , 8052); //use userInput_field as
connection method to the laptopserver : this connection persists in the other scenes too.
            socketConnection = new TcpClient(customIP, 8052);
            Byte[] bytes = new Byte[1024];
            while (true)
            {
                // Get a stream object for reading
                using (NetworkStream stream = socketConnection.GetStream())
                {
                    int length;

                    // Read incomming stream into byte array.
                    while ((length = stream.Read(bytes, 0, bytes.Length)) != 0)
                    {
                        var incommingData = new byte[length];
                        Array.Copy(bytes, 0, incommingData, 0, length);

                        // Convert byte array to string message.

                        string serverMessage = Encoding.ASCII.GetString(incommingData);
                        retrievedMessage = serverMessage;
                    }
                }
            }
        }
        catch (SocketException socketException)
        {
            Debug.Log("An error occured: " + socketException);
        }
    }

    /// <summary>
    /// Send message to server using socket connection.
    /// </summary>
    public void SendClientMessage(string tag)
```

```
    {
        if (socketConnection == null)
        {
            return;
        }
        try
        {
            // Get a stream object for writing.
            NetworkStream stream = socketConnection.GetStream();
            if (stream.CanWrite)
            {
                // Convert string message to byte array.
                byte[] clientMessageAsByteArray = Encoding.ASCII.GetBytes(tag);
                // Write byte array to socketConnection stream.
                stream.Write(clientMessageAsByteArray, 0, clientMessageAsByteArray.Length);
                // Debug.Log("Client sent his message - should be received by server");
            }
        }
        catch (SocketException socketException)
        {
            Debug.Log("Socket exception: " + socketException);
        }
    }

    public void StopThread()
    {
        clientReceiveThread.Abort();
    }
}
```

DBConnect.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Net.Sockets;
using System.Text;
using System.Threading;
using UnityEngine;
using UnityEngine.SceneManagement;
using TMPro;

public class DBConnect : MonoBehaviour
{
    // Use this for initialization
    public TMP_InputField enterIP;
    public TextMeshProUGUI DebugLog;

    private string customIP;
    public void connectToDB()
    {
            customIP = enterIP.text;
            TCPTestClient.TCPClient.ConnectToTcpServer(customIP);
            SceneManager.LoadScene(1);
    }
}
```

ShowText.cs

```
using System.Collections;
using System.Collections.Generic;
using System.Text;
using UnityEngine;
using UnityEngine.UI;
using TMPro;

public class showText : MonoBehaviour
{

    public GameObject StaticBox;
    jsonDataClass jsonData = new jsonDataClass();
```

```
changing_color colorHandler = new changing_color();
#region panels
public GameObject EMNBox;
public GameObject levelBox;
public GameObject pressureBox;
public GameObject temperatureBox;
public GameObject motorRPMBox;
public GameObject MotorTemperatureBox;
public GameObject ValveOneStateBox;
public GameObject ValveTwoStateBox;
#endregion
#region TMProUGUI
public TextMeshProUGUI EquipmentModuleName;
public TextMeshProUGUI Level;
public TextMeshProUGUI Pressure;
public TextMeshProUGUI Temperature;
public TextMeshProUGUI MotorRPM;
public TextMeshProUGUI MotorTemperature;
public TextMeshProUGUI ValveOneState;
public TextMeshProUGUI ValveTwoState;

public TextMeshProUGUI TestReceivedText;
public TextMeshProUGUI debugLog;
#endregion
private bool isActive;
bool isSameTag;

private string TempTag;
private string ButtonTag;
private string currentTag;

private string oldMessage = null;

private Collider isRayed;

List<TextMeshProUGUI> ListElements = new List<TextMeshProUGUI>();
List<GameObject> ListPanels = new List<GameObject>();

public List<Transform> Objs;
// Start is called before the first frame update
void Start()
{
    ListElements.Add(EquipmentModuleName);
    ListElements.Add(Level);
    ListElements.Add(Temperature);
    ListElements.Add(Pressure);
    ListElements.Add(MotorRPM);
    ListElements.Add(MotorTemperature);
    ListElements.Add(ValveOneState);
    ListElements.Add(ValveTwoState);

    ListPanels.Add(EMNBox);
    ListPanels.Add(levelBox);
    ListPanels.Add(pressureBox);
    ListPanels.Add(temperatureBox);
    ListPanels.Add(motorRPMBox);
    ListPanels.Add(MotorTemperatureBox);
    ListPanels.Add(ValveOneStateBox);
    ListPanels.Add(ValveTwoStateBox);

    StaticBox.SetActive(false);
    foreach (GameObject d in ListPanels)
    {
        d.SetActive(false);
    }

}

// Update is called once per frame
void Update()
{
    //StartCoroutine(getDataToo());
    CheckForHit();
```

```
    if (ButtonTag != TempTag)
    {
        GetData(ButtonTag);
        displayStaticBox(isSameTag);
        TempTag = ButtonTag;
    }
}

public void CheckForHit()
{
    if (Input.touchCount > 0 && Input.touches[0].phase == TouchPhase.Began)
    {
        Ray ray = Camera.main.ScreenPointToRay(Input.touches[0].position);
        RaycastHit hit;

        if (Physics.Raycast(ray, out hit))
        {
            if (hit.collider != null) //if a hit is detected
            {
                TCPTestClient.TCPClient.SendClientMessage(hit.collider.tag);
                bool sameTag = false;
                var rippleRenderer = hit.collider.GetComponent<Renderer>();
                //set all ripple effects to blue

                if (hit.collider.tag == currentTag)
                {
                    sameTag = true;
                    rippleRenderer.material.SetColor("_Color", Color.yellow);
                }
                if(hit.collider.tag != currentTag)
                {
                    foreach(Transform Tform in Objs)
                    {
                        Tform.GetComponent<Renderer>().material.color = Color.blue;
                    }
                    rippleRenderer.material.SetColor("_Color", Color.green);
                }

                isRayed = hit.collider;
                displayStaticBox(sameTag);
                currentTag = hit.collider.tag; //for this function only
                ButtonTag = hit.collider.tag; //to make it work with the testing buttons
            }
        }

    }

}
#region test buttons
public void testCLTLeft()
{
    ButtonTag = "CleaningTankL";
    TCPTestClient.TCPClient.SendClientMessage(ButtonTag);
}

public void testCLTRight()
{
    ButtonTag = "CleaningTankR";
    TCPTestClient.TCPClient.SendClientMessage(ButtonTag);
}
#endregion
//public void displayStaticBox(bool sametag)
//{
//    if (!isActive)
//    {
//        StaticBox.SetActive(true);
//    }
//    else if (isActive && sametag)
//    {
//        StaticBox.SetActive(false);
//    }
//    isActive = !isActive;
//}
```

```
public void displayStaticBox(bool sametag)
{
    if (!StaticBox.activeInHierarchy)
    {
        StaticBox.SetActive(true);
    }
    else if (StaticBox.activeInHierarchy && sametag)
    {
        StaticBox.SetActive(false);
    }
    //isActive = !isActive;
}

public void GetData(string tag)
{
    //receive the json data from azure's reply and make it into an array
    string AzureString = TCPTestClient.TCPClient.retrievedMessage;

    if (AzureString != "" || AzureString != oldMessage)
    {
        string jsonSTR = "{ \"Sensors\" : " + AzureString + "}";
        processJSONData(jsonSTR);
        displayList(tag);
        oldMessage = AzureString;
        //every time a message is received, send another request to check for updates
        //TCPTestClient.TCPClient.SendClientMessage(tag);
    }
}

public IEnumerator getDataToo()
{
    string AzureString = TCPTestClient.TCPClient.retrievedMessage;
    if (AzureString == "" || AzureString == oldMessage)
    {
        Debug.Log("waiting...");
        yield return new WaitForSeconds(1f);
    }
    string jsonSTR = "{ \"Sensors\" : " + AzureString + "}";
    processJSONData(jsonSTR);
    displayList(tag);
    oldMessage = AzureString;
}

//THIS THE DESERIALIZER
public void processJSONData(string json)
{
    jsonData = JsonUtility.FromJson<jsonDataClass>(json);
}

public void displayList(string tag)
{
    foreach (GameObject g in ListPanels)
    {
        g.SetActive(false);
    }

    int y = 0;

    ListElements[0].text = tag;
    ListPanels[0].SetActive(true);

    foreach (Sensor x in jsonData.Sensors)
    {
        ListPanels[y + 1].SetActive(true);
        ListElements[y + 1].text = x.psid + " : " + x.value;
        y++;
    }
}

public void changeColor()
{
    foreach (Transform Tform in Objs)
    {
        Tform.GetComponent<Renderer>().material.color = Color.red;
```

```
        }
    }
}
```

Server Edgecontroller

program.cs

```
using System;

namespace TCPTestServer
{
    class Program
    {
        static void Main(string[] args)
        {
            TCPTestServer test = new TCPTestServer();
            test.Start();
            while (true)
            { }
        }
    }
}
```

Testserver.cs

```
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
using System.Threading;


namespace TCPTestServer
{
    public class TCPTestServer
    {
        #region private members
        /// <summary>
        /// TCPListener to listen for incomming TCP connection
        /// requests.
        /// </summary>
        private TcpListener tcpListener;
        /// <summary>
        /// Background thread for TcpServer workload.
        /// </summary>
        private Thread tcpListenerThread;
        /// <summary>
        /// Create handle to connected tcp client.
        /// </summary>
        private TcpClient connectedTcpClient;
        #endregion

        //create new AzureConnection object.
        AzureConnection ACtest = new AzureConnection();
        private string localIPAddress;

        // Use this for initialization
        public void Start()
        {
            // Start TcpServer background thread
            tcpListenerThread = new Thread(new ThreadStart(ListenForIncommingRequests));
            tcpListenerThread.IsBackground = true;
            tcpListenerThread.Start();
        }

        private void getIP()
        {
            //string host = ;
            var host = Dns.GetHostEntry(Dns.GetHostName());
            foreach(var ip in host.AddressList)
            {
```

```
        if(ip.AddressFamily == AddressFamily.InterNetwork)
        {
            localIPAddress = ip.ToString();
            Console.WriteLine("Use this IP address in the app: " + localIPAddress);
        }
    }
}

/// <summary>
/// Runs in background TcpServerThread; Handles incomming TcpClient requests
/// </summary>
private void ListenForIncommingRequests()
{
    getIP();
    try
    {
        // Create listener on local server port 8052.
        //tcpListener = new TcpListener(IPAddress.Parse("192.168.1.127"), 8052);
        tcpListener = new TcpListener(IPAddress.Parse(localIPAddress), 8052);
        tcpListener.Start();
        Console.WriteLine("Server is listening");
        Byte[] bytes = new Byte[1024];
        while (true)
        {
            using (connectedTcpClient = tcpListener.AcceptTcpClient())
            {
                // Get a stream object for reading
                using (NetworkStream stream = connectedTcpClient.GetStream())
                {
                    int length;
                    // Read incomming stream into byte arrary.

                    while ((length = stream.Read(bytes, 0, bytes.Length)) != 0)
                    {
                        var incommingData = new byte[length];
                        Array.Copy(bytes, 0, incommingData, 0, length);
                        // Convert byte array to string message.

                        string clientMessage = Encoding.ASCII.GetString(incommingData);

                        Console.WriteLine("\n_________________________________________");
                        Console.WriteLine("client message received as: " + clientMessage);
                        string tempResponse = ACtest.GetReqAzure(clientMessage);
                        SendMessage(tempResponse);
                        Console.WriteLine("\n_________________________________________");
                    }
                }
            }
        }
    }
    catch (SocketException socketException)
    {
        Console.WriteLine("SocketException " + socketException.ToString());
    }
}

/// <summary>
/// Send message to client using socket connection.
/// </summary>
private void SendMessage(string parMessage)
{
    if (connectedTcpClient == null)
    {
        return;
    }

    try
    {
        // Get a stream object for writing.
        NetworkStream stream = connectedTcpClient.GetStream();
        if (stream.CanWrite)
        {
            // Convert string message to byte array.
            byte[] serverMessageAsByteArray = Encoding.ASCII.GetBytes(parMessage);
```

```
                    // Write byte array to socketConnection stream.
                    stream.Write(serverMessageAsByteArray, 0, serverMessageAsByteArray.Length);
                    // Console.WriteLine("Server sent his message - should be received by client");
                }
            }
            catch (SocketException socketException)
            {
                Console.WriteLine("Socket exception: " + socketException);
            }
        }
    }
}
```

AzureConnect.cs

```
using System;
using System.Text;
using Microsoft.Data.SqlClient;


namespace TCPTestServer
{
    public class AzureConnection
    {

        private string DBresults;

        //first string is the same as the tag that is passed, second part is the corresponding query.
        string[,] EquipmentModuleList = new string[,] { { "CleaningTankL", "SELECT TOP (5) t.psid,
JSON_VALUE([values], '$[0].value') AS value FROM [dbo].[ECdata] t WHERE t.tid = '6ff045fe-68a1-4fe7-
a03d-b0a72a6ba51b' FOR JSON PATH" },
                                                        { "CleaningTankR", "SELECT TOP (3) t.psid,
JSON_VALUE([values], '$[0].value') AS value FROM [dbo].[ECdata] t FOR JSON PATH" },
                                                        { "QuadPump", "Some query that represents
QuadPump" },
                                                        { "MotorBottom", "Some query that represents
MotorBottom" },
                                                        { "LemonadeDispenser", "Some query that
represents LemonadeDispenser" },
                                                        { "RoutingAssembly", "Some query that
represents RoutingAssembly" },
                                                        { "MixingTank", "Some query that represents
MixingTank" },
                                                        };

        public string GetReqAzure(string EquipmentModule)
        {
            //init the sql string: will be filled differently when this function is called again.
            String sql = "SELECT TOP(3) [tid] , [psid] FROM ECdata";

            //this loop compares the received tag to a list of tags - with accompanying queries.
            //When match, it assigns the query to the 'sql' string.
            for (int i = 0; i < EquipmentModuleList.GetLength(0); i++)
            {
                if (EquipmentModuleList[i, 0] == EquipmentModule)
                {
                    //Console.WriteLine(EquipmentModuleList[0, 1]);
                    sql = EquipmentModuleList[i, 1];
                    break;
                }
            }

            StringBuilder tempResult = new StringBuilder();
            try
            {
                SqlConnectionStringBuilder builder = new SqlConnectionStringBuilder();
                builder.DataSource = "unitytestdatabase.database.windows.net";  //servername
                builder.UserID = "unitydatabase";                               //log in credentials
                builder.Password = "Unitytest1";
                builder.InitialCatalog = "EdgecontrollerData";                  //database

                using (SqlConnection connection = new SqlConnection(builder.ConnectionString))
                {
                    connection.Open();
```

```
                    Console.WriteLine("\nQuery data example:");

                    using (SqlCommand command = new SqlCommand(sql, connection))
                    {
                        // ExecuteScalar only gets the first result of the query
                        // Since this does not apply to the 'FOR JSON PATH', this method can be used
                        DBresults = command.ExecuteScalar().ToString();
                        tempResult.Append(DBresults);
                        Console.WriteLine(tempResult);
                        //using (SqlDataReader reader = command.ExecuteReader())
                        //{
                        //    while (reader.Read())
                        //    {
                        //        //Console.WriteLine("{0} {1}", reader.GetString(0),
reader.GetString(1));
                        //        DBresults = (reader.GetString(0) + " " + reader.GetString(1));
                        //        tempResult.Append(DBresults);
                        //    }
                        //}
                    }
                }
            }
            catch (SqlException e)
            {
                Console.WriteLine(e.ToString());
            }
            return tempResult.ToString();
        }
    }
}
```

UNIVERSITY OF TWENTE
Drienerlolaan 5
7522 NB Enschede

P.O.Box 217
7500 AE Enschede

P +31 (0)53 489 9111

info@utwente.nl
www.utwente.nl