BACHELOR THESIS

INCREASING TECHNICIAN'S EFFICIENCY IN EXECUTING PROACTIVE MAINTENANCE TASKS
AN AUGMENTED REALITY CONCEPT

BEEREND GERATS
CREATIVE TECHNOLOGY / FACULTY OF EEMCS

SUPERVISORS:
DR. J. ZWIERS (UNIVERSITY OF TWENTE)
IR. R.G.A. BULTS (UNIVERSITY OF TWENTE)
IR. J. VAN KREIJ (INNOVADIS)

JULY 2017

UNIVERSITY OF TWENTE.

Abstract

This report describes the design process of an augmented reality tool to assist technicians during maintenance work at machines within the factory of tire manufacturer Apollo Vredestein. Aim of this study is to improve the efficiency of technicians performing proactive maintenance tasks, to reduce the high workload that they are currently experiencing. After observations at the Apollo Vredestein factory and semi-structured interviews with technicians, a technician supervisor and an expert in the domain of factory maintenance, it is decided to construct a communication tool. This tool enables technicians to assist each other over distance with visual communication. A tablet application is build that provides technicians with the ability to draw and place virtual instructions on top of a machine. The virtual content aligns with the machine by augmented reality technology, which can be enabled by scanning a custom designed image target at a machine area. When scanning an image target, the technician is provided with specific documentation of the corresponding machine area. An evaluation is performed with two technicians that are asked to work with the prototype during two real use situations. From the results of this evaluation can be concluded that the augmented reality solution has large potential to increase the technicians' efficiency in performing both proactive and reactive maintenance work. Both technicians considered the product useful, usable and learnable. During the whole process of this study, participatory design is executed with a technician of Apollo Vredestein as co-designer of the augmented reality tool. This resulted in a product that fits the technicians' needs and made the technician feel as an 'owner' of the final prototype.

Acknowledgements

First, I would like to thank both supervisors from the University of Twente, dr. Job Zwiers and ir. ing. Richard Bults, for the support, feedback and helpful ideas during my graduation process. Second, I would like ir. Jurgen van Kreij to provide me with the opportunity to perform the graduation project at Innovadis. Third, Leon Pelders I would like to thank as well for his time, auxiliary discussions and feedback on the report from the perspective of expert in factory maintenance. Also I am grateful that Dennis Bakker, Maarten Kanis and Bert Boerrigter were available to participate in product discussions, interviews and/or the user evaluation. Without them it was not possible to obtain the data necessary to perform this study. Last, I would like to thank my family, friends and colleagues at Innovadis for the support and insights during this period.

Table of Contents

List of Figures

List of Abbreviations

Abbreviation	Description
PIPO	Periodieke Inspectie, Periodiek Onderhoud (English: Periodic Inspection, Periodic Maintenance)
PISA	Poetsen, Inspecteren, Smeren, Afstellen (English: Clean, Inspect, Lubricate, Adjust)
HMD	Head-mounted Display
AR	Augmented Reality
VR	Virtual Reality
OBD	On Board Diagnostics
JIT Management	Just In Time Management
DMS	Document Management System
SAP	System Applications Products
PC	Personal Computer
HOTT	Hands On Tool Time
PACT	People, Activities, Context, Technology

1. Introduction

Technological evolution has caused major shifts in the industry environments over the last centuries. After three “industrial revolutions”, “advanced digitalization, […] Internet technologies and future-oriented technologies in the field of ‘smart’ objects seems to result in a new fundamental shift in industrial production.” [6] According to this study, this new way of production causes opportunities for customization and product batches with the size of one. Together with this fourth industrial revolution, the machines that are within the factory become more complex. This means that the machines consist out of more and smaller components, of which the percentage of electrical components increases compared to the amount of mechanical components. As a result from this complexity, highly customized products can be manufactured.

Although such an evolution offers many opportunities, the environment of the factory should also evolve in order to remain production processes optimal. Technicians, responsible for maintenance and repair of the machines, should obtain more challenging knowledge than before and they should use smart tools to work more efficiently and more effectively. Additionally, since factories are working with Just-In-Time management nowadays, including smaller buffers between machines, it forces technicians to respond quicker on a defect. According to technicians working at the factory of Apollo Vredestein, a combination of these factors has increased their workload substantially. They tell that while the total amount of technicians in the factory has decreased, the total amount of work has only increased since then. On the other hand, machines have become more reliable, causing less defects, and maintenance processes have increased in efficiency. This case asks for observations at the actual state of the technicians work and how their tasks could be performed more efficiently without the technicians losing value for their job.

In 2016, the software company Innovadis collaborated with the two companies Eluxis and Pelders Maintenance Management with the aim to build a digital tool assisting technicians in Dutch factories. This tool provides clear maintenance tasks towards technicians, at which they could give feedback. It is expected that this tool will increase the technicians’ efficiency and effectivity, since the tool will give these workers an opportunity to mobilize information. However, the processes within the way of working of technicians ask for further improvement, said by Leon Pelders, owner of Pelders Maintenance Management and an expert in the domain of factory maintenance. He aims to search for tools that include advanced technology to assist technicians and will enable them to work more efficiently and more effectively. Although it is expected that advanced technology can help the technicians, such tools are regularly not included in their toolboxes. This aims for a study that investigates the value of such an advanced tool for technicians in factories.

An advanced technology that is being explored with high interest by many technology firms nowadays is augmented reality (AR). This technology enables a person to see a layer of virtual objects over his view of the real world, meaning this world is enhanced with additional information. The augmented reality technology offers such large opportunities that leading technology firms are working intensively on augmented realty development or are changing their strategic choices towards it. This large amount of interest towards augmented reality leads to the question whether it would be useful for the maintenance domain, especially for assisting technicians in their way of working. When searching for literature about augmented reality, it can be found that many studies are focussed on the technology or are exploring the status quo of existing AR solutions. However, few studies can be found on how users react to an augmented reality experience in their own work field. This study helps filling this gap with researching the obtained value of technicians towards using an AR solution in their work.

1.1 Research Questions

The given actual situation of the way of working of technicians at Apollo Vredestein and the actual status of the augmented reality technology, gives the idea that an augmented reality product could be created that is helping the technicians to be more effective in their work. Goal of this research is to develop an augmented reality product, and to test whether technicians value this product as assisting them in working more efficiently. The scope of this research is to the proactive maintenance tasks of complex machines in factories. Proactive maintenance tasks are more easily to be observed, since they occur in a periodic manner. Complex machines of factories are chosen, since this is also the target of the digital tool of Innovadis. Combining these variables results in the following main research question of this study:

RQ: "Can an augmented reality solution improve technicians' efficiency in executing proactive maintenance tasks to complex machines in factories?"

To be able to answer the main research question, four topics are examined first. These four topics resulted in four subquestions that are answered along the study. In order to obtain knowledge on what exists in the domain of augmented reality in combination with maintenance and repair, a state of the art study is performed. This provides the first subquestion:

SQ1: "Which categories of augmented reality products exist in the domain of maintenance and repair?"

Before thinking of an idea for an augmented reality product that could assist the technicians at Apollo Vredestein, it explored what technicians' needs are that could be fulfilled by this product. To obtain these, the way of working of the technicians at the factory of Apollo Vredestein is examined and perceived in detail. Therefore, the second subquestion of this study is:

SQ2: "How are proactive maintenance tasks executed by technicians in the factory of Apollo Vredestein?"

After detailed information is obtained on the state of the art of augmented reality in the domain of maintenance and repair and on the way of working of technicians at the factory of Apollo Vredestein, concepts are described on how an augmented reality product could hypothetically improve these technicians' efficiency. Therefore, the third subquestion is:

SQ3: "How will an augmented reality product possibly improve the technician's efficiency at Apollo Vredestein?"

Last part of this study includes examination of the perceived value of the technicians at the factory of Apollo Vredestein towards a constructed prototype of an augmented reality product. This constructs the last subquestion of this study:

SQ4: "What are the added values for technicians at Apollo Vredestein when using the developed augmented reality product?"

The four subquestions are answered in four different phases of this research: state of the art, ideation, specification and evaluation (respectively, chapters 3, 4, 5 and 7). In order to create a prototype that provides an augmented reality experience in the maintenance and repair domain, a realization phase is present in which this prototype is implemented (chapter 6). These phases follow the 'Design Process for Creative Technology' described in [4]. Before the description of the phases, the 'Methods and Techniques' chapter will describe all methods and processes that are used during this study (chapter 2). Last, conclusions and future work are described in the discussion (chapter 8).

2. Methods and Techniques

To be able to answer the research questions of this project, an augmented reality product has been designed. This product enables user testing at the evaluation phase, which results in description of conclusions and future work. In order to design a prototype fulfilling user needs, methods and processes are used to ensure the quality of the outcomes. Beneath, two processes are described that are used during this project. These from the framework of this project. Inside these processes a number of methods is used to obtain information in a theoretical 'right way'. The methods are listed and explained briefly in this chapter as well.

2.1 Design Process for Creative Technology

A process used regularly in graduation projects of Creative Technology students is the 'Design Process for Creative Technology' in [4]. This literature describes the design process along four phases that are also implemented in this study: ideation, specification, realization and evaluation. Additionally, the design process describes these phases as an iterative process, meaning the project should include space for reconsidering design choices.

As described in [4] "result of the ideation phase is a (more) elaborated project idea, together with the problem requirements", and includes the evaluation of early ideas with clients, users and experts. In order to understand the problem requirements, the way of working of the users should be understood first. To know who the end-user group will include and exclude, and to be able to understand the environment of this group, a stakeholder identification is performed at the Apollo Vredestein environment (section 4.1). In order to obtain a deeper understanding of the end-users' way of working, observations are performed within this factory (section 4.2). These observations result in a PACT analysis including user scenarios that describes the current way of working of the end-users (section 4.3). To expand understanding, semi-structured interviews are performed with the end-user and an expert. During these interviews the results of the observations are evaluated as well as an early prototype.

Where the ideation phase ends with a product idea, the specification phase elaborates this idea to a concrete product concept. This phase includes a number of iterations that consists of the construction of mock-ups or prototypes and the evaluation with the end-user and the expert. This evaluations are performed in semi-structured interviews. Based on these evaluations, designer perspective scenarios are created that describes situations with the use of an augmented reality product. At the end of the iterative process in the specification phase, the final set of functional and non-functional requirements are determined together with the end-user and the expert. In order to prioritize these, the MoSCoW method [24] is used.

The requirements are fulfilled with a prototype that is build during the realization phase. Since the functional requirements are prioritized it is clear which functionalities are necessary to create first, and which later in the project.

The evaluation phase is the last phase of the design process, and includes methods on evaluating the prototype that is created during the realization phase. Two methods are used for the evaluation: functional evaluation and user evaluation. From the results of these evaluations, project conclusions and future work is described.

2.2 Participatory Design

An essential part of this research is finding a solution that fits the user needs, instead of a solution that is only considered valuable by the researcher. [26] writes that "participatory design is a diverse collection of principles and practices aimed at making technologies […] more responsive to human needs." Cause of this aim is that the people are directly involved in design choices for technology which they will use eventually.

In addition, the support by end-users for the functionality of a new system is an important factor for the acceptation of this system in a later phase. A realistic concern for designers of new technology is that end-users will not accept technology that changes to rapidly. Participatory design is "enabling people to develop realistic expectations, and reducing resistance to change, where the ambition is to ensure that the resulting artifacts support the current and planned changes" [26]. The participatory design methodology, described in [25], is a way to "ensure that participants' interpretations are taken into account in the research." Essential to this methodology is that these users are participating in design choices during all phases of the system design process.

Spinuzzi describes in [25] the three stages of participatory design:

1. *Initial exploration of work*: describing the importance of exploring the way of working of the users, including procedures, technology, routines and workflows.
2. *Discovery processes*: which uses the outcomes of the initial exploration, describes users' goals and "envisions a future workplace" [25]. These goals and a future perspective should be constructed and discussed with the participating users.
3. *Prototyping*: iteratively, several prototypes should be constructed and discussed with the participating users.

The participatory design process is used in this research during the phases: ideation, specification and evaluation. During the ideation phase, initial exploration of work is performed with observations and interviews with the participating user. Also discovery processes are performed in this phase. These include the product idea that is evaluated with the user. During the specification phase these discovery processes remain present. After a number of iterations with the participating user, the research "envisions a future workplace" with three designer perspective scenarios. The last stage of participatory design, prototyping, is performed during the iterations in the specification phase. Last, the participatory design will be evaluated in the evaluation phase with the examination of the feeling of ownership towards the prototype at the participating user.

2.3 Methods

During the description of the design process and the participatory design process, several methods were named. These are listed beneath, and briefly explained per method.

2.3.1 Stakeholder Identification & Analysis

As described in the design process, a stakeholder identification and analysis is performed with the goal to understand the environment of an augmented reality product that is used by technicians in the factory of Apollo Vredestein. More specifically, a goal is to obtain information on who are possibly affected by such a product, and who of these are potentially end-users of the product. Although the definition for stakeholder varies a lot within different researches, the definition for a stakeholder is chosen to be a definition from Dix [7], since his definition is used frequently in software engineering. This definition describes a stakeholder as "anyone whose jobs will be altered, who supplies or gains information from it, or whose power or influence within the organization will increase or decrease."

The stakeholders are identified with the stakeholder identification methodology of Sharp [5]. This study describes identification of "baseline stakeholders" first, after which clients, suppliers and "satellite stakeholders" are identified as well. Baseline stakeholders form the group of stakeholders that are closest to the product or have the largest influence over the product. This makes the baseline stakeholders the most important stakeholders. Furthermore, Sharp describes that this group of stakeholders consists of four categories of people: users, which could exist out of primary, secondary and tertiary users, developers, which are building the product, legislators, which could affect the product by rules and regulations, and decision-makers, which have mostly the largest influence within affiliated parties. Besides baseline stakeholders, other stakeholders are identified as well: clients, suppliers and satellites. Where clients are relevant customers of the involved parties, suppliers are the relevant deliverers of resources for these parties. Satellite stakeholders are people that interact with the baseline stakeholders and are of any relevance concerning the product.

The stakeholder identification is performed within the same environment in which the prototype is evaluated, the factory of Apollo Vredestein. According to the maintenance expert Leon Pelders, this factory is a representative environment for other factories in the Netherlands. Information about organizational structure is obtained from an interview with the same expert.

2.3.2 User & Expert Interviews

Multiple interviews are conducted during several phases of the project. These interviews were performed with the following people: an expert in factory maintenance, a technician participating in the design process, a technician without pre-knowledge of the project and a team leader. The last three people were employees of Apollo Vredestein. Since all interviews included an explorative character, the interviews were semi-structured. This means that, in preparation of such an interview, a number of topics are chosen to be discussed. Although these topics are mentioned and evaluated during the interview, additional space is available for the interviewee to bring up new topics. Goal of these interviews is to obtain information about the Apollo Vredestein environment, the technicians' way of working and to examine the validity of researcher its conclusions.

A number of interviews are conducted as informal interviews, as described in [17]. Where traditional interviews include meetings with participant for a pre-defined amount of time, informal interviews are unstructured and performed in the form of informal discussions. The goal of these informal interviews is both gaining trust with the participants as well as the researcher being as open as possible to all comments of its target group. Especially in situations in which as much information should be obtained as possible, like during observations, this open character is important.

2.3.3 Observations

To get to know the technicians' needs it is important to understand their goals and, therefore, the daily practice of this group. In literature, many ways are described to obtain information of a way of working, like interviews, observations and surveys. Observations are chosen to be performed during the ideation phase for understanding the technicians' way of working. Surveys are tightened in their questioning and therefore a better method when the researchers are known to the daily practices of the participants. Although interviews are assessed as a better method for understanding practices, participants of interviews tend to forget telling specific details or possibly remember things differently than these in reality are [10]. Additionally, it is possible that participants of interviews intentionally tell things differently to mask certain topics. Furthermore, observations are a method that is used frequently for generating new ideas and obtaining insights in new subjects. A combination of observations and evaluating these observations during interviews will include all advantages of both methods. Within this research, these methods are performed during the ideation phase.

According to [10], three kinds of observation methods exist: controlled-, naturalistic- and participant observations. Controlled observations are regulated by the researchers, meaning that the participants are observed in an environment that is equipped by the researchers on pre-determined actions. Since these observations fits bets within a developed phase of a study in which it is clear which actions should be observed, this method does not fit the early stage in which the observations are performed in this study. Participant observations are a research tool in which researchers are fully engaged in the group that they are studying, joining the group in their daily activities to obtain a deep insight in their lives. Since participant observations require deep knowledge about the environment of the group that will be observed, this method is too time consuming for this study. Naturalistic observations, in which the researchers observe a group within the group's environment without pre-knowledge, fits best to the conditions of this study.

Because naturalistic observations are often conducted with a small amount of participants, this method possibly lacks a representative sample. Additionally, naturalistic observations possibly are less reliable than controlled observations, since variables cannot be controlled by the researchers. However, [10] describes naturalistic observations to be a suitable method for generating new ideas.

Most of the naturalistic observations studies tend to be executed in a typical procedural arc, consisting of four phases: the descriptive phase, the focusing phase, the selective phase and the saturation point [11]. Since the scope of the observations of this study is to explore the technicians' way of working generally, only the first two phases will be performed. The descriptive phase includes descriptive reporting, in which the researcher writes down as much information as possible, without a specified focus and without making assumptions to what is relevant or how things should be interpreted. Remarkable observations from the first phase are used as factors that are observed with higher interest in the selective phase of observations. However, the researcher remains considering how these factors fit into the larger context. In the third phase, the selective phase, the focus is narrowed to only the focus points and the researcher tries to deepen his knowledge about these points as much as necessary. A saturation point is reached when new observations by the researcher only lead to conclusions that are already drawn by observations done earlier, and no new findings will appear. It was decided to perform a two day observation of the technician's work in the factory of Apollo Vredestein. Where the descriptive phase is executed on the first day, the researcher executed the selective phase on the second day of observations. Input for this phase are focus points that are observed in the first phase.

Elaborating on the quality of the observations, validity of the results are of importance. This means that what is observed also happens in reality. According to [11], researchers triangulate their research methods and conduct the observations with multiple observers. Although the resources of this study are making it impossible to perform the observations with multiple observers, it is possible to vary in methods while observing. Therefore, the observations are always conducted in combination with multiple informal interviews with the technicians. These informal interviews are conducted as described in [17].

2.3.4 PACT Scenarios

According to [20] a scenario is a proper tool for discussion since they represent concrete situations and people find it easier to reason and talk about concrete situations. PACT scenarios are written during the ideation phase of this project. The conclusions from the observations of the technicians at Apollo Vredestein are used as input for the PACT scenarios. Goal of these scenarios is to elaborate on these conclusions, and to combine these in user stories. [20] describes user stories as "the real or imagined experiences of people; what they do and what they want." These scenarios sketch the current way of working of the technicians without the help of an augmented reality product. [20] argues for using the PACT framework in order to obtain more specific context to the scenarios. Within this framework it is important to think about people (users), activities, context and technology first, before writing the full scenario. During the specification phase, designer perspective scenarios are created that were used as a starting point for prototyping and the evaluation of design ideas. These scenarios are to the point and clear about which design decisions are not yet made, and describe the way of working of the technicians including the help of an augmented reality product.

2.3.5 Product Requirements & MoSCoW

In order to concretize the product idea that is formulated in the designer perspective PACT scenarios, product requirements are listed describing the products functionality. In order to include participatory design within this process, these requirements are evaluated and prioritized with the end-user and the maintenance expert. The product requirements are divided in two sections: functional and non-functional requirements. Since the focus of this research is on functionality rather than effectivity of the product, the list of functional requirements is more comprehensive than the list of non-functional requirements.

Because the time limit of this project, not all functionalities can be included in the prototype that will be evaluated in the last phase of the design process. To build a prototype that fits best to the user needs, the most important functionalities should be constructed first. Therefore, the list of product requirements is prioritized using the MoSCoW method that is frequently used for IT projects. In this method, all requirements are divided in one of the four categories ‘Must haves’, ‘Should haves’, ‘Could haves’, ‘Won’t haves’ [24]. Goal for this project is to build all ‘Must haves’ and ‘Should haves’ in the first prototype, which will cover all essential functionalities of the product idea. Product requirements that are prioritized differently, are not fulfilled during this project.

2.3.6 Evaluation

The goals of the evaluation are to examine wether the goal stated at 2.3.5 is achieved, and to experiment with the prototype in a real use situation. The experiment examines whether the functionalities that are expected to be useful appear to remain meaningful in the way of working of the users. [26] argues that in a participatory design project “ideally, a project outcome should be evaluated in a real use situation, when users have had a chance to integrate it into whatever they are doing and (eventually) develop new forms of practice.” However, for this project not enough resources are available to let the users integrate the technology into their own work. Additionally, the prototype is only ready for performing the three designer perspective scenarios and not for a pilot study. Therefore, it is chosen to evaluate these three user scenarios in which the prototype is expected to be of an advantage, in situations that are previously observed in the way of working of the technicians. After this experiment with end-users, the participating technicians are asked about their experiences with the product. This evaluation includes questions addressing ‘user gains’ [26]. These questions can be indirect, considering whether participants have the feeling that they are heard in the design process, as well as direct, considering whether personal skills and quality of work will improve when using the product in their daily work.

In order to met the two goals, the evaluation consist of two phases:

1. Functional testing, including an internal evaluation and an evaluation with the technician that contributed in the participatory design. Within the internal evaluation, all MoSCoW product requirements are tested with the prototype. Afterwards, a semi-structured interview with a technician is executed considering his expectations towards the prototype, and whether these are met.
2. User evaluation: an experiment in which technicians use the prototype in three situations that come very close to a real use situation. In this phase, the end-users are asked to use the product with pre-defined tasks. These tasks should correspond with their daily work, and therefore should correspond with the user stories and the designer perspective scenarios. After the experiment the users are asked about both direct and indirect ‘user gains’ as well as usability and learnability of the product.

3. State of the Art

Before starting with the ideation phase, an understanding of context around the research topics should be obtained by the researcher. Therefore, a brief literature study towards the status quo of augmented reality is performed first. The definition of augmented reality, how this technology has evolved during the last decades and what is generally expected from AR in the future, is described in section 3.1. To make a decision on what technological device to use for the AR product later in this study, the status quo of AR head-mounted display is researched (see section 3.2). Last, related work is found in the domain of augmented reality for technicians and described in section 3.3. This will prevent an AR product as outcome that is already available on the market.

3.1 Technology

Within this paper, augmented reality is defined according the definition of [1] "as a real-time direct or indirect view of a physical real-world environment that has been enhanced / augmented by adding virtual computer generated information to it." Although many different ways to create an augmented reality system exist, all of them share the same three properties [2]:

- augmented reality combines objects from the real world with virtual objects and places these virtual objects in the real environment;
- includes an interactive, real time experience; and
- places the virtual objects in alignment with the real world objects.

In the reality-virtuality continuum of Milgram [8], displayed in figure 3.1, it can be seen how augmented reality is related to other definitions in this continuum. The two opposites in this continuum are the real environment and and a fully virtual environment. The real environment, which requires no technology to be seen by people, includes no virtual objects. The fully virtual environment, which requires virtual reality technology to be seen, includes no real objects. The term "mixed reality" includes an environment in which both virtual and real objects are combined. Although virtual reality has many appearances, this technology is generally known for the head-mounted devices like the HTC Vive and the Oculus Rift. These devices are being carried on the head of the user and includes two displays that cover the full view of the user. Virtual reality is regularly used for gaming purposes, but also offers solutions in more 'serious' fields like military training, surgery simulation and architecture. Augmented virtuality is an environment that is fully virtual, including objects that originate from the real world. A good example of a situation in which augmented virtuality is used is a presentation of a weatherman in front of a chromakey[1]. With the use of this technology, the weatherman is cut out of the camera view and is pasted in a virtual environment, mostly in the form of a map including weather information. Since the environment, the virtual map, is computer generated and objects within that environment are both virtual or originating from the real wold, the weatherman, this technology falls into the domain of augmented virtuality.

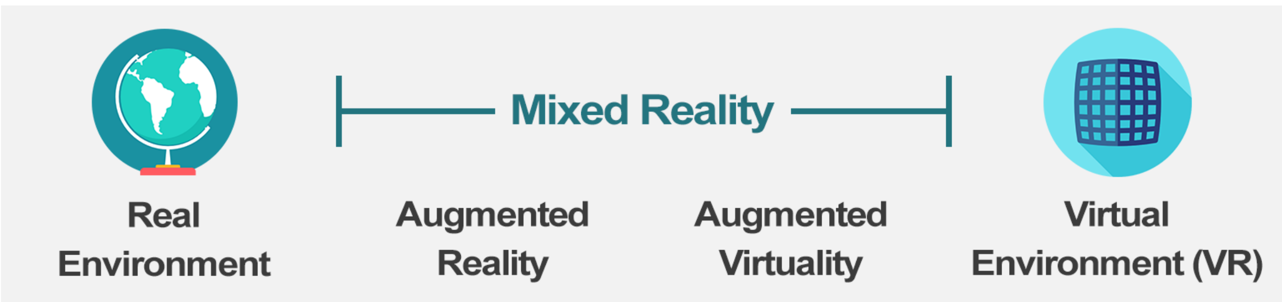


Fig. 3.1 - Milgram's Reality-Virtuality Continuum

[1] "Chroma keying, is a visual effects / post-production technique for compositing (layering) two images or video streams together based on color hues (chroma range)." [30]

Although AV and VR are both interesting research topics, this paper goes deeper into the literature of augmented reality. As mentioned earlier, this technology uses a real world environment and enhances this world with computer generated objects that are placed in alignment with the real world objects. Examples of AR technology can be found in all sorts of domains: education, commerce, entertainment, medical science and transportation [1].

One of the earliest mentions of an augmented reality system, in the very broad definition of AR being technology combining real world and non-existing experiences, is a device named Sensorama that is described in the paper “The Cinema of the Future” by Morton Heilig, already in 1955 [16]. Morton is a multimedia specialist that used his knowledge as a cinematographer to create a device that tried to increase the experience of “standing inside the scene” by exciting more senses than only sight and hearing. According to [1], the first head-mounted display is invented by Ivan Sutherland in 1966. A head-mounted display (HMD) is a device that is meant to be put on the user’s head providing an AR, AV or VR experience to the user with the use of internal displays. Earlier mentioned examples of HMD’s in the field of VR are the HTC Vive or the Oculus Rift. From both the Sensorama and the head-mounted three dimensional display from Sutherland it can be concluded that people are experimenting with the augmented reality technology for over half a century. However, in [2] is written that this technology started making fast progress only from the 1990s. From these years, augmented reality and head-mounted displays have developed with large steps. This can also be concluded when comparing two studies about the status of augmented reality at their time, with one published in 2001 [2] and the other study published in 2010 [1]. Where in 2001 much is written about the limitations of technology, in 2010 the major limitations seems to be in the fields of social acceptance and privacy. In addition, since 2010 many has changed in the development of augmented reality devices, especially with head-mounted displays. Since 2010, Google has experimented with the Google Glass, which is a HMD able to show live content in the upper right corner of the users view, Epson developed HMD’s for businesses able to show virtual objects in the center of the users view that are aligned with the real world objects, and Microsoft has launched the Hololens in 2016, which is an HMD that is able to scan the users environment and place virtual objects over a part of the view of the user.

The evolution of augmented reality through the past decades is very promising, and therefore it can be very interesting to start developing with this technology in any domain. This is strengthened by calculations of Digi-Capital predicting a growth of the AR market from $3.9 billion in 2016 to $82 billion in 2021 [23]. Also the Global Market Insights expect such exponential growth and predicts that “the global market for AR products will surge 80 percent to $165 billion by 2024” [36]. Reasons for these predictions are leading technology firms announcing to start working intensively with AR in the near future. According to [36], “Apple has embarked on an ambitious bit to bring the technology [augmented reality] to the masses” and that an augmented reality device “may even supplant the iPhone.” In April 2017, Facebook announced to launch “the first mainstream augmented reality platform” within their services [37]. Although Google’s experiment with the Google Glass has stopped, they invested $793.5 billion in the augmented reality start-up ‘Magic Leap’ [38]. Other examples of augmented reality, especially found in the field of maintenance engineering, can be found in section 3.1.3, discussing related work. In the next section, available technology in the field of HMD’s will be discussed.

3.2 Head-mounted Displays

This section is meant to give an insight on the actual status of augmented reality HMD’s that are commercially available at this moment. This can become important when, later in the research, a platform for the augmented reality solution will be chosen.

Google Glass

As written in the previous section, Google started one of the first augmented reality HMD of which information was widely spread. Although their HMD, called the Google Glass, was meant for developers, it was eventually meant to bring a developed version of the Google Glass to the consumer market. This glass included a prism in the upper right corder of the user’s view on which virtual objects are projected (see figure 3.2). Although the Google Glass is not produced anymore, there are speculations say the company is working on new augmented reality technology. [31]

Microsoft Hololens
In 2016, Microsoft launched a first developer edition of the Hololens. Of all augmented reality HMD's, it is considered as one of the most promising. This is caused by the fact that it is running without being wired to a computer, that it has four cameras for tracking and the virtual objects are placed realistically in the user's view. The device should be operated with the use of hand gestures. In addition, a large number of companies, mostly within the domain of engineering and designing, have announced a collaboration with Microsoft for developing and using Hololens software. However, this technology has its disadvantages: it is a relatively heavy and large device compared to other HMD's. Also, it is only possible to see the virtual objects when these are in the center of the user's view [32]. An image of the Microsoft Hololens is given in figure 3.3.

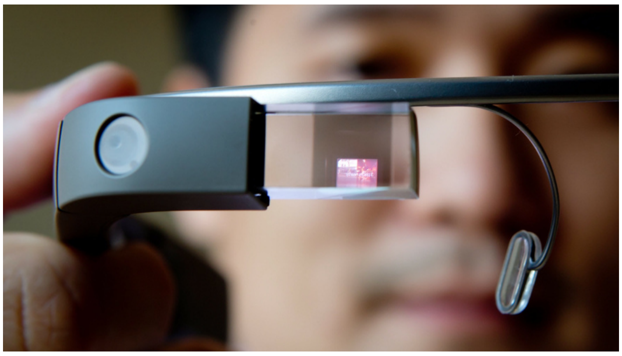


Fig. 3.2 - Google Glass

Fig. 3.3 - Microsoft Hololens

Epson
The company Epson is making augmented reality glasses mainly for businesses. It has already two glasses on the market: the Moverio BT-100, and the Moverio BT-200, which is a developer version. The successor of these glasses, the Moverio BT-300, has been launched in the spring of 2017 (see figure 3.4). All Epson devices glasses work with Android, and should be operated with an additional trackpad device. [33]

Vuzix
Vuzix has one glass for businesses on the market, called the M100. These glasses are enhanced with a small screen in the upper-right corner of the user's view, like the Google Glass. The company is planning on launching two new device for enterprises, the M300 and the M3000. Last, Vuzix is experimenting with a glass in which holograms are projected on the user's view, close to the experience that the Hololens can provide (see figure 3.5). [34]

Fig. 3.4 - Epson Moverio BT-300

Fig. 3.5 - Vuzix AR3000 smart glasses

Daqri
In the field of industry, the company Daqri is working with both a smart helmet and smart glasses, both providing augmented reality experiences [35]. Both products are in development phase, but

developer kits can be ordered. Images of these technologies can be found in the figures 3.6 and 3.7.

When looking at product video's and reading documentation of these augmented reality HMD's available it is clear that most of AR HMD's are under development. Although there are some glasses that are fully ready, most glasses can only be ordered as a development kit. In addition, the HMD's that are ready, are only meant for business purposes, not for the consumer market. Also, these HMD's only offer an augmented reality experience that is not covering a large part of the user's view. Although this suggests that the technology is not ready yet for consumer production, it can be interesting for businesses to start experimenting with the available HMD development kits. An overview of the researched augmented reality HMD's can be found in table 3.1.

Fig. 3.6 - Daqri Smart Helmet

Fig. 3.7 - Daqri Smart Glasses

HMD	AR technology	Operating System	Price
Google Glass	Prisma in upper-right corner	Android	N/A
Microsoft Hololens	See-through holograms projecten on center of view	Windows 10	$3000
Epson Moverio BT-100	N/A	Android	$699,99
Epson Moverio BT-200	See-through holograms projecten on center of view	Android	$699,99
Epson Moverio BT-300	See-through holograms projecten on center of view	Android	$779,99
Vuzix M100	Display in upper-right corner	Unkown	$900
Vuzix M300	Display in upper-right corner	Unkown	N/A
Vuzix M3000	See-through holograms projecten on center of view	Unkown	N/A
Vuzix Blade 3000	See-through holograms projecten on center of view	Unkown	N/A
Daqri Smart Helmet	See-through holograms projecten on center of view	N/A	N/A
Daqri Smart Glasses	See-through holograms projecten on center of view	N/A	$4.995

Table 3.1 - Overview of AR HMD's available or planned for launching, prices as of 1/12/2016

3.3 Related Work

Fieldbit

The Israeli company Fieldbit is in the development phase of making a augmented reality product, meant for the industrial industry. Although a small amount of information is available on their new product, Fieldbit's website is able to explain the product idea [12]. The company focusses on increasing the success rate of technicians repairing a machine at the first time they are visiting that machine, also called the "first-time fix rate". Fieldbit believes in external guidance from an expert located in a service center. When the technician starts with a repairing task, contact can be established with an expert via an Epson Moverio BT-200 smart glass. These glasses are able to project an augmented reality layer in the center of the technician's view, and is powered and controlled by a mobile touchpad. The technology makes it possible that the expert receives the camera view of the smart glasses to which the expert can react with advise and instructions to the technician. The advise or instructions can be communicated over audio, but also by the expert drawing instructions like circles and arrows in the augmented reality view of the smart glasses worn by the technician. Next to contact with an service center, the technology also enables communication with co-workers or supervisors via the smart glasses. Lastly, it is possible to connect with enterprise systems. However, the exact workings of this functionality is not clear. An overview of how these functionalities of Fieldbit's technology works can be found in figure 3.8.

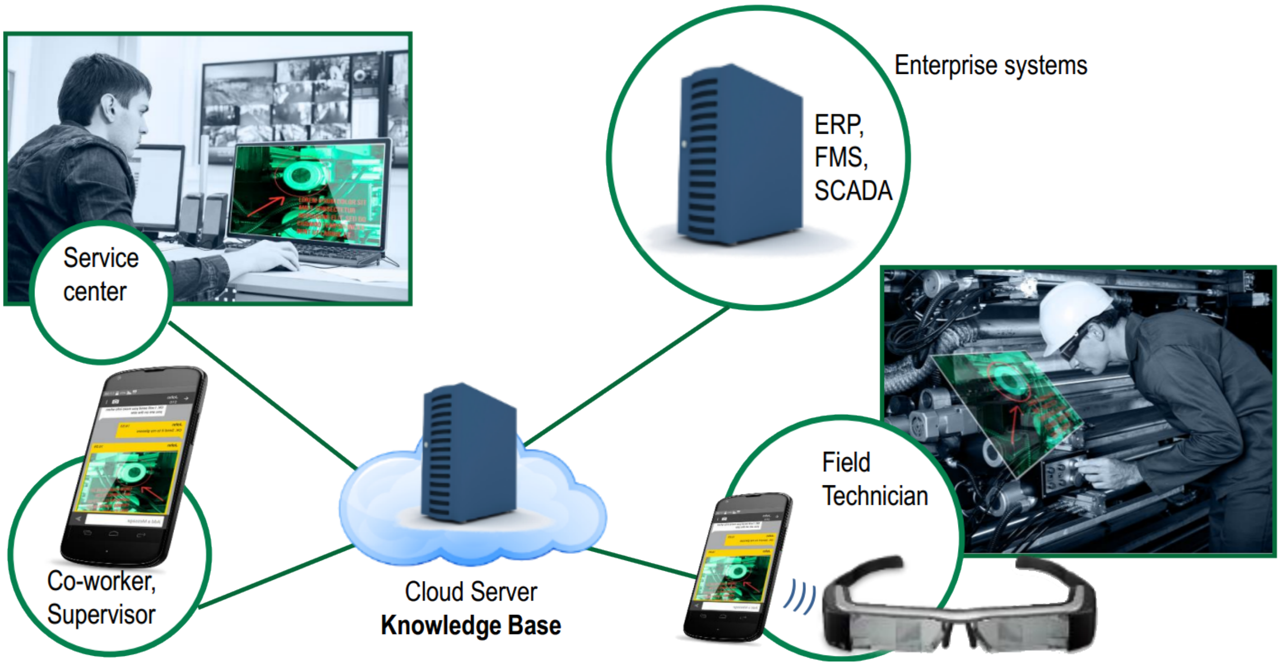


Fig. 3.8 - Overview of all functionalities of Fieldbit's technology

Hyperindustry by Inglobe Technologies

In collaboration with Huawei, Inglobe Technologies has performed a case study to augmented reality instructions for Huawei Technologies' technicians [9]. Researched is the improvement in efficiency of the technicians work when carrying out common tasks with the use of an augmented reality tool. The tool is a mobile application designed for Android and iOS smartphones and tablets and is able to recognize targets due to its 3D tracking technology. In this case, the target is always a SUN2000-KTL inverter, which is a device that mainly converts DC power from photovoltaic power systems on commercial rooftops. After a target is recognized, a list of available procedures are displayed to the technician. Additionally, a maintenance manager can assign specific tasks to the technicians. When an technician starts a task, the application displays a series of steps, including instructions and augmented reality elements that can be highlighted points on the target or procedural animations (see figure 3.9). Secondly, the application is able to show additional contextual information, such as technical documents, photos and videos. Last, the technician's performance is recorded and send to the maintenance managers for analysis. Outcomes of the case study included the finding that "compared with the traditional paper-based or digital instructions, one of the most obvious benefits […] is the technician's immediate understanding of each step in the operation." This outcome is possibly both helping the execution of procedures by new technicians, but also helping the experienced technicians in resolving occasional and complex issues.

Fig. 3.9 - An augmented reality animation overlaying the SUN2000-KTL

CN2Tech

The American company CN2Tech offers augmented reality solutions in three different products: Insight, Assist and Experience [13]. From CN2Tech's products, "Assist" and "Experience" are most relevant to this study, since they offer solutions for an industrial environment. "Insight" is meant for companies sales and makes it possible that customers of those companies can try products in their own environment, via augmented reality on a tablet. For example, a customer could try whether a set of tables and chairs from a furniture shop fits into its living room. The product "Assist" is a combination of the earlier mentioned Bitfield's and Inglobe Technologies' products: a tablet that enables a connection to an expert in a service center. The user then can ask assistance from the expert, who can draw instructions like circles and arrows to the tablet screen of the user. These instructions will overlay the camera view, and will maintain aligned with the objects in the real world. An example of this is displayed in figure 3.10.

The product "Experience" is taking a paper user manual to the next level. It provides 3D augmented reality instructions telling users of a machine step-by-step which actions should be performed. An example that CN2Tech is providing on their own website is the task of putting a new cartridge in a printer. When the user hovers the tablet over the printer with CN2Tech's application, the software will display these instructions on top of the camera view of the tablet. In addition, the application knows when the step is executed and automatically displays the next step to the user.

Fig. 3.10 - CN2Tech "Assist": an expert provides augmented reality instructions to the user

Japan Airlines

According to one of Microsoft's blog writers Suzanne Choney, Japan Airlines has created two proof-of-concepts that are developed using the Microsoft Hololens, with the goal to train technicians more effectively [14]. Where the company worked with documents based on text and pictures in the past, Japan Airlines is now experimenting with a new way to train the technicians with augmented reality. A training with a real plane often results in waiting for an appointment, an available location or when a plane is placed in the hangar for maintenance, which is time consuming. Japan Airlines designed two training tools. First, a virtual plane engine floating in the users environment. Secondly, a virtual cock-pit that can be operated using voice and vision. Because the Microsoft Hololens is a head-mounted display that is able to overlay augmented reality objects over the full vision of the user, it is possible to place a plane engine anywhere in the surrounding, with an adjustable size and a possibility to walk around the engine entirely (see figure 3.11). In addition, the software enables the user to remove parts from the engine in order to see the parts that lay behind.

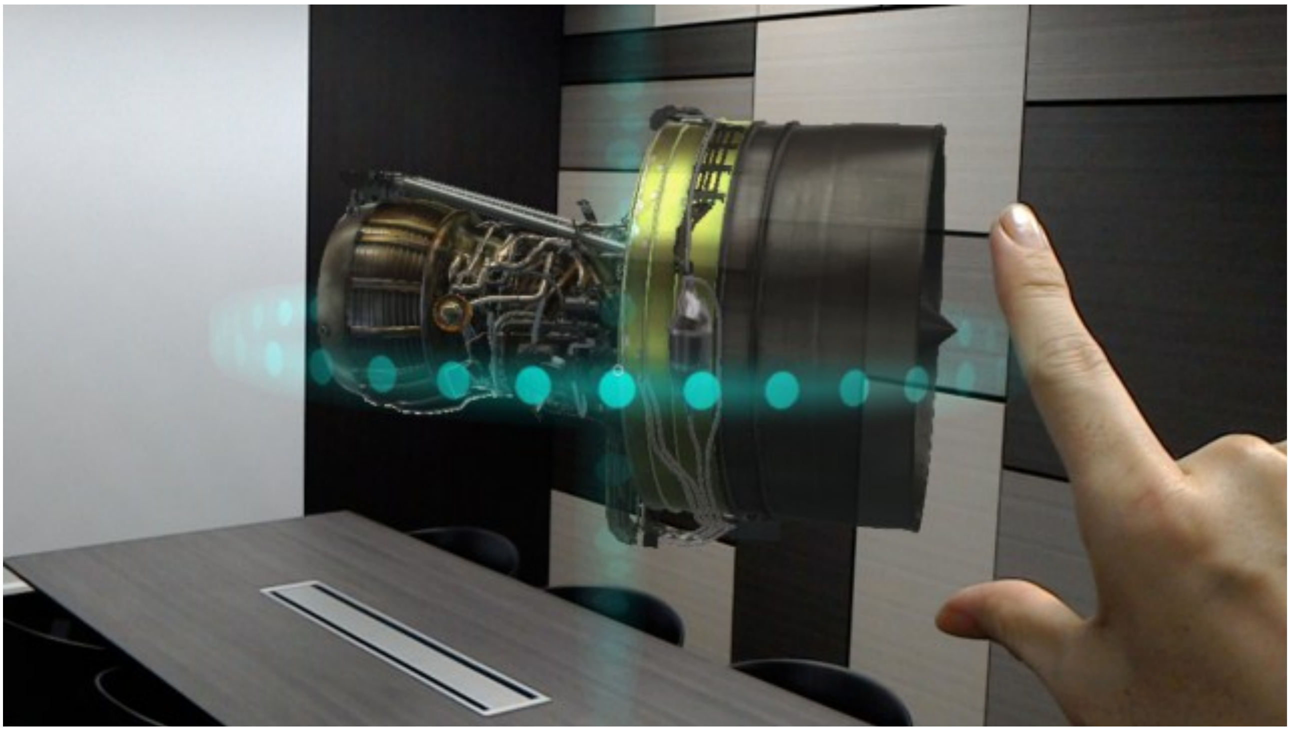


Fig. 3.11 - One of the two proof-of-concepts of Japan Airlines: a 3D hologram of a plane engine

Augmented Reality for Maintenance and Repair (ARMAR)

At the Columbia University, Henderson and Feiner are working on their ARMAR project in which the potential is researched for the use of augmented reality to support technicians in the performance of maintenance and repair tasks [15]. Their experiments include real time computer graphics, overlaying the physical view of the user via head-worn motion tracking displays, that should "[…] improve the productivity, accuracy and safety of maintenance personnel." These graphics includes "[…] sub-component labeling, guided maintenance steps, real time diagnostic data, and safety warnings". An example of how such a set-up looks is given in figure 3.12. Within this figure, the message in the bottom left corner is warning the user for danger. In addition to the earlier mentioned advantages of augmented reality to the work of a maintenance engineer, the software preventing the user from causing harm is potentially an important aspect to the use of augmented reality.

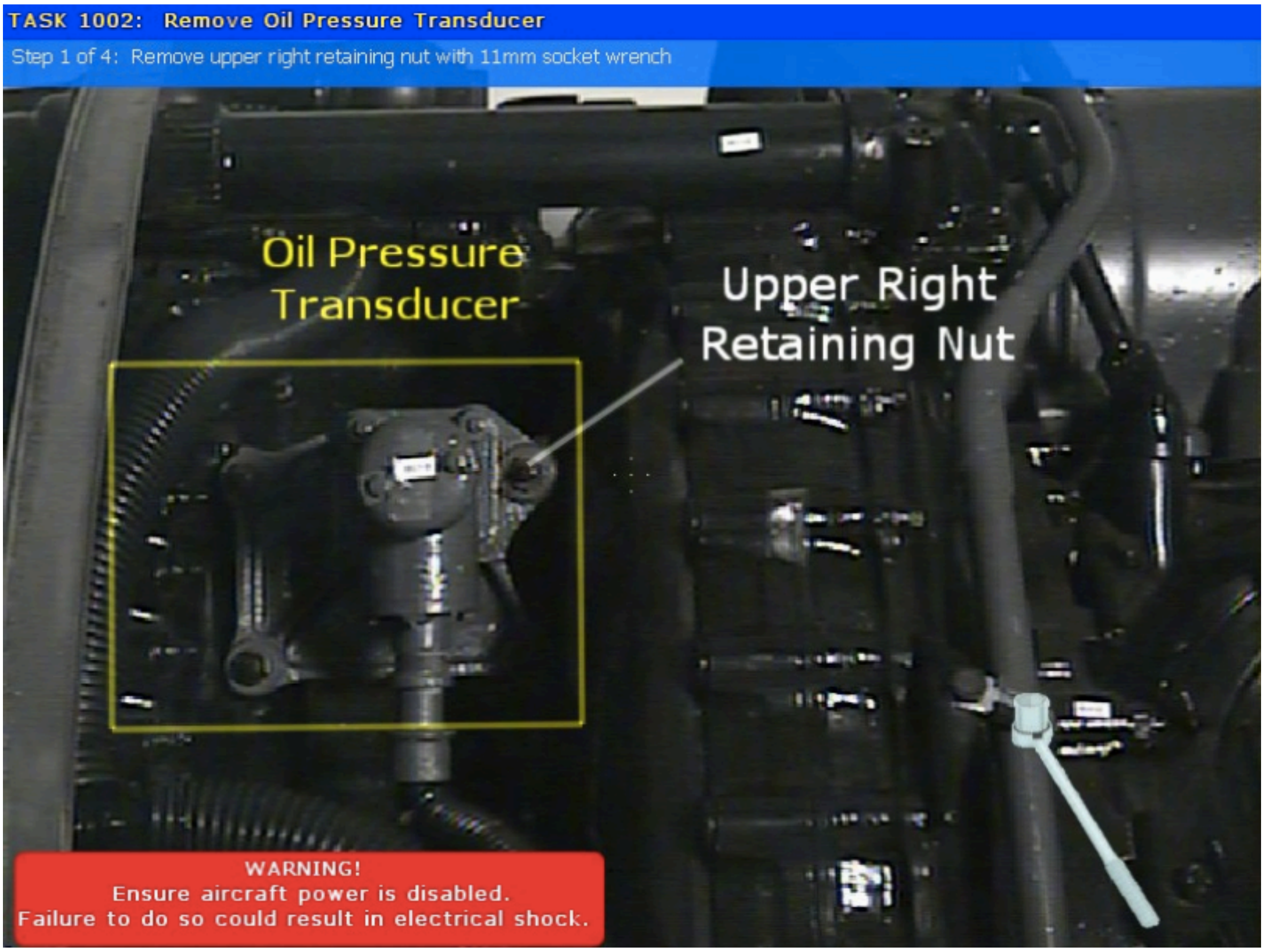


Fig. 3.12 - An impression of one of the experiments within the ARMAR project

In the related work of other companies and researches that have created an augmented reality product for the maintenance and repair domain, three different existing concepts can be distinguished. This answers the first subquestion of this study. The three concepts are explained below.

1. Training of an technician with the use of interactive 3D models that can float in space. The technician gains a quicker understanding of a machine, since the technician can place the machine or parts of the machine in the surrounding space, walk around the machine and even take specific parts away from the machine to be able to see the inside.
2. An augmented reality instruction, that provides the technician with information on what to do step-by-step. Keeping the products of CN2Tech, Fieldbit and Inglobe Technologies in mind, this can be reach in two ways:

a. Automatically; the software recognizes the user's surroundings or a device in front of the user, and immediately displays an instruction for the next step on top of the technician's view.
b. With the help of an expert, possibly working at a service center, who knows what task the technician would like to perform. This expert gives advice and instructions by drawing on the camera view of the technician. These drawings occur as augmented reality overlay to the view of te technician.

3. Software, recognizing the technician's surroundings or nearby devices can warn the technician for possible causes of accidents.

3.4 Tip4Support

In order to come up with an augmented reality concept that is not only fitting the technicians' needs, but also fits the related product (named Tip4Support) of Innovadis, it is important to understand reasons behind this product and to understand how this product will be build in the future. Together with the maintenance expert Leon Pelders and the company Eluxis, Innovadis started this project in September 2016. How the product idea of Tip4Support is established, is described below. This information is gathered during meetings with people from all three companies and interviews with the maintenance expert.

3.4.1 Product Description

PISA-Activities

In the past century, factories have become a place of large machines that are installed for very specific actions. These machines require maintenance and repairs in order to extend their economic lifetime. These maintenance-activities are described as "PISA-activities", referring to "Poetsen, Inspecteren, Smeren en Afstellen". Especially for companies producing for the food-industry, it is important that machine inspections are performed frequently and carefully, since these companies should guarantee the safety of their food. Pelders Maintenance Management is working with an spreadsheet-based system to provide lists including PISA-activities, which should be completed. These lists are given to operators and technicians so that they know which actions they should perform, including a small description of the actions.

Machine Documentation

In order to perform the maintenance actions correctly factories have extended documentation describing their machines in detail. Operators and technicians use this information to work more efficiently and to decrease the chance of making mistakes. Eluxis is creating documentation and instruction-content for factories. Their products varies from instruction-video's to detailed descriptions of machines' components. Clients of Eluxis are mainly using this documentation for training purposes. However, this information can also be used during work operations.

Product Specification

Both products from Pelders and Eluxis are successful. However, these products are based on paperwork. Although Eluxis has an online structure for their documentation, there is no intuitive platform for the operators and technicians to access this information quickly. The documentation is only accessible in a folder-structured way, meaning the employees need to search for every document making this a time consuming activity. Although the activities-lists of Pelders are accessible via Excel-sheets, the operators and technicians has to print the lists in order to work with them. Therefore, both companies see a way to improve their products with a digital platform. This platform displays which tasks a user should perform, how these tasks should be executed. Additionally, monitor functionality is included for team leaders and managers to improve the efficiency of their operators and technicians. With the knowledge of Innovadis on building business software, they see a way to combine knowledge in one product.

4. Ideation

The ideation phase is present with the goal to obtain a project idea, that serves the target-users' needs. Since the specific needs, but also the target-users were not given at the start of this study, these were explored first. Future users of the system, together with other stakeholders, are identified in section 4.1. Afterwards, the way of working of the target-group is studied in section 4.2. From these observations, three PACT scenarios are filtered that describe the way of working (section 4.3). The augmented reality product should improve the way of working of the technicians in these three scenarios. In section 4.4, four potentially interesting project ideas are described. From these four projects, one project is chosen to work with during the next project phases. This idea is described and visualized in section 4.5.

4.1 Stakeholder Identification

4.1.1 Stakeholders

As described in the method section of this research, methodology of Sharp is used for stakeholder identification. Since the augmented reality product must have an opportunity to be implemented in the Tip4Support application, all stakeholders are identified in the hypothetical situation that the product is implemented in Tip4Support and frequently used at the factory of Apollo Vredestein.

All stakeholders identified as baseline stakeholders are given in table 4.1. It can be seen that the product has three potential users: technicians, operators and team leaders. In collaboration with Leon Pelders, the technicians are chosen as primary user group of the augmented reality product. This makes the technicians the target group for this project. Since operators will also be users of Tip4Support, it is expected that they will be able to use functionality of the augmented reality solution a later stadium. This makes the operators the secondary user group, which is left out of scope. Team leaders are expected to use Tip4Support to supervise the technicians and operators. Since supervising is also outside the scope for this project, and team leaders are expected to use the product less frequently than technicians and operators, they are identified as the tertiary user group.

Besides users, the baseline stakeholder group is filled with developers, legislators and decision-makers. It is important to identify the requirements and needs of these as well. However, since this project has the focus on a prototype that is working for the target group, these activities should be performed after this project. Additionally, because this project is focussed on the interaction between an augmented reality product and technicians, even further stakeholders like suppliers, clients and satellite stakeholders have not been investigated.

	Role	Company
Users	Technicians (primary)	Apollo Vredestein
	Operators (secondary)	Apollo Vredestein
	Team leaders (tertiary)	Apollo Vredestein
Developers	Programmers	Innovadis
	Designers	Innovadis
	Testers and maintainers	Innovadis
	Programmers	Eluxis
	Maintenance expert	Pelders Maint. Management
	Project management	Eluxis, Innovadis

	Role	Company
Legislators	Government	-
	Safety executive	Apollo Vredestein
	Quality assurance	Apollo Vredestein
Decision-makers	Executive board/management/CEO	Eluxis, Innovadis, Grolsch
	Process manager	Apollo Vredestein
	Maintenance manager	Apollo Vredestein

Table 4.1 - Baseline stakeholders, divided in four categories

4.1.2 Organizational Structure

The stakeholders that are identified in table 4.1 can be placed in an hierarchal structure. This will gain knowledge on how all stakeholders in the factory of Apollo Vredestein are connected. A structure of how an organizational structure of a factory generally look like is given in figure 4.1. This includes the hierarchal structure of Apollo Vredestein. Descriptions of these stakeholders are given below.

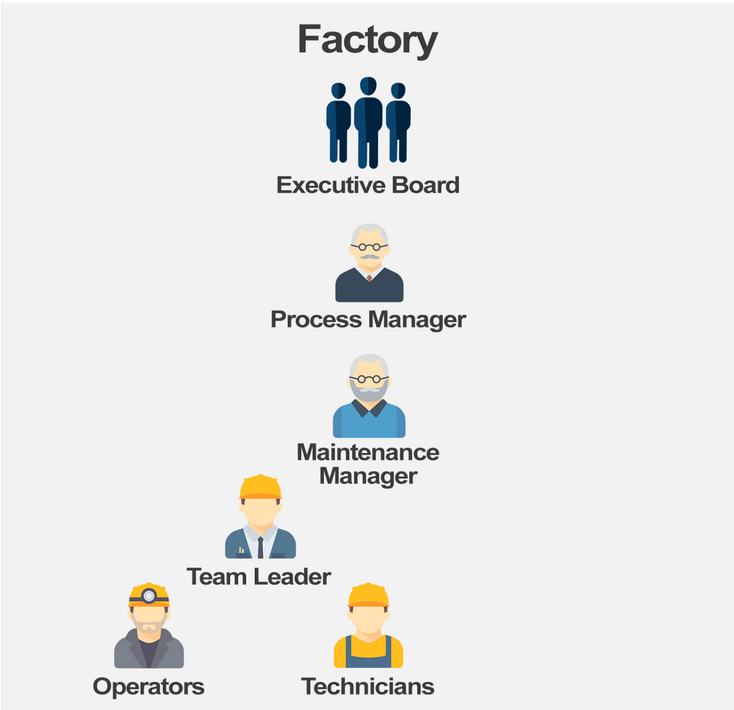


Fig 4.1 - Example of an organizational structure of a Dutch factory

Executive Board, Process Manager and Maintenance Manager
The roles of executive board, process manager and maintenance manager are on the level of policies and regulations. However, they all have a different amount of distance to the people carrying out the work, which are the operators, technicians and team leaders in the maintenance domain. The executive board has the largest distance, but has most influence on policies and regulations, since these cover all domains in the factory. They make financial and strategic decisions for the company. In the factory, multiple process managers are present. All of these have a different "process" or domain to which they make decisions on policy and regulations. An example of a domain is work-processes. Closer to the people carrying out the work in the maintenance domain is the maintenance manager. This function is responsible for the durability of the machines in the factory.

Operators, Technicians and Team Leaders
Teams of operators and technicians are formed with the task to execute the maintenance activities that they are receiving. The operators are responsible for the “first line maintenance” of the machines in the factory, including PISA-activities and production. This means that they are frequently inspecting the machines, set these machines in the right positions and keep them lubricated. When they see an error during the inspections, they send a request for repair to the technicians or, if possible, they fix the issue themselves. The technicians are responsible for repair of the machines when errors are detected, but also for planned maintenance and repair. Teams of operators and technicians are instructed by a team leaders or supervisors, which are monitoring the performance of their teams.

4.2 Observations

4.2.3 Observations of Technicians at Apollo Vredestein

Two days of focusing observations are performed at the factory of Apollo Vredestein. Mainly technicians, but also operators, have been observed in their way of working. During the analysis of the observations, all observations are divided between eight categories. In table 4.2, the categories are displayed together with a brief explanation. From the descriptive phase of observations that is executed on the first day, seven focus points are addressed. These focus points serve as input for the selective phase of the observations, executed on the second day. Beneath, the results of the observations at Apollo Vredestein are discussed. General observations are mentioned first, while the focus points are discussed afterwards. The full list of observations are given in appendix A.1.

Category	Explanation
Environment	All observations addressed to the direct environment of the technicians are categorized here. In this study, the environment is the factory.
Tools	All observations addressed to the tools and materials that the technicians are using to perform their daily tasks are categorized here.
Technician	All observations addressed to the technicians itself, e.g. clothing, are categorized here.
Actions	All observations addressed to specific actions that the technicians perform during their work are categorized here.
Learning	All observations addressed to the way that technicians learn about their job are categorized here.
Information sources	All observations addressed to sources that the technicians use to inform themselves to perform their work are categorized here.
Technology	All observations addressed to technology, e.g. iPads, are categorized here.
Communication & planning	All observations addressed to communication between people in the factory, or towards the planning of maintenance work, are categorized here.

Table 4.2 - Categories of the observations at Apollo Vredestein

General observations
The observations are performed within the extrusion department of the factory, which is one of the middle processes in the manufacturing process of tires. In this department different layers of rubber are combined and processed. Result of this process are materials that could be formed in to different kinds of tires. Machines within this department can be old, originating from 1950 or 1970, but are mostly build a couple of decades from now. Also newer machines are used within the extrusion-department. The technicians that were observed and interviewed worked at the ‘Quadruplex’ machine. This name refers to the working of the machine: extrusion with multiple

layers of rubber, with a maximum of four layers. It is a machine of approximately 60 meters, build around 1990. Like other machines in the factory, two groups of people are working with the machine: operators, responsible for operating the machine when producing, and technicians, responsible for maintenance and repair. The maintenance tasks can be both scheduled and incidental. Traditionally, two types of technicians exist: electricians and technicians. During the week, the technicians can work on two types of work. First, in production-shifts that include reactive maintenance on any machine. Second, in a PIPO (also called PISA) that includes planned maintenance to one or multiple machines. For example, a PIPO is executed at the Quadruplex every two weeks. In the factory of Apollo Vredestein, all machines have one expert-technician, which is the 'emotional owner' of the machine. This expert has extended knowledge about the machine and is responsible for assigning tasks to technicians within the preparations of the PIPOs.

The observations were during one day of preparation for the two weekly PIPO of the Quadruplex and one day of execution of the PIPO. During the preparation of the PIPO, it took the expert-technician approximately seven hours to be ready for the execution of the PIPO-activities. These preparations included: checking availability of technicians for the PIPO, checking whether ordered parts have arrived at the warehouse, moving the needed parts and material from the warehouse to the factory, searching on different information sources for tasks that should be executed to the machine, searching for personal tasks, prioritizing the maintenance tasks, deciding how many tasks to execute during the next PIPO, assigning the tasks to the available technicians and printing the required documents for the day after. During the day of the execution of the PIPO, five technicians were working on their assigned tasks. The expert-technician instructed the technicians, inspected the machine on abnormalities and checked if the executed work was performed correctly. At the end of the PIPO an unscheduled issue occurred. Since two defects caused the machine not to work, the issue was complex. Because of the knowledge of earlier issues to the machine of the expert-technician the issue was solved relatively fast, in three quarters of an hour.

Information sources

First focus point from the descriptive phase included information sources. All tangible sources that the technicians used for obtaining information are listed in table 4.3. It can be seen that the information is spread amongst a large number of different sources. This makes the work of the technician more complex and time consuming than necessary. Additionally, since most information sources are located at computers in a communal room and not at the machine itself, technicians spend more time than necessary on walking towards and from these computers to the machines. Also mentioned by the technicians themselves is that their work will become more efficient when they could obtain more information at the machines.

Also working at the computers is a time consuming activity for the technicians. This is caused by signing in time, working with different software programs, the time delay that SAP includes when the software is connecting to the database and a possible unavailability of the computers when all five are occupied by other technicians.

Tasks outsourced to specialists

Most of the maintenance work is executed by the technicians of Apollo Vredestein themselves. However, there are cases in which tasks are outsourced to specialists, or in which specialist are asked for assistance. For example, in the case that machine parts are cleaned with acid, this is outsources to a specialist. Also when issues are too complex to be fixed by the technicians of Apollo Vredestein, external specialists are asked for assistance in repairing the machine.

Physical and mental health

The technicians of Apollo Vredestein say that their job is both physically and mentally challenging. However, during the observations and informal interviews the technicians complaint more about the work pressure, causing the job to be mentally challenging, then about their physical health.

Name	Description	Place
Mail + calendar	Work orders, notifications of arrived deliveries and other communication purposes.	PC
SAP	Tasks assigned to technicians, including scheduled repair, inspection or calibration work.	PC
Machine logbook	Including very short reports on the most important events that occurred at the machine, filled in by the operators.	PC
Weekly Excel from managers	Including short reports on all scheduled or unscheduled maintenance tasks that are executed in the week before.	PC
DOS-software with incidents	Generating a list of maintenance tasks that should be performed in the future, ordered per machine.	PC
Document Management System (DMS)	Overview of all parts and components of the machines, including wide information on availability at the warehouse, estimated price when ordering, schematic drawings, documentation and work instructions.	PC
Intranet	Software a.o. for reading data from SAP, which is way more quickly.	PC
Documents on personal drive	Every technician has a personal drive, meant for keeping own important documents. Some technicians use this often, others do not.	PC
Whiteboard at machine	Including very short reports on the most important events that occurred at the machine, filled in by the operators.	Mach.
Printed lists and schemes	Check-lists, list including tasks per technician and including particularities to the PIPO.	Mach.
Displays	The machine has six displays where information for the operator and technician is placed. This varies from raw data (e.g. temperatures) to alerts of failures.	Mach.

Table 4.3 - All information sources used by the technicians

Specialistic assistance
The way that the expert-technician is used within the factory is working effectively, but not efficient. Since this person is asked to handle a lot of issues and questions from other technicians, the workload for the expert-technician increases further. This causes that not all personal tasks can be executed in time for this technician.

Learning and tests
As mentioned in the general observations, every machine in the factory has an emotional owner. This person is a technician that in essence knows most about the machine, including mechanical and electrical parts. Whenever a complex issue occurs at the machine it is likely that this emotional owner, also called expert-technician, is called for assistance. When technicians are performing difficult maintenance tasks without the assistance of the expert-technician, it takes generally more time before the machine is ready for production again. Another observation in this category is that technicians mostly learn by doing. this means that, after they finished a MBO-education for technician, they learn by executing maintenance work and asking experienced colleagues if necessary.

Work pressure
Since the factory is working with a Just In Time (JIT) management, the machines are highly dependent on each other. When a machine has stopped producing unplanned, it causes delays at other machines easily since the factory strives to have as less buffers between machines as possible. In addition, it is only possible for a machine to run when no “red errors” are found, while there is a plausible chance to find one after a PIPO.

Hands On Tool Time
The average Hands On Tool Time (HOTT) of a technician when working on a maintenance task, can be improved significantly. This time, representing the amount of time that a technician is working directly or indirectly productively at the machine, is important to maintenance engineers since losses of HOTT means that the production of the machine has stopped while the technician is working unproductively. Examples of a technician working unproductively are waiting for a colleague or walking through the factory to pick the right up tools or machine documentation.

4.3 User Stories

In order to concretize the observations at Apollo Vredestein, three user stories are written. These stories include scenarios describing the way of working of the technicians. The product that will be examined in this study should eventually improve the way of working in these three scenarios. The user stories are created using the PACT method. Additionally, these user stories gives an answer on the second subquestion of this study.

Scenario 1

Scenario Title
Expert assistance

Scenario Type
User Story

Rationale
This scenario is written in order to give insights in the way of working of the technicians at Apollo Vredestein, and is based on observations at the factory. The topic that is described in this scenario is “expert assistance”. This topic includes the way that a technician is asking for assistance from an experienced colleague.

PACT analysis
People: Robert, a non-expert electrical technician, 23 years old, educated as electrical technician, working for four years in the factory. Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on ‘his’ machine, knowledge of mechanics and electronics.
Activities: Performing a planned maintenance task with the electrical panel of the Quadruplex.
Context: The Quadruplex is a machine in the factory of Apollo Vredestein.
Technology: The technicians use a cell phone to call each other.

Scenario
Robert is an electrical technician, working at the Apollo Vredestein factory for four years. He is 23 years old and finished his education to electrical technician four years ago. In this time he gained experience through asking, listening and trying to succeed the maintenance tasks on his own. It is 8:30 in the morning and Robert is assigned to a PIPO shift for the Quadruplex, a machine that he has never worked with before. At 8:00, when the PIPO started, Robert got instructions from the expert-technician. However, these instructions were brief since the machine was shut down making it important to get started with the maintenance tasks as soon as possible.

Within the given instructions, Robert was told that in the electrical panel a failure was detected in one module. Since the whole machine automatically stops when such a fault is detected, other electrical technicians had quickly installed a bypass over the defect component. After 20 minutes of searching for the bypass without success, Robert decided to ask for help of an experienced technician. His goal is to perform the task as fast as possible without making any mistakes, and since it took already 20 minutes Robert thought the task would be performed faster with expert help. Therefore, he picks his cell phone and he calls Joseph. When Joseph picked up the phone, Robert explains that he cannot find the bypass in the electrical panel, and asks Joseph where he could find it. Joseph mentions that he does not know the specific location either and suggests that he will meet Robert at the machine. Joseph was executing another repairing task, but stops his work for this moment to assist Robert. When Robert has waited for five minutes at the machine, Joseph arrives. Joseph looks to the paper documents including schedules of the machine that he printed for Robert yesterday. He notices that the technicians that quickly installed the bypass did not mention the exact location of this component. After five minutes of searching, Joseph finds the bypass and shows Robert the location. Now, Robert removes the bypass and repairs the defect component. Joseph walks back to the location where he was performing a maintenance task in five minutes.

Scenario 2

Scenario Title
Report instructions for colleagues

Scenario Type
User Story

Rationale
This scenario is written in order to give insights on the way of working of the technicians at Apollo Vredestein, and is based on observations at the factory. The topic that is described in this scenario is reporting instruction for colleagues. When the expert-technician is preparing a PIPO for the next day, instructions are noted for the technicians that will execute the PIPO-tasks.

PACT analysis
People: Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on 'his' machine, knowledge of mechanics and electronics. Dave, a non-expert technician, 31 years old, educated as mechanical technician, working for two years in the factory.
Activities: Inspecting the Quadruplex, looking for tasks to pick up in the next PIPO.
Context: The Quadruplex is a machine in the factory of Apollo Vredestein.
Technology: Software at the computers within the communal room of the technicians.

Scenario
Joseph, an expert-technician and emotional owner of the Quadruplex machine, is preparing the PIPO for tomorrow. It is 11:00 in the morning, and he already spend three hours on a short inspection of the machine, the selection of tasks to include in the PIPO and picking up the right materials and parts. Joseph's goal for this preparation is that the PIPO will be executed as efficient as possible without any mistakes. Therefore, it is important that all technicians know exactly what they should do, and have all necessary tools and machine parts immediately available during the time that the machine is shut down.

One of the tasks that will be executed tomorrow is the substitution of a defect roll, which is a part of the roller conveyor that is transporting rubber from the extruder to the cooling zone. The rotating shaft of the roll got an accidental hit, resulting the shaft being unbalanced. This causes friction that could affect the quality of the product. Since rubber is transported over the roller conveyor at production time, this task can only be performed when the machine is shut down.

Joseph assigned this task to mechanical technician Dave, and he wants that Dave has all resources to quickly replace the defect roll for a new roll tomorrow. Joseph walks from the machine to the computer to print additional information for Dave. When Joseph arrives, he writes in the task description "Replace defect roll (the fifth from the machine opening)". Additionally, he prints the work document for Dave on how to substitute rolls. Since Dave has never replaced rolls before Joseph thinks it could help him perform the task without making a mistake. He will give this document to Dave personally tomorrow. Since a new machine part is required for the task, Joseph walks to the warehouse to pick up the roll. In order to be sure that Dave can quickly access the roll, Joseph puts the box including the roll at the machine, next to other boxes that also include machine parts that will be attached tomorrow.

Scenario 3

Scenario Title
Report task rejection

Scenario Type
User Story

Rationale
This scenario is written in order to give insights on the way of working of the technicians at Apollo Vredestein, and is based on observations at the factory. The topic that is described in this scenario is reporting instruction for colleagues. When the expert-technician is preparing a PIPO for the next day, instructions are noted for the technicians that will execute the PIPO-tasks.

PACT analysis
People: Dave, a non-expert technician, 31 years old, educated as mechanical technician, working for two years in the factory. Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on 'his' machine, knowledge of mechanics and electronics.
Activities: Replacement of a defect roll of a roller conveyor of the Quadruplex, as a planned maintenance task.
Context: The Quadruplex is a machine in the factory of Apollo Vredestein.
Technology: The DMS software program at the computers in the communal room of the technicians.

Scenario
It is 12:30 in the afternoon and Dave, a mechanical technician at Vredestein, is working on his PIPO tasks on the Quadruplex. The PIPO of today is scheduled till 13:00, meaning that the machine should start producing at 13:30. Dave finished all his tasks except for the last task on his list, the replacement of a defect roll of a roller conveyor of the Quadruplex[1]. This morning, Dave received an instruction and a work document from Joseph to perform the maintenance task. He walks to the place where all new parts are deposited in boxes and looks in several boxes to find the new roll. After three checks, he found the right box and walks with the new roll to the roller conveyor. Since Joseph wrote the note that he should replace the fifth roll from the machine opening, Dave could count which roll to replace. However, technicians use different names for machine parts. This causes that Dave is in doubt from which side to count, and he decides to examine this first, to prevent replacing the wrong roll. After he identified the unbalanced roll, Dave started to detach the defect roll. He finds that the roll is so stuck that it cannot be replaced following the normal way. It is already 12:45 and Dave calls Joseph for assistance. Joseph says that he will come take a look himself. Joseph, which stopped his own repair task for the moment, arrives after five minutes at the roller conveyor. He tries to detach the roll himself, but concludes that it would take at least 30 minutes to detach the roll. Since it is more important to get the machine running at 13:00, Joseph decides to let the roll as it is for the moment and walks back to place where he was performing his task earlier. After the PIPO, Joseph walks to the computers in the communal room, and opens the Excel-sheet in which he should fill in what tasks are performed and what tasks could not be performed. Joseph types a description of the defect roll and notes to prepare special material for the next PIPO to detach the roll then.

End Notes
1. See scenario 2 for detailed explanation of this task.

4.4 Idea Generation

Now the way of working of the technicians at the factory of Apollo Vredestein is described with scenarios, concept ideas can be developed. These ideas have a potential for improving the technicians efficiency during their work. First, four ideas are described below. These are based on both the user needs and the information found during the state of the art of this study. Second, these ideas are shortly evaluated. As a result of this evaluation, one idea is chosen to examine in this research.

4.4.1 Augmented Reality Concept Ideas

Training

According to [18] "augmented reality […] has large potential to provide learners with a new type of learning material." Promising about this technology in the domain of maintenance in industrial context is the necessity for proper education of technicians and the large amount of available documentation in this domain [19]. A product using augmented reality that is training technicians can be designed in three ways:

- A system that recognizes the environment, the equipment and the target machine. While performing a task, the augmented reality device explains what to do next to the technician with text, visual content or speech, and displays virtual objects on the technician's view in order to clarify the explanation. The virtual objects can be a full "hologram" of an arm or equipment virtually completing the task, or can be clarifying figures like arrows and circles.
- A second way to implement an augmented reality training is described in [18] using a fusion between the traditional explanation on paper, enhanced with additional information. The additional information is then displayed with augmented reality on the paper, and can include small virtual objects in 3D, videos or live information.
- Third, an illusion can be created in which a learning object, like an engine, is floating in front of the technician that can interact with this object by turning and moving it in all directions. In addition, it can be possible that the technician removes parts of the object in order to look at the inside of it.

Part finder

A product helping the technician to find the part of the machine where the technician is looking for can has a positive effect on the efficiency of the technician's work. Especially for young workers or for workers that are repairing a machine area that is unknown for them such a product can save time.

Task instruction

The experiment described in [3] researched the comparative effectiveness of an augmented reality instruction. This instruction is providing the user with an additional 3D overlay guiding the user what to do step-by-step. The researchers found that this reduced the error rate while performing the task with 82%, concluding that a similar augmented reality instruction could increase users effectivity with large numbers. A task instruction can also help the user in the domain of maintenance, especially when completing complex or infrequent tasks. In addition to a decrease of the error rate, an augmented reality instruction can increase the efficiency of the technicians' work. Other products in the related work that involve task instruction can be divided into three categories:

- The most simple way of using augmented reality in a task instruction product is with textual information. Text can be aligned to the corresponding real world objects, or can be placed statically in the users view. Since such textual information mostly already exists in the domain of machine maintenance in the form of documentation, implementation of this technology will not become expensive or time consuming. Additional to textual information, static images or videos can be displayed to the user.
- A more developed stage of augmented reality is the use of virtual objects like arrows, circles and lines to guide the users attention to the right point and to provide textual information with

clarifying visuals. Since this technology should involve tracking of the users environment, and the software should recognize targets around the user, creation of such a product will become more expensive and time costing.

- Existing augmented reality products mostly involves customized virtual 3D objects that are displayed on top of the real world object. An example of this in the domain of maintenance can be found in the result of Inglobe Technologies' case study [9], which involves virtual wires in different colors and a virtual tool to attach these wires in order to show the technician how these wires should be attached to the device and what tool should be used. Since this technology requires to design customized 3D objects and tracking of the users environment, such a product will be the most expensive and time consuming option.

Specialistic help
As seen in the examples of Fieldbit's service [12] and CN2Tech's product 'Assist' [13], another way to include augmented reality in a product is with involving a human being as specialist into the product instead of the software doing all the work. Within such a product, the user having problems with performing a task is able to share its view with a specialist that can be located anywhere else in the world. The specialist uses its deep knowledge about a device, machine or other object to help the user performing the task, preventing making mistakes. In order to make such a product more efficient and effective, the specialist is able to draw virtual objects and text on the view of the user. Since these objects will automatically align with the real world objects in the users environment, it guides the user through the steps more easily than with only information over speech. In addition, since it can reduce the possibility of miscommunications which will increase effectivity. No costs are involved for creating customized 3D models. Although this will safe costs at the development of the software, it should be noticed that constant costs are involved for paying the specialists during the usage of such a product.

4.4.2 Evaluating the Concept Ideas

In order to gain a quick insight in the opportunities for the four technologies, the advantages and disadvantages of each augmented reality solution are given in table 4.4. This evaluation is based on the hypothetical situations that the ideas would be used in the factory of Apollo Vredestein. An overall score is given by the researcher to each technology by outweighing the disadvantages to the advantages. Explanation of these scores are given below the table.

Solution	Advantages	Disadvantaged	Score
Training	+ Learning more efficient & effective	- Constantly new costs for developing additional training	+/-
Part Finder	+ Increase efficiency of technician's work	- Constantly new updates required for new machines - Mostly necessary for starting technicians	-
Task Instruction	+ Increase efficiency and effectivity of technician's work + Costs and time will be saved for the use of specialists	- High costs for developing instructions for all machines	+
Specialistic Help	+ Increase efficiency and effectivity of technician's work + It will save the specialist time + After launch of product, only small updates are required	- Costs for developing software and specialists	++

Table 4.4 - Advantages and disadvantages to four augmented reality solutions

When looking to the results displayed in table 4.4, it can be found that a 'specialistic help' or 'task instruction' potentially are strong tools for Apollo Vredestein. While the maintenance expert tells that communication between the technicians is a topic that asks for improvement, 'specialistic help' is expected to be a tool that strengthen this communication. The largest advantage of the 'specialistic help' compared to other solutions is that, next to costs for developing software, there are no additional costs for creating content involved. The technicians are creating content themselves, instead of developers creating 3D models and task instructions on forehand. Since the experts are mostly already available within the factory, no additional costs are involved for them too. The advantage of a 'task instruction' is the increased efficiency of the technicians, which should be a goal of the concept idea. Disadvantage of the 'task instruction' is the costs for creating content, like 3D models and task descriptions. Therefore, the concept idea should include the already existing work documents, so that no 3D models are required. The concept idea should be an extension of the Tip4Support application. Within this application it is possible for technicians to check off tasks that they performed and it is possible to reject tasks that could not be executed for some reason. A concept in which the technicians could visually show other technicians what the reason of rejection is, would be a fitting extension. This is verified with Innovadis. All aspects taken into account, it can be concluded that an augmented reality tool for specialistic help has most potential to successfully be implemented and used in Apollo Vredestein's factory. Therefore this idea is chosen to be developed further and examined in this research. The next section will describe the product idea in more detail.

4.5 Product Idea

The idea for a product that will be examined during this study can be described as "a communication tool for technicians with which technicians can inform colleagues with drawing visual objects on machines that are displayed with the use of augmented reality technology." This means that the technicians can inform each other while these are using the software simultaneously and when the technicians use the software at different times. The information that they share should mostly include visual information, since this prevents misunderstandings of written text. However, the product should include ways to explain visual information with text. A visualization of how the visual information could look like is given in figure 4.2.

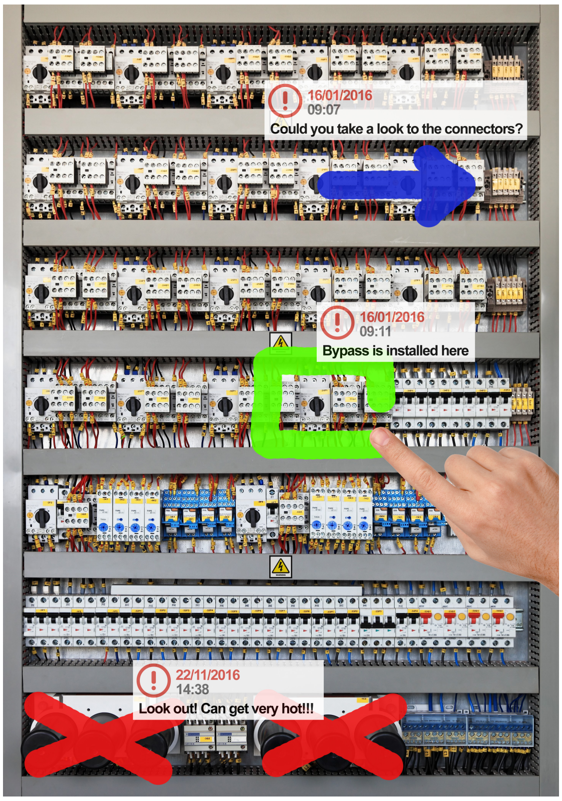


Fig. 4.2 - A visualization of the product idea: visual communication for technicians

5. Specification

The meaning of the specification phase is exploring the product design space. Prototyping and discussion with the end-users serve as input for the designer perspective scenarios. These scenarios form specified stories on how the product idea from the ideation phase can operate in the technicians' work. At the end of the scenarios, the need of augmented reality technology in the scenario's situations is described. Goal of this part is to ensure that traditional techniques are not already a solution for the challenges in the technician's work. Lastly, these scenarios are discussed with the envisioned end-users of the product. Their input is used to create a list of functional and non-functional requirements, which serve as input for the realization of the final prototype.

5.1 Prototype

To explore what possibilities are existing with augmented reality technology, this phase started with prototyping. Early in this process it became clear that using the Vuforia (https://vuforia.com) software offered a reliable way of creating an augmented reality experience. Using this software in Unity3D (https://unity3d.com), a 3D environment program, enables developers to customize the augmented reality experiences in all kinds of aspects. Since it is possible to use C# scripts in Unity3D, developers can program their own game, tool or application around the Vuforia augmented reality package. How this is implemented within the final prototype is explained in the realization section. The early prototype that resulted as input for the designer perspective scenarios can be seen in figure 5.1. This prototype was able to detect an image of a machine (see figure 5.2) and could overlay three different planes over the machine, which were static images. These images included machine information, a warning about the machine and a graphical user interface of three buttons.

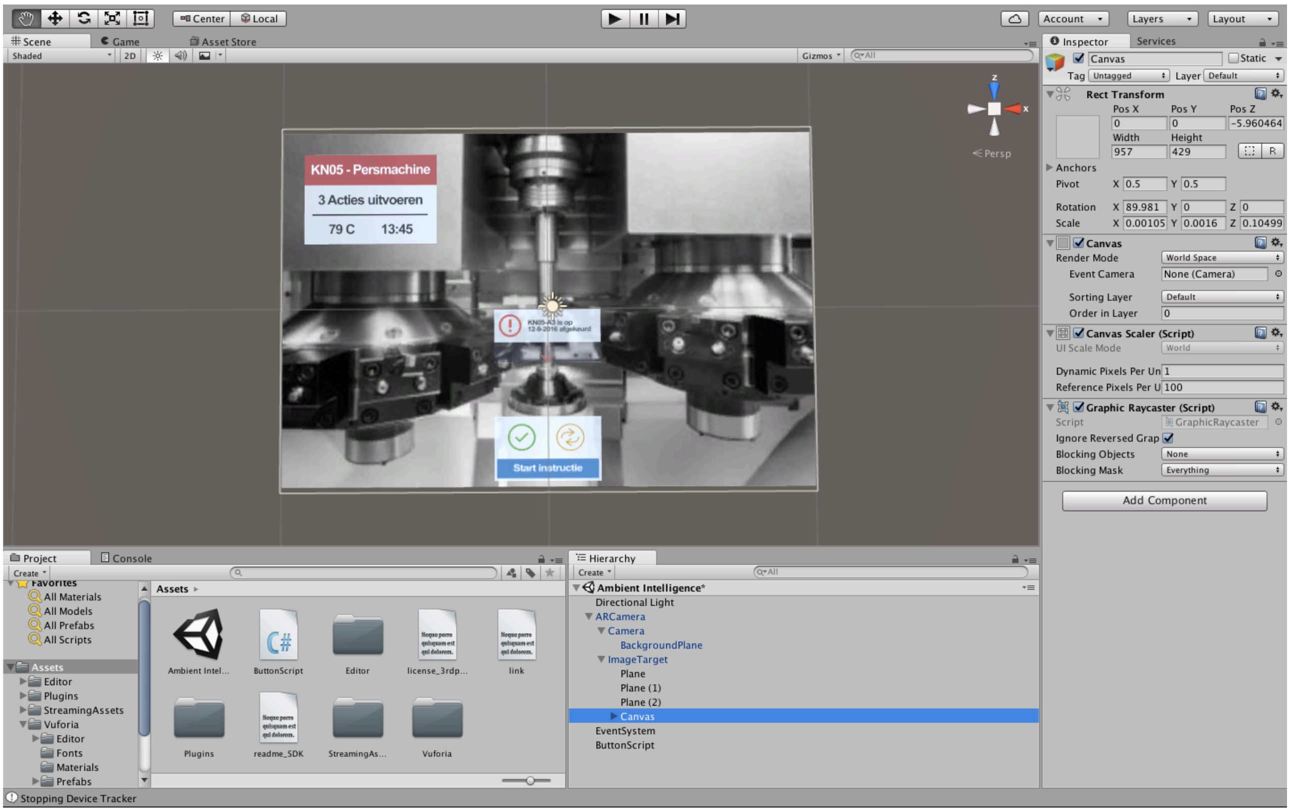


Fig. 5.1 - Augmented reality prototype, using Vuforia in Unity3D

At the end of the prototyping phase, solutions were found on how to get the application build in Unity3D on a laptop could be installed on an iPad or iPhone. This is done with the "Simple Mobile Placeholder" package of Unity3D [22], in combination with Xcode (https://developer.apple.com/xcode/). More details on this topic are also given in the realization section.

Fig 5.2 - Image of machine that could be detected by the prototype

5.2 Designer Perspective Scenarios

The augmented reality solution, described within the three written scenarios below, focusses on the challenges that are found within the observations of the technicians at the Apollo Vredestein factory. Theoretically, if this solution could cause a faster and more effective communication between the technicians, it could improve two situations within the factory. The corresponding goals of these scenarios are given beneath. The three scenarios are answering the third subquestion of this study.

1. An improvement of the HOTT for technicians working on their PIPO tasks while the machine is shut down. This could increase the efficiency of the PIPO tasks that are executed, and will decrease the average amount of time that a PIPO takes.
2. Since technicians know better what their PIPO tasks include, the frequency of asking for help to the expert technician will decrease. In addition, when the expert technician is asked to help a colleague the goal is to decrease the amount of time that is needed for helping, causing that the expert technician has more time to focus on its own work.

Scenario 1

Scenario Title
Fast expert assistance over distance

Scenario Type
User Story

Rationale
This scenario is written in order to give insights on how an augmented reality solution could improve the efficiency of technicians' work. Goals in this scenario are a faster communication between technicians and the non-expert technicians knowing better how to perform their tasks. In addition, the assisting time of the expert-technician should be used efficiently without loosing too much time for its own tasks.

PACT analysis
People: Robert, a non-expert electrical technician, 23 years old, educated as electrical technician, working for four years in the factory. Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on "his" machine, knowledge of mechanics and electronics.
Activities: Performing a planned maintenance task with the electrical panel of the Quadruplex.
Context: The Quadruplex is a machine in the factory of Apollo Vredestein.

Technology: A device, could be a head-mounted display or a tablet, that enables the user to look to its own view enhanced with augmented reality elements. In addition, this device is able to share its camera view via a wireless connection with another device.

Scenario
Robert is an electrical technician, working at the Apollo Vredestein factory for four years. He is 23 years old and finished his education to electrical technician four years ago. In this time he gained experience through asking, listening and trying to succeed the maintenance tasks on his own. It is 8:30 in the morning and Robert is assigned to a PIPO shift for the Quadruplex, a machine that he has never worked with before. At 8:00, when the PIPO started, Robert got instructions from the expert-technician. However, these instructions were brief since the machine was shut down making it important to get started with the maintenance tasks as soon as possible.

Within the given instructions, Robert was told that in the electrical panel a failure was detected in one module. Since the whole machine automatically stops when such a fault is detected, other electrical technicians had quickly installed a bypass over the defect component. After 20 minutes of searching to the bypass without success, Robert decided to ask for help to an experienced technician. His goal is to perform the task as fast as possible without making any mistakes, and since it took already 20 minutes Robert thought the task would be performed faster with expert help. Therefore, he picks his assistance-device, chooses “Video call” from the menu and starts a call with Joseph, the expert technician of the Quadruplex. Joseph picks up the call[1] and he is immediately able to look to the camera view of Robert, as if Joseph was looking over the shoulder of Robert in the direction of the electrical panel. With his voice Robert explains briefly his problem to which Joseph responds with a small explanation where to find the bypass. In order to make sure both are talking about the same area, Joseph draws a circle on Robert’s view[3]. This circle aligns itself with the real world environment of Robert and sticks to the same position in the environment for every movement Robert makes. In addition, Joseph thinks that a technical sketch of the electrical panel could help Robert to better understand what he should do. Therefore, Joseph adds this sketch to Robert’s view[2]. Within three minutes Robert obtained enough information to perform the task. He thanks Joseph for his assistance and ends the call. Also now, Robert is still able to see the drawn circle and the sketch in his view. When he has completed the task, he deletes both items from the view and continues to the next PIPO task.

End Notes
1. The device that Joseph is using does not necessarily has to be the same kind of device as Robert is using.
2. This does not necessarily mean that Robert sees the sketch floating in his surrounding environment. For example, when using a tablet it would be easier if the sketch could be opened full-screen, instead of presence in the camera view.
3. In order to align the AR-objects to the real world, the software should have a reference in the real world that can be recognized. To make the product as most reliable as possible, the reference objects ideally are parts of the machine.

Scenario 2

Scenario Title
Report instructions for colleagues

Scenario Type
User Story

Rationale
This scenario is written in order to give insights on how an augmented reality solution could improve the efficiency of technicians’ work. Goals in this scenario are a faster communication between technicians and the non-expert technicians knowing better how to perform their tasks. In

addition, the assisting time of the expert-technician should be used efficiently without loosing too much time for its own tasks.

PACT analysis

People: Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on "his" machine, knowledge of mechanics and electronics. Dave, a non-expert technician, 31 years old, educated as mechanical technician, working for two years in the factory.

Activities: Inspecting the Quadruplex, looking for tasks to pick up in the next PIPO.

Context: The Quadruplex is a machine in the factory of Apollo Vredestein.

Technology: A device, could be a head-mounted display or a tablet, that enables the user to look to its own view enhanced with augmented reality elements. In addition, this device is able to share data with the devices of other technicians.

Scenario

Joseph, an expert-technician and emotional owner of the Quadruplex machine, is preparing the PIPO for tomorrow. It is 11:00 in the morning, and he already spend three hours on a short inspection of the machine, the selection of tasks to include in the PIPO and picking up the right materials and parts. Joseph's goal for this preparation is that the PIPO will be executed as efficient as possible without any mistakes. Therefore, it is important that all technicians know exactly what they should do, and should all necessary tools and machine parts be immediately available during the time that the machine is shut down.

One of the tasks that will be executed tomorrow is the substitution of a defect roll, which is a part of the roller conveyor that is transporting rubber from the extruder to the cooling zone. The rotating shaft of the roll got a hit, resulting the shaft being unbalanced. This causes friction that could affect the product. Since rubber is transported over the roller conveyor at production time, this task can only be performed when the machine is shut down.

Joseph assigned this task to mechanical technician Dave, and he wants that Dave has all resources to quickly replace the defect roll for a new roll tomorrow. Therefore, Joseph walks to the place where the task will be executed and starts the augmented reality application on his device. With this device he scans the machine shortly, so that the application can identify the place of the machine[1], and loads the "Substitute defect roll" task to his view. Now, Joseph can add visual information to the view in order to explain Dave in more detail what he should do tomorrow. First, he highlights the roll that should be replaced. Secondly, Joseph adds a work document to the view on how to substitute rolls. Since Dave has never replaced rolls before Joseph thinks it could help him perform the task without making a mistake. Lastly, Joseph walks to the place where all machine parts that will be used in the PIPO are deposited, and highlights the box including the new roll. Tomorrow, the system will show Dave the same augmented reality objects that Joseph drew today. Besides that, the application knows that Joseph assigned the task to Dave, meaning that when Dave still requires help the system will suggest him to contact Joseph. Tomorrow, Dave does not have to spend time on searching for the right roll or the right box, causing that the task will be executed more efficiently[2].

End Notes

1. This scan could vary between the identification of one static 2D image, like a QR-code, to the immediate identification of parts of the machine. Ease of this scan depends on the tracking method.
2. Since the device that Dave is using during the PIPO does not necessarily have to be the same device as Joseph has used for preparation, it is important that all augmented reality devices share their data.

Scenario 3

Scenario Title
Report task rejection with visual information

Scenario Type
User Story

Rationale
This scenario is written in order to give insights on how an augmented reality solution could improve the efficiency of technicians' work. Goals in this scenario are a faster communication between technicians and the non-expert technicians knowing better how to perform their tasks. In addition, the assisting time of the expert-technician should be used efficiently without loosing too much time for its own tasks.

PACT analysis
People: A non-expert technician, 31 years old, educated as mechanical technician, working for two years in the factory. Joseph, an expert-technician, 40 years old, emotional owner of the Quadruplex, normally instructing other technicians working on "his" machine, knowledge of mechanics and electronics.
Activities: Replacement of a defect roll of a roller conveyor of the Quadruplex, as a planned maintenance task.
Context: The Quadruplex is a machine in the factory of Apollo Vredestein.
Technology: A device, could be a head-mounted display or a tablet, that enables the user to look to its own view enhanced with augmented reality elements. In addition, this device is able to share data with the devices of other technicians.

Scenario
It is 12:30 in the afternoon and Dave, a mechanical technician at Vredestein, is working on his PIPO tasks on the Quadruplex. The PIPO of today is scheduled till 13:00, meaning that the machine should start producing at 13:30. Dave finished all his tasks except for the last task on his list including the replacement of a defect roll of a roller conveyor of the Quadruplex[1]. He opens this task on his augmented reality device, and sees that Joseph added visual information during the preparations of the PIPO. When Dave is at the place where all new parts are deposited in boxes, he sees that the box he needs is automatically detected and highlighted. Dave picks the new roll out of the box and walks to the roller conveyor, where he sees that the defect roll is already highlighted. Also, he notices the work document that is attached to this task, and he quickly reads the instructions. When Dave has started detaching the defect roll he finds that the roll is so stuck that it cannot be replaced following the normal way. It is already 12:40, and after a video call with Joseph, they concluded that detaching the roll will already take 30 minutes. Since Joseph said it was more important to get the machine running at 13:00 than to have the roll replaced, he instructed Dave to reject the task in the system.

When Dave has rejected the task, the system suggest to add visual information to the rejection so that the supervisor, Joseph and the technician that will follow up this task in the next PIPO will know where to find the defect. Dave scans the machine with the augmented reality device, points the camera towards the position of the stuck roll and draws circles around the bolts that are stuck. In addition, Dave types a description of the defect and suggest to prepare special material for the next PIPO. In this way, the technician that will be assigned to this task in the next PIPO will immediately obtain the right information and materials.

End Notes
1. See scenario 2 for detailed explanation of this task.

5.2.2 The Need for Augmented Reality

It is expected that the use of augmented reality within the three scenarios are increasing the technicians efficiency better than other solutions not using the augmented reality technology for two reasons. First, actions like drawing a circle or highlighting a component of the machine are not time consuming while they can be a very clear form of communication between technicians. Especially in more difficult situations, or when it is hard to explain with voice which component of the machine is meant, additional visual communication can more effectively explain a solution or issue. Because it is possible to immediately draw on a technicians view, no extra time is necessary for transferring this visual communication between the technicians. Second, augmented reality objects that are aligned with the real world give more context than other visual solutions like pictures. Since photos are taken from one perspective, a technician receiving pictures mostly needs an amount of time to figure out where the pictures are taken exactly. In addition, photos can also cause misunderstandings when they could be taken at more than one place in the factory. Augmented reality will always align according to the position of the user, causing that the technician does not have to spend time on figuring out where the additional information is placed. Also, it is not possible that misunderstandings will happen about the place of the information, since the information will only be visible at the place where the first technician drew it.

The advantage of using an augmented reality solution over traditional ways of marking visual information, is that it is not possible to mark moving parts directly while the machine is producing. In this situation it is only an option to highlight a machine part indirectly, possibly resulting in misunderstandings. In addition, marks like a post-it, a note or a drawn cross that are placed on not moving parts of the machine can accidentally disappear. In the observations at Apollo Vredestein it is concluded that due to the chaotic environment in the factory and movement of the machine such marks can be removed. Lastly, when an expert technician explains an issue to a colleague over distance, it is not possible to physically mark in the colleagues environment at that moment.

5.3 Interview with Users

In this phase it is important to introduce the augmented reality concept, as described within the three scenarios, to the target group. In order to do so, and to get a better insight in the users requirements to the concept, a semi-structured interview is performed with an expert technician and with a supervisor of a group of technicians at the factory of Apollo Vredestein. This interview includes two parts: a check of the conclusions drawn from the observations in the factory, and the elicitation of functional and non-functional requirements of the proposed augmented reality solution. To give the participants of this interview a better understanding of the three scenarios, a live demo was showed at the start of the second part. The full list of questions and answers of the user group can be found in appendix A.

5.3.1 Discussion Scenarios

While discussing the scenarios and the early prototype (see section 5.1), it became clear that both the supervisor and technician saw large advantages to visual communication between technicians, as support to the nowadays used verbal communication. In contrast to the scenarios that are mostly based on proactive work of the technicians, both participants mainly named the advantages for reactive and incidental situations. Especially the situation in which an operator can visually communicate with an technician at another location, was appointed as presumably very valuable. Since the technician can instruct the operator on executing a simple task, like parts ordering or detaching parts, these could already be executed when the technician arrives at the location of the defect. Eventually, it can result in the machine being fixed earlier.

Suggested functionality by the supervisor and technician were about making important machine(-component) information available within the software. They saw advantages to the situation in which a technician has this kind of information on a mobile device at the machine, instead of the PC's. This can save time, and likely prevents mistakes about component names, execution of tasks, etc. Information can be technical schemes, component names, manufacturer, warehouse availability and working documents. Lastly, the technician suggested functionality for newer machines, that has I/O tests on every component. When the software would be connected to such a machine, it could display in augmented reality the results of these I/O tests over the components of the machine.

Disadvantages were seen in the costs of the introduction of mobile devices at the factory. Apollo Vredestein had plans already for having a pilot with tablets, which would cost €1500,- per technician. In addition, costs for infrastructure of the devices should be encountered. This includes a help-desk, distribution of the devices and people that take care of installing the right software and executing repair. The technician also named that software should be clear and easy to understand.

5.3 Product Requirements

From the outcomes of the interview a list of functional- and non-functional requirements is constructed. The list of functional requirements is more comprehensive than the list of non-functional requirements, since the focus of the interview was on possible functionalities of the augmented reality system. In a later interview with only the expert, the requirements are evaluated and prioritized. The prioritization is executed following the MoSCoW methodology, which is frequently used for IT projects. In this method, all requirements are divided in one of the four categories ‘Must haves’, ‘Should haves’, ‘Could haves’, ‘Won’t haves’ [24].

In order to build a proof of concept that can evaluate the essential of the product idea, all functional ‘Must haves’ requirements should be implemented. In addition, when the prototype should also evaluate other wishes of the technicians, the ‘Should haves’ must also be implemented in the realization. The functionalities listed in the ‘Could haves’ category are decided not to be included in the implementation of the prototype. The reason for this is that this research should stay focussed on the main functionalities of the product idea, making room for a thorough evaluation of these.

No.	Requirement	Prio.
R1	The user is able to see its environment through the device, possibly by camera view	Must
R2.1	The software recognizes the user’s environment and real world perspectives	Must
R2.2	The user is instructed to scan an object that the 3D engine can recognize	Must
R2.3	Feedback is provided when the user’s environment is recognized	Must
R3.1	The user is be able to draw figures, like lines and circles, on the view	Must
R3.2	The user is able to add objects with three different colors: red, green/yellow and blue	Must
R3.3	The drawer figures stay aligned with the user’s environment	Must
R4.1	The user is be able to add text-messages to the view	Must
R4.2	When a text-message is displayed, date and time of creation is given together with the name of the author	Must
R4.3	The added text-messages stay aligned with the user’s environment	Must
R5	The user is be able to remove figures and text-messages from the view	Must
R6.1	The software is be able to identify specific parts of the machine	Must
R6.2	The software only shows figures and text-messages that are relevant for the identified part of a machine	Must
R7.1	The user is able to share the camera view with the user of another device	Must

R7.2	The user of the other device is be able to draw figures, add text-messages and remove these to/from the views of both devices	Must
R7.3	The user is able to stop sharing the camera view with the user of the connected device	Must
R8	The devices automatically synchronize the figures and text-messages, so that all information on parts of machines is the same for both users	Must
R9	Both users should be able to talk to each other, like being on the phone	Must
R10.1	When a machine-part is recognized, the software can show a list including all components	Should
R10.2	In the list, for all components relevant information can be shown, including name/type, manufacturer, warehouse availability, technical scheme and workings of the component	Should
R11	The user is able to obtain relevant working documents to a component or machine part	Should
R12.1	The software is able to identify different components within a part of the machine	Could
R12.2	The software can show the components names	Could
R12.3	Other relevant component information can be shown, including manufacturer, warehouse availability, technician scheme and workings of the component	Could
R13	The user is able to order a component from the application	Could
R14.1	The software shows live information from the machine	Could
R14.2	The live information of the machine is aligned the the right components in the machine	Could

Table 5.1 - Functional requirements

R1	The user interface should be clear and very easy to understand	Must
R2	Tracking target is recognized from a distance that the technician is normally standing away from a machine when working an a repairing task	Must
R3	Connection when video sharing has enough quality for fluent visual communication	Must
R4	The audio communication is loud enough to be hearable over the machine noises	Must
R5	Figures and text-messages stay visible when tracking target is out of view	Must
R6	The machine/component information will be retrieved from the DMS system	Should

Table 5.2 - Non-functional requirements

6. Realization

In this section it is described how the final prototype is constructed. First, the system architecture is explained, partly consisting of an analysis on user interaction of the system, and partly consisting of a system decomposition. Second, it is described how these compositions and user interactions are designed and implemented in the final prototype. This is explained along the topics: programming software, user interface, augmented reality tracking and networking.

6.1 System Architecture

6.1.1 User interaction

In order to get a clear overview of which interactions the system should enable, all requirements that involve user interaction are included in a use case diagram (see figure 6.1). This diagram includes two different users: technicians, who are users that could be executing or preparing PIPO-tasks, and expert technicians, who are called by the technicians for help. Both users are interacting with an iPad, which are connected with each other. Each side of the system includes a list of interactions that are enabled for the user on the same side. Interactions that are displayed in the middle are 'include relationships', or 'extend relationships' [27]. A blue line between two interactions means that the middle interaction will definitely follow from the interaction on the side, in UML called an 'include relationship'. A yellow line between two interactions means that the middle interaction could follow from the interaction on the side, but not necessarily. In UML, these are called 'extend relationship'. In the diagram it can be seen that the technician includes the same interactions as the expert technicians, extended by other interactions.

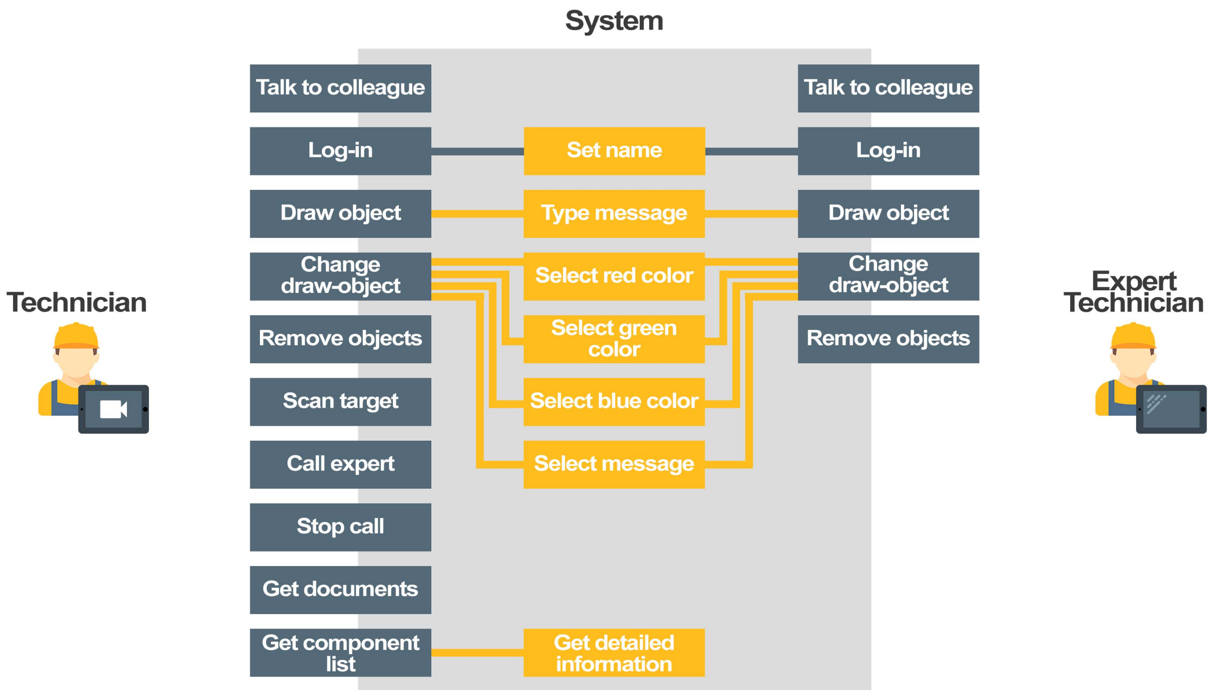


Fig. 6.1 - Use Case Diagram

6.1.2 System Decomposition

One way to design a systems architecture, is to describe the system, or system parts, as black boxes. Each black box has inputs an an outputs. These boxes are designed on a high level first, including a box that represents the entire system. Afterwards, the black boxes will represent smaller parts of the system, and will become more detailed.

In figure 6.2, the entire system is represented by one black box. In this case, two users are interacting with the system. All user input, which is listed in the use case diagram in the previous paragraph, is generalized as “user interaction”. These interactions include operating the application with the touch-screen of an iPad as well as talking to the system. The technician executing or preparing the PIPO receives output from the application in three categories. First, the voice of the helping colleague. Second, the camera view including augmented reality objects. Last, documentation of the machine. This includes both PDF-documents and a list with details from machine components.

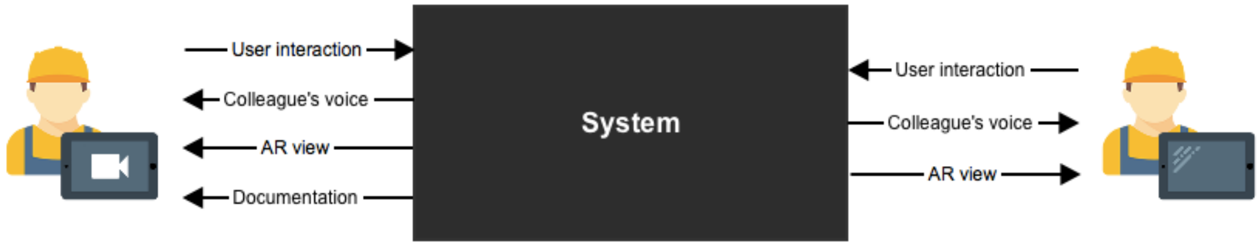


Fig. 6.2 - Black box 1: input and output of the entire system

System

When opening the back box of the entire system, it can be seen that the system consists of two iPads that communicate over a network (see figure 6.3). Which protocols are used to establish the network communication, and what messages are communicated, will be discussed in more detail in the next chapter about implementation.

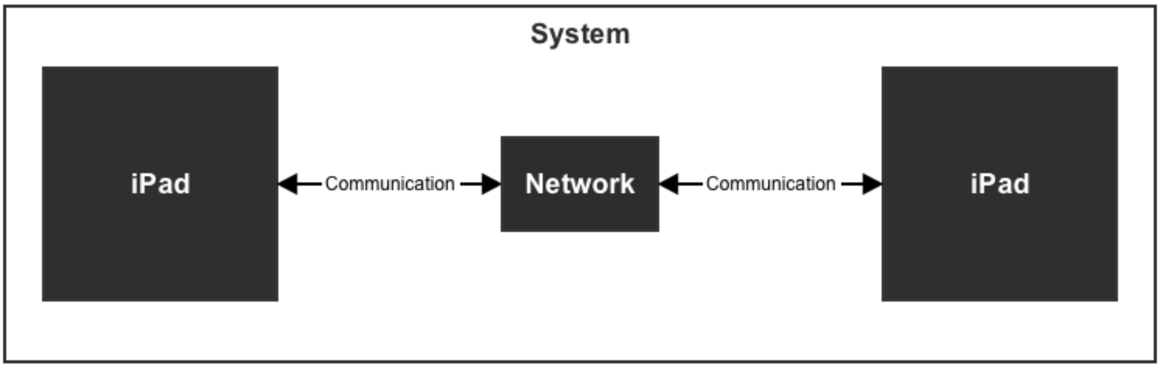


Fig. 6.3 - Black box 2: the entire system consists of two iPads, communicating over a network

iPad

The iPads consist of four main components that are collaborating to process the users' input into the desired output for these users. The input and output to the system seen in figure 6.2 are processed first by input and output components on the devices. Voice is registered with a microphone, while other user input is registered by the iPad's touchscreen. Camera input is registered by the camera of the iPad. Output to the user is given via audio and the iPad's display. User input and application output is processed by the operating system, and functions as a mediator between applications and input/output components. Although more communication is present between the operating system and other parts, only communication that is specifically build for or used by the prototype is given (see figure 6.4). Since the operating system includes the networking component, all applications are communicating over the network via the operating system.

Besides the augmented reality application, two other applications are used on the iPad. Safari, the standard internet browser of Apple, is used to open PDF-documentation. FaceTime, the standard video call software of Apple, is used to establish a voice communication between the two iPads. The required input for the augmented reality applications, besides network communication, is the camera view and touchscreen information. Only the black box of the augmented reality application will be described in more detail, since other black boxes within the iPad were parts that are not specifically designed for the prototype. All yellow components and input/output connections are used by the application for the technician that is preparing or executing PIPO tasks. The application for the assisting technician is not using these components and connections.

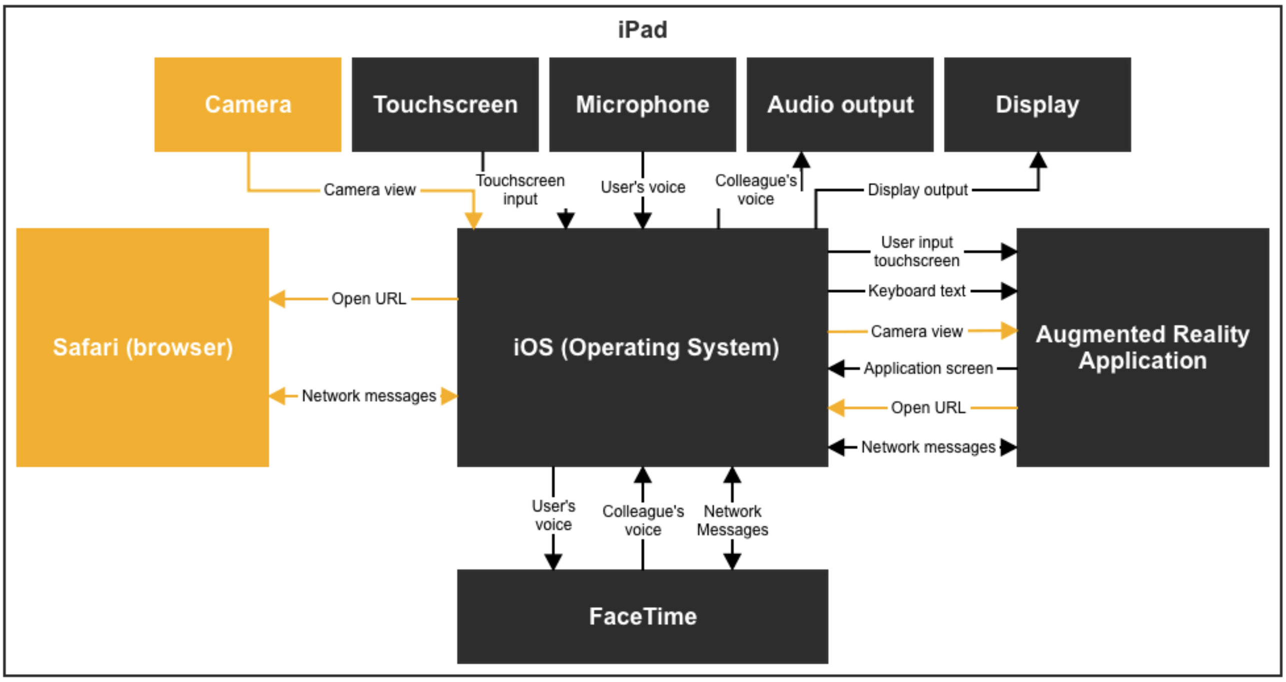


Fig. 6.4 - Black box 3: all components on the iPad used for the prototype

Augmented Reality Application

The augmented reality application consists of an application core that communicates with three main software parts (see figure 6.5). Since only the core can communicate with the operating system, all communication from and towards outside the application needs to be communicated by the core. It is decided to divide the software in the three parts networking, user interface and augmented reality tracking. The networking part is responsible for the communication with the other iPad. This includes networking protocols, handling network changes and creating or reading network messages from and towards the other iPad.

The user interface includes components, like buttons and input-fields, that are places statically on the view of the application. These components do not align with the camera view or the 3D environment. The user interface is responsible for displaying or obtaining information towards or from the user. The augmented reality tracking part is aligning a 3D-environment to the camera view that it receives from the core. When a machine area is recognized in the camera footage, and the 3D-environment is build accordingly, virtual objects can be drawn inside this environment. The augmented reality tracking component is responsible for the recognizing machine areas and aligning virtual objects to this environment, so that these create an augmented reality experience. All yellow connections are only used by the application of the technician that is preparing or executing PIPO tasks. The blue connection is only used by the application of the assisting technician.

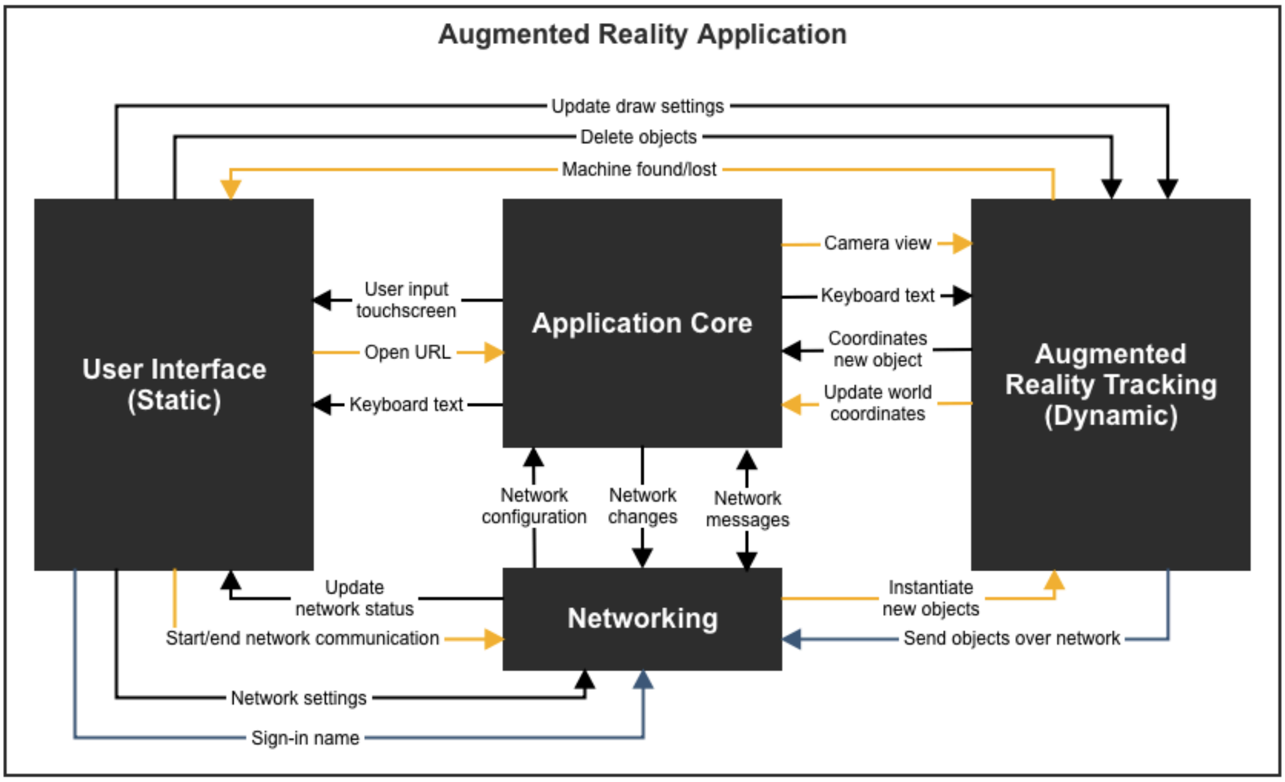


Fig. 6.5 - Black box 4: the four main components of the augmented reality application

User Interface

The user interface of the application consists of four categories: drawing, machine information, network settings and network information (see figure 6.6). The category drawing includes components to influence the type of drawing within the application. The term ‘object selection’ refers to the functionality to select a color or to select drawing messages instead of drawing circles. The category machine information requires information on which machine is found by the tracking component of the application. This category uses this information to display detailed machine information to the user. This category has the ability to open Safari with a URL to an online PDF-document including schemes. Third category, network settings, can be adapted before the network connection is set up. The networking component of the application will read this information. This information includes the IP-address of the server, but also with which frame rate and image quality pictures needs to be sent to the client application. The category network information displays the status of the connection to the user. This information is displayed with text and requires the networking component of the application to update with this information. The start/end call functionality is a button that the user can press to start or end a call to the assisting technician.

Like with previous black box diagrams, all yellow components are only included within the application for the technician preparing or executing PIPO tasks. Blue components are only included within the application for the assisting technician. The communication named ‘sign-in name’ is colored blue because the sign-in name for the application of the ‘regular’ technician is included within the update of draw settings. This update will be send towards the augmented reality tracking part of the application instead of the networking part (see figure 6.5).

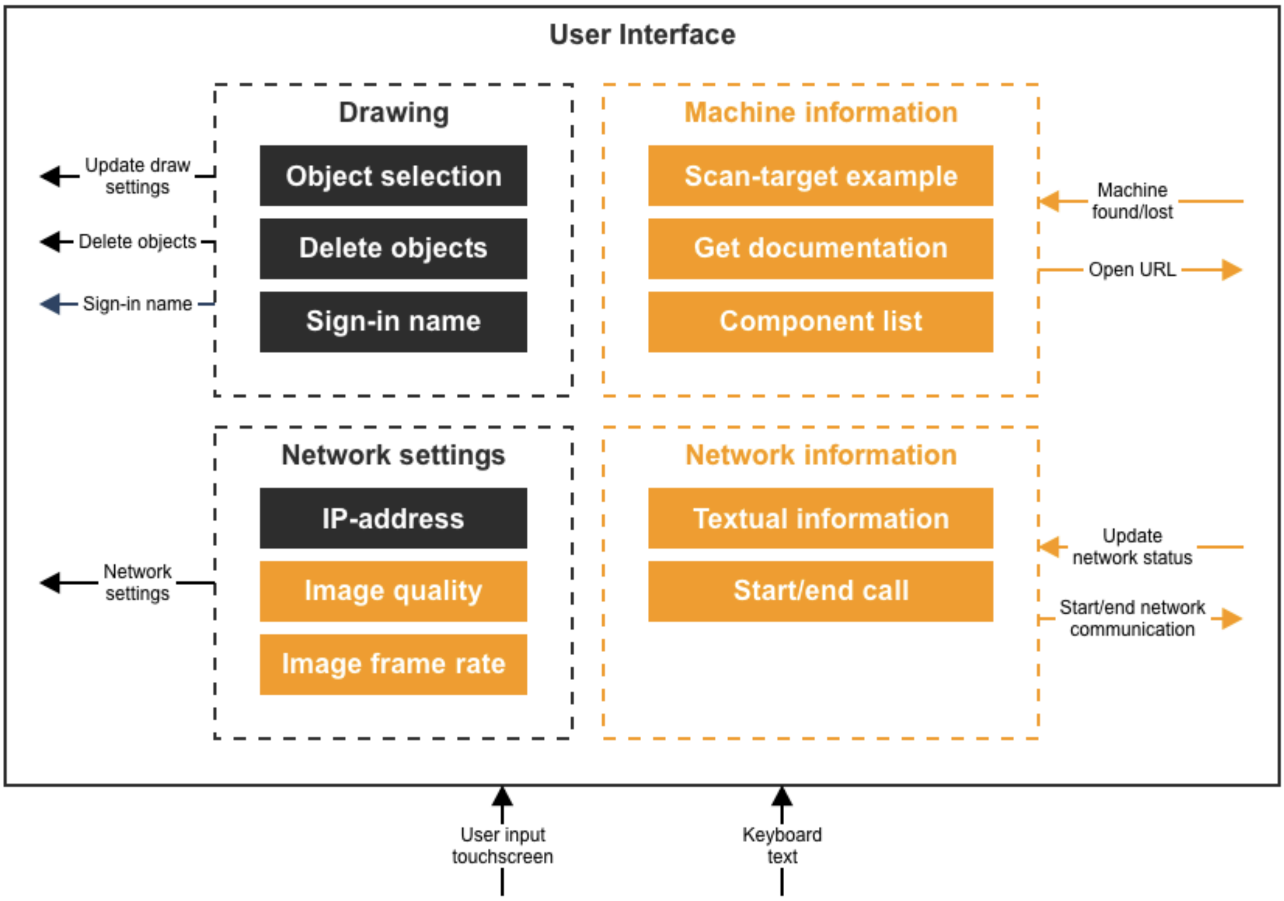


Fig. 6.6 - Black box 5: all functionality included in the user interface

Augmented Reality Tracking

In figure 6.7 it can be seen which components and functionalities are included within the augmented reality tracking part of the product's software. The iPad that is controlled by the PIPO executing technician has a component which tracks the 3D-environment around the user. Since the software is tracking a 3D-world from a camera view, this is done visually. When this part recognizes the world perspectives it generates new world coordinates which the application core can use to adjust the virtual world to align to the real world that is around the user. In addition to the world perspectives, the tracking software also recognizes the machine area that is on the camera view. When a machine area is recognized, the software will return the machine number as output to other components of the application.

The other half of the augmented reality tracking part is responsible for instantiating the virtual objects like circles and warning messages. As input it receives coordinates of where the user is pressing the touchscreen as well as drawing settings. These settings include which object to instantiate, what color it should be and what author name to draw on a new warning message. This part of the software is also responsible for deleting virtual objects and, if the iPad is controlled by the assisting technician, sending generated objects over the network to the other iPad. Last, when an iPad receives objects over the network, it should instantiate these as well. Because virtual objects should only be visible at one machine, and disappear with other machines, all objects are associated with a machine number when instantiated. Therefore, it requests the tracking component for the actual machine number.

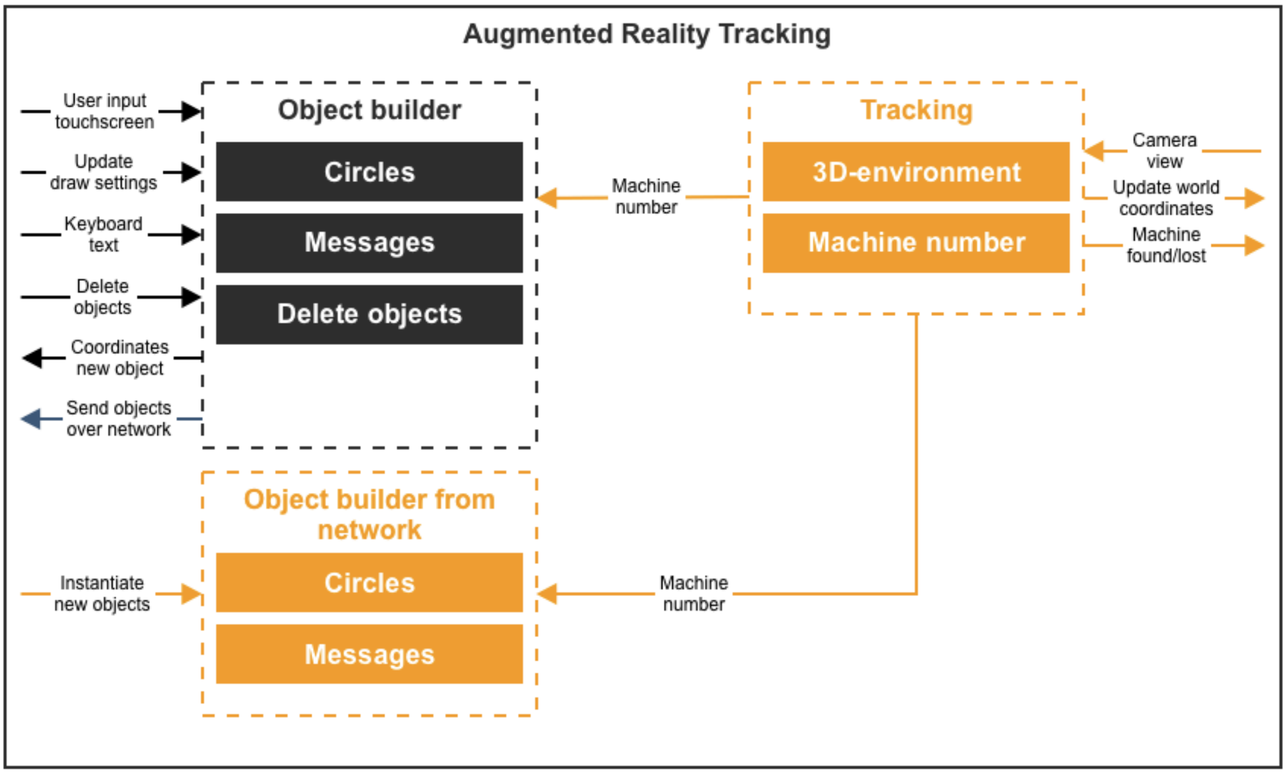


Fig. 6.7 - Black box 6: all functionality included in the augmented reality tracking

Networking

The networking component of the software application consists of two parts: a network builder, responsible for network connection so that messages can be transmitted, and network messages, which is responsible for what is included in the network messages. The application that is used by the assisting technician is always the server, waiting for clients that would like to connect, so that a video call can be started. This process requires network settings, such as an IP-address, and needs to be updated when the network status changes. The client is able to start and and the connection at every moment.

According to the status of the network, the network messages part should send certain messages to the connected device. When a client just connects to the server, the client sends its screen resolution to the server. With this information the server can adjust the coordinates of new objects to fit the screen size of the client. The server then sends the name of the technician to the client, so that the client can include this name onto the warning messages that it receives from the server. After this process, the client starts sending video images to the server with a frame rate and quality set by the user. When the server receives these images, it sets these images to the background of the application. Simultaneously, the server sends messages of generated objects by the assisting technician. The client receives the coordinates and the content of objects and let the augmented reality tracking instantiate these. Content can be the color of the object, when the object is a circle, or text, when the object is a warning message. All network messages are transmitted via the application core, which communicates with the operating system of the iPad.

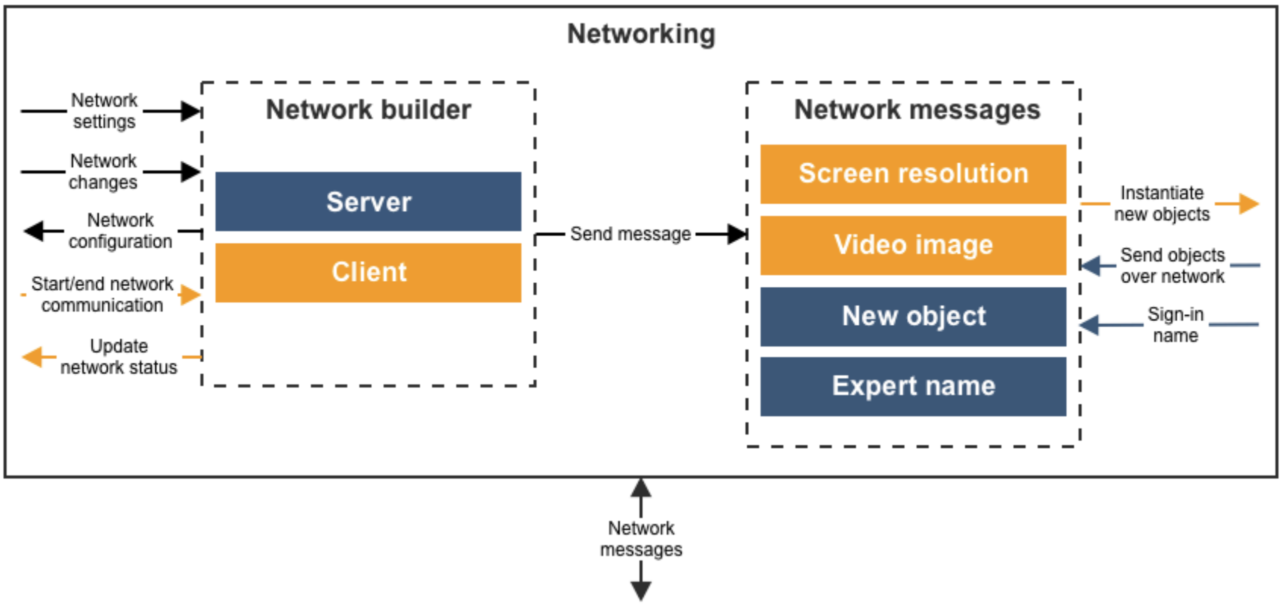


Fig. 6.8 - Black box 7: the two components of the networking part within the application

6.2 Implementation

6.2.1 Programming software

In order to build an iPad application, an existing programming framework is necessary for developing the application. Since an augmented reality application requires development in a 3D-environment, it is chosen to develop the application with a framework that already includes working with 3D-environments. Since the software Unity3D is development software that enables developing in 3D, and includes customization with C# scripting, this program is chosen as framework for the prototype. In addition, this software is widely used, meaning documentation is available on most aspects of the program.

Unity3D offers a clickable interface to create, place and adjust virtual objects within the 3D-environment, as well as C# scripting to adjust the objects' behavior. Before a Unity3D project can be installed on an iPad, the project should be exported to a Xcode-project first. Xcode is the standard coding software of Apple, and is able to install development projects on Apple devices. In order to export the Unity3D project, Unity3D offers the library "Simple Mobile Placeholder" that automatically exports the project [22].

6.2.2 User Interface

Since non-functional requirement R1 stated that the user interface should be clear and easy to understand, it is chosen to base the interface on mainly visual objects. This requirement was based on the observation that technicians rather worked visually than with reading. Beneath, the implementation of all four categories within the user interface are discussed. C# code of the classes used by the user interface components can be found in the appendix C.1.

Drawing settings
First, object selection is possible on the upper left corner of the screen (see figure 6.9). This functionality received this prominent place, because it is expected that these buttons are pressed most. For the selection of a color (red, green or blue) or a warning message, the user should press the corresponding button to select the object variation. In order to prevent accidentally pressing the delete button, this functionality is placed on the upper right corner of the screen.

Fig. 6.9 - Object selection buttons and the delete button

The signing in functionality looks different in both applications. In order to force the assisting technician to sign-in, the application opens on a sign-in screen first, which can be seen in figure 6.11. It is necessary that the user signs in to prevent confusion about who created what messages. Since the regular application will also be used in situations that the user will only ‘read’ the machine signals, it is not required to sign-in. This asks for a different way to sign-in. Therefore, the regular application has a pop-up functionality at the right side of the screen (see figure 6.10) that the user can use at all moments.

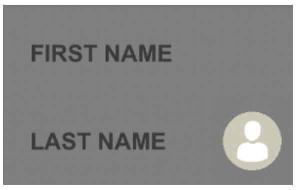


Fig. 6.10 - Sign-in functionality on the regular application

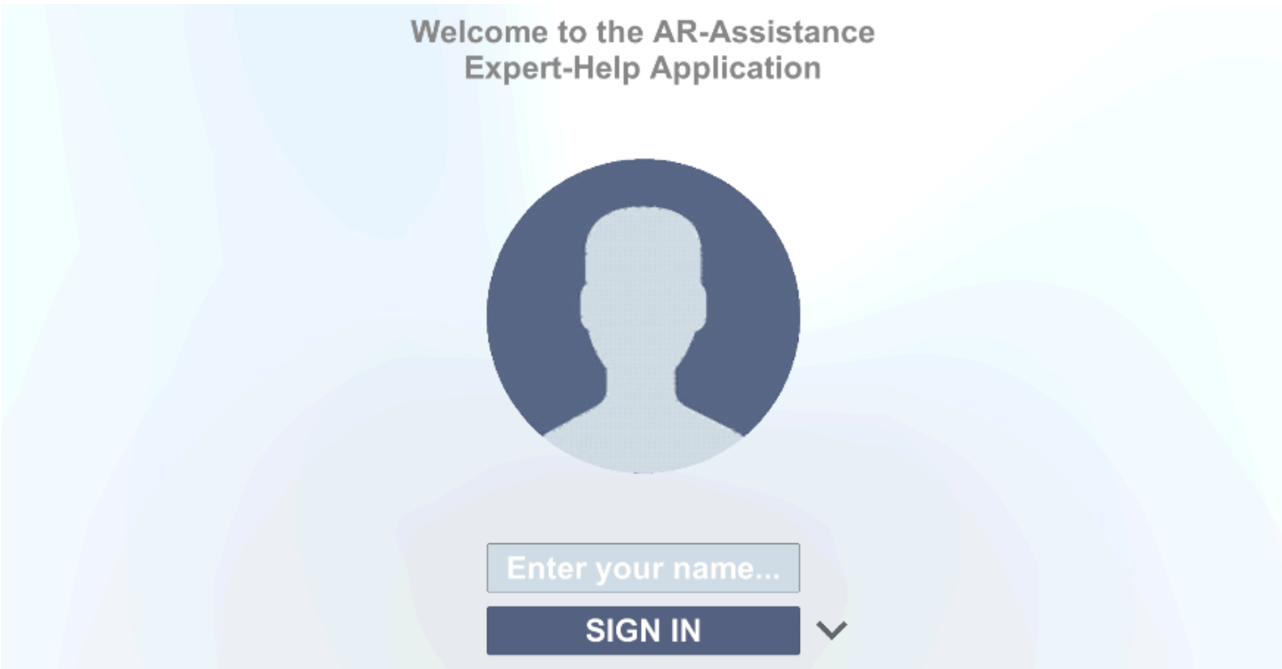


Fig. 6.11 - Sign-in functionality on the assisting application

Machine information

As stated in non-functional requirement R1, the user interface should be easy understandable. Therefore, it is decided that is should be specifically mentioned to the user to scan a target object. This message, displayed in figure 6.12, is present at all times when the software is not recognizing a machine area in the camera view. The red icon is an example of how a target object looks like so that the user will know for what visual object to search in its environment.

Fig. 6.12 - Message indicating the user to scan a target object

In the bottom left corner, the application shows machine information of the machine area that is recognized in the camera footage. From the functional requirements R10.1 and R11 is is required that the software is able to show relevant working documents from a machine area and to show a list of components that the machine area includes. When no machine is found, the document button turns grey and a text will display “No machine found” (figure 6.13.a) . Whenever the software recognizes a machine area, the ‘show component’ button becomes visible, and the document button turns blue. Additionally, the name of the machine area is displayed in text (figure 6.13.b). If the user presses the ‘show components’ button, a list of all components that are within the machine area is shown (figure 6.13.c).

a. b. c.

Fig. 6.13 - Machine information when (a) no machine is found, (b) a machine is found and (c) the user has pressed the ‘show components’ button

Network settings

As described in the previous section, three variables can be adapted for the network connection: the IP-address of the server, the image quality and the image frame rate. The last to will affect the images that are send from the camera footage. Since these settings are only required whenever a connection should be set up, this functionality looks different for both applications. Within the assisting application the IP-Address is set as localhost (127.0.0.1) as default and can be edited under the sign-in name (see figure 6.14). Just like the signing in functionality the network settings can be accessed in a pop-up window at the right side of the screen. These settings can be adjusted whenever no network connection is already made (see figure 6.15). To prevent errors, also these variables are set to defaults. The IP-address is set to localhost, the image frame rate to once every 10 frames and the image quality to four (on a scale from one to one hundred).

Fig. 6.14 - Network settings in the assisting app

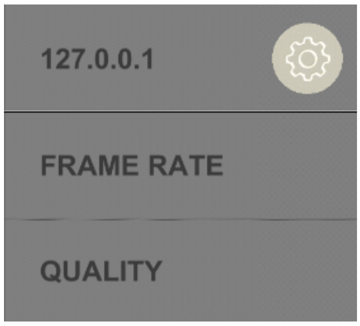


Fig. 6.15 - Network settings in the regular app

Network information

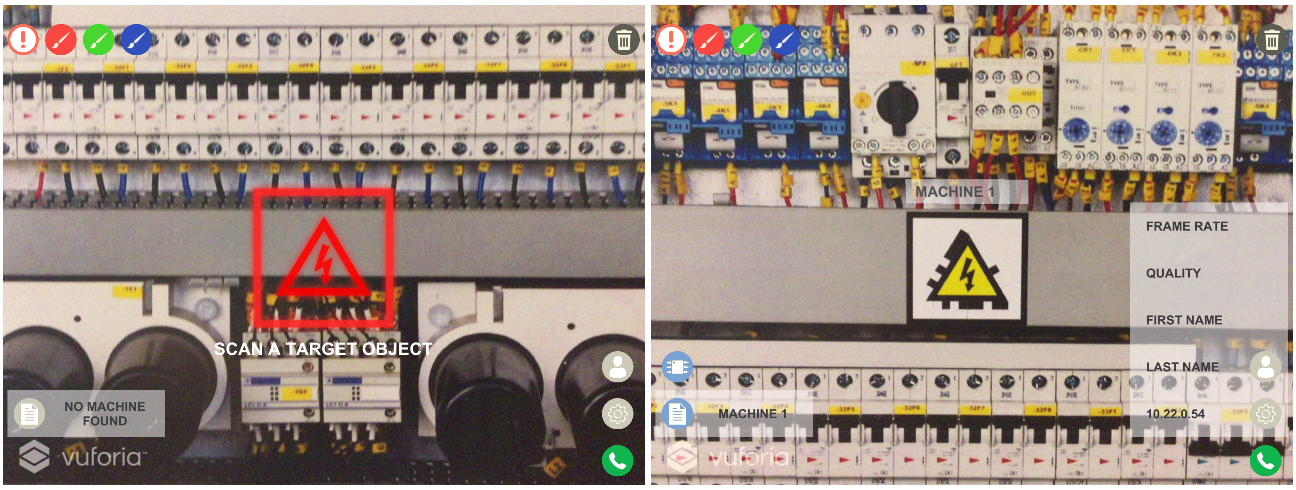
The network information section gives the user feedback on the status of the network connection with the assisting technician. This section can be found in the bottom right corner of the screen. When no connection is made, the network information only shows a green call button (figure 6.16.a). When the user presses this button, the software starts a client and tries to connect to the given IP-address (figure 6.16.b). Textual information will then display that the software is 'calling' the technician. When the connection is successfully set, the network status changes to 'connected with expert', together with a red call button (figure 6.16.c). The user can press this button to end the network connection with the assisting technician. Whenever the server is not available or the wrong IP-address is given, the network information displays the message 'no experts available' (figure 6.16.d).

Fig. 6.16 - Four stages of network information

Screens

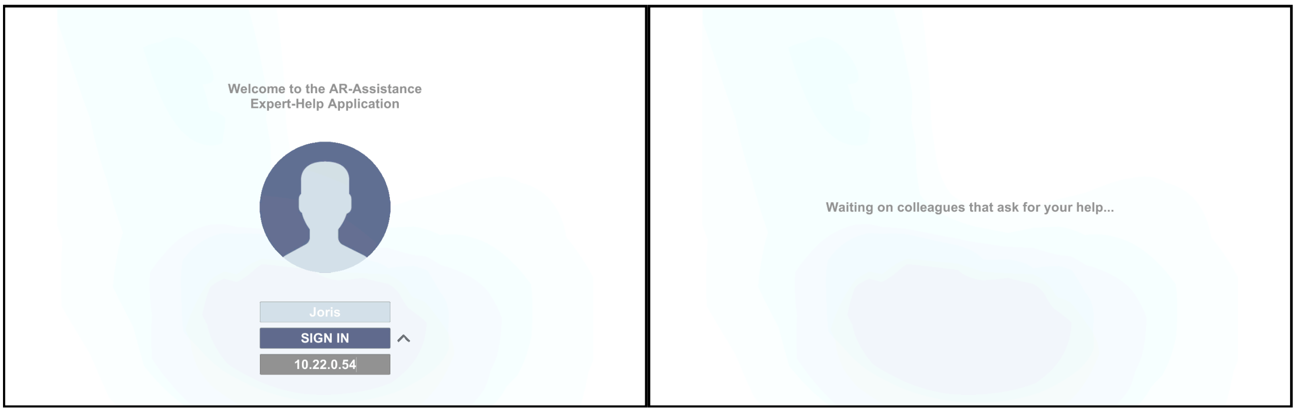
All user interface components combined result in the screens given in figure 6.17 and 6.18. In figure 6.17 a screen is given of the regular application. It can be seen that most items of the user interface are made transparent. Because the camera view is an important element of the application, it is decided that the user interface must interrupt minimally with this view, while user interface items should remain easy accessible. Additionally to the transparent backgrounds, these items are pop-up items. When the user presses a button including an icon, like 'settings' or 'account', a transparent menu will pop-up in which the user can adjust the variables. Figure 6.17.a gives the user interface with the pop-ups folded, where the unfolded pop-ups can be seen in figure 6.17.b.

Figure 6.18 shows the flow of the assisting application. As described earlier, the signing in functionality is present in a separate screen (figure 6.18.a). After signing in, the application shows a screen which indicates that the user should wait to be called by a colleague (figure 6.17.b). When a network connection is set, a user interface will be displayed to draw virtual objects together with the camera footage from the other device (figure 6.17.c).



a. b.

Fig. 6.17 - The regular application with (a) pop-up menu items folded and (b) pop-up menu items unfolded



a. b.

c.

Fig. 6.17 - The assisting application when (a) signing in, (b) waiting for colleagues that ask for assistance and (c) a colleague that is connected

6.2.3 Augmented Reality Tracking

Tracking: 3D-environment

The most challenging aspect of augmented reality is tracking. Because the prototype obtains camera footage to which it should align virtual objects, a reliable way to achieve an augmented reality experience is with tracking visually. This means that software is processing camera footage, and maps the 3D environment that is filmed by the device. Since this project has its focus on how to use an augmented reality tool rather than the focus on how to track a 3D environment for augmented reality purposes, it has been decided to use external software for the augmented reality tracking. A company that provides a reliable library for Unity3D is Vuforia. From their website packages can be downloaded, which can be implemented within the Unity3D project. Additionally, Vuforia includes a wide variety of documentation which could make a development process quicker.

The Vuforia library is able to recognize five kinds of real world objects:

1. 2D-images, uploaded to the Vuforia-engine. Uploading can also be done via an IPA-call, meaning this could be done automatically.
2. Small 3D objects, scanned with the Vuforia Object Scanner. This is an application for the Samsung Galaxy S5 and the Google Nexus 5 [21]. After scanning the object, the scan should be uploaded to the Vuforia website.
3. Cilinders.
4. Cubes.
5. VuMarks, which are 2D-images that can be displayed in the real world with a small size and can include digital information like a string or a number of bits.

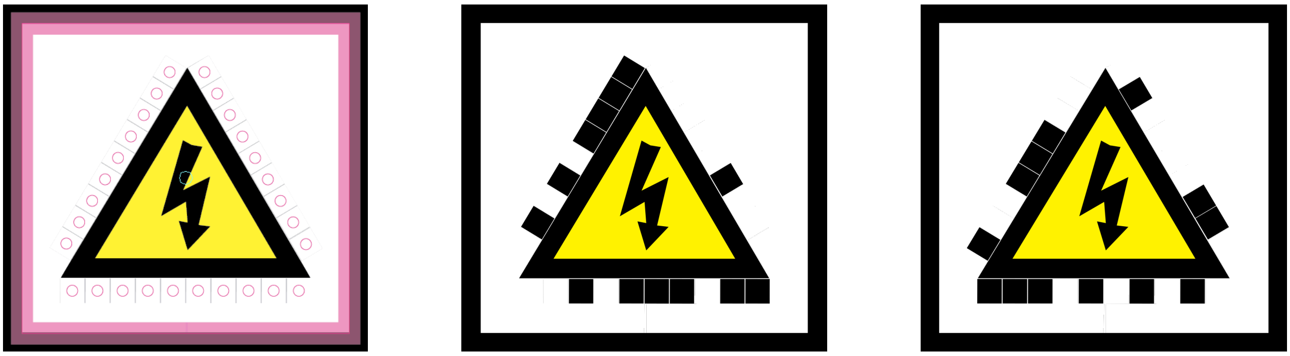
Since the product should fit as easy as possible into the daily work of the technician. This means that adding additional scan objects to the technicians' environment would include additional practices for ordering and placing scan objects to machine areas. In theory it would be ideal when the software could recognize and track 3D perspectives from the machines. However, with the Vuforia library it is only possible to scan small real world objects. Additionally, the scanning of 3D objects is not the most easy practice and therefore less reliable to include in the technicians daily practices. A solutions that includes both aspects is that the software is tracking the warning stickers that are already included in most machine areas. These warning could for example be warnings for electrical shocks or messages to wear earplugs. Warnings that were used during prototyping with this way of tracking are displayed in figure 6.19. Because the warning given in figure 6.19.a did not include enough recognizable points for the Vuforia software, experimenting is executed with recognizable patterns behind the warning, like shown in figure 6.19.b. This appeared to be reliable enough to track on 3D perspectives.

a.

b.

Fig. 6.19 - Warning messages that are (a) not reliable for tracking and (b) reliable for tracking with Vuforia

Tracking: machine number
Although the warning messages are recognizable with the Vuforia software, all stickers and thus all machine areas look similar for the software. This resulted in virtual objects that were drawn at a certain machine and remained visible when another machine was scanned afterwards. A way to include information within a 2D-trackable object in the Vuforia library are VuMarks. These objects work like QR-codes, but include more visual information. An Adobe Illustrator project that is provided by Vuforia let developers create their own VuMarks. Within these VuMarks a required amount of elements should be included that can be colored white or black. The combination of certain elements being white or blue indicates the software with what string or number it corresponds. Because strings require a large number of elements, and only machine numbers are necessary to be read from the VuMarks, it is decided to include only numbers within the element combinations. Within figure 6.20.a it can be seen how the VuMark looks that is used for this project. At the positions of the pink circles the Vuforia database can generate white or black blocks. A combination of these 28 blocks indicates a bit between one and ten to the tracking software. For example, figure 6.20.b corresponds to the number one, while figure 6.20c corresponds to the number two. The C# code that reads the number from a recognized VuMark is given in appendix C.2.



a. *b.* *c.*

Fig. 6.20 - The VuMark that is designed for this project

Object builder
When the user of the application touches the screen, and is not pressing a user interface element, the software draws a virtual element to the 3D environment. Depending on what the user has selected as drawing object and color, the software draws a “prefab” to the position that the user presses. When a circle is selected as drawing object, the software remains drawing circles that will form lines eventually. When a message is selected as drawing object, the message prefab is only instantiated when the user clicks the screen. This means that messages will not remain being created while the user holds its finger on the screen. With the Unity3D standard function “ScreenToWorldPoint”, the software can calculate the finger touch to a position in the 3D environment. Because it is expected that all objects should align with the machine, the virtual objects are instantiated at a distance that the VuMark is from the camera.

Whenever a new object is created, the software creates the prefab as a child of the VuMark, meaning that it will move together with the tracked VuMark in the camera view. This causes that all virtual objects will remain aligned with the machine, and will create an augmented reality experience. In order to only display virtual objects at one machine, instead of at all machines, the virtual objects are associated with only one machine. When an object is created, its name is changed to start with the machine number. For example, if the user creates a blue circle at machine one, the software will change the circle’s name to “1-BlueCircle”. At all times, the software is using a script that only displays the virtual objects which names start with the same number as the recognized machine area. Additionally, when the user presses the delete button, only the virtual objects that start with the same number as the recognized machine area are being deleted. C# scripts of these functionalities can be found in appendix C.2.

Object builder from network
When the application is connected to the assisting application, it can receive virtual objects from the other device. These objects are handled the same way as objects that are created by the user, as described in the previous paragraph. However, the coordinates of the received objects are being adapted to fit the screen size first, before these objects are instantiated. How this is done is explained at the networking paragraph.

An example of how virtual objects are displayed on in the application is given in figure 6.21. This image includes objects that are created by the user as well as by the assisting technician. Additionally, it can be seen that the software uses the corresponding author name, together with the actual date and time, so that the user could identify the messages. C# scripts that are designed for these functions are given in appendix C.2.

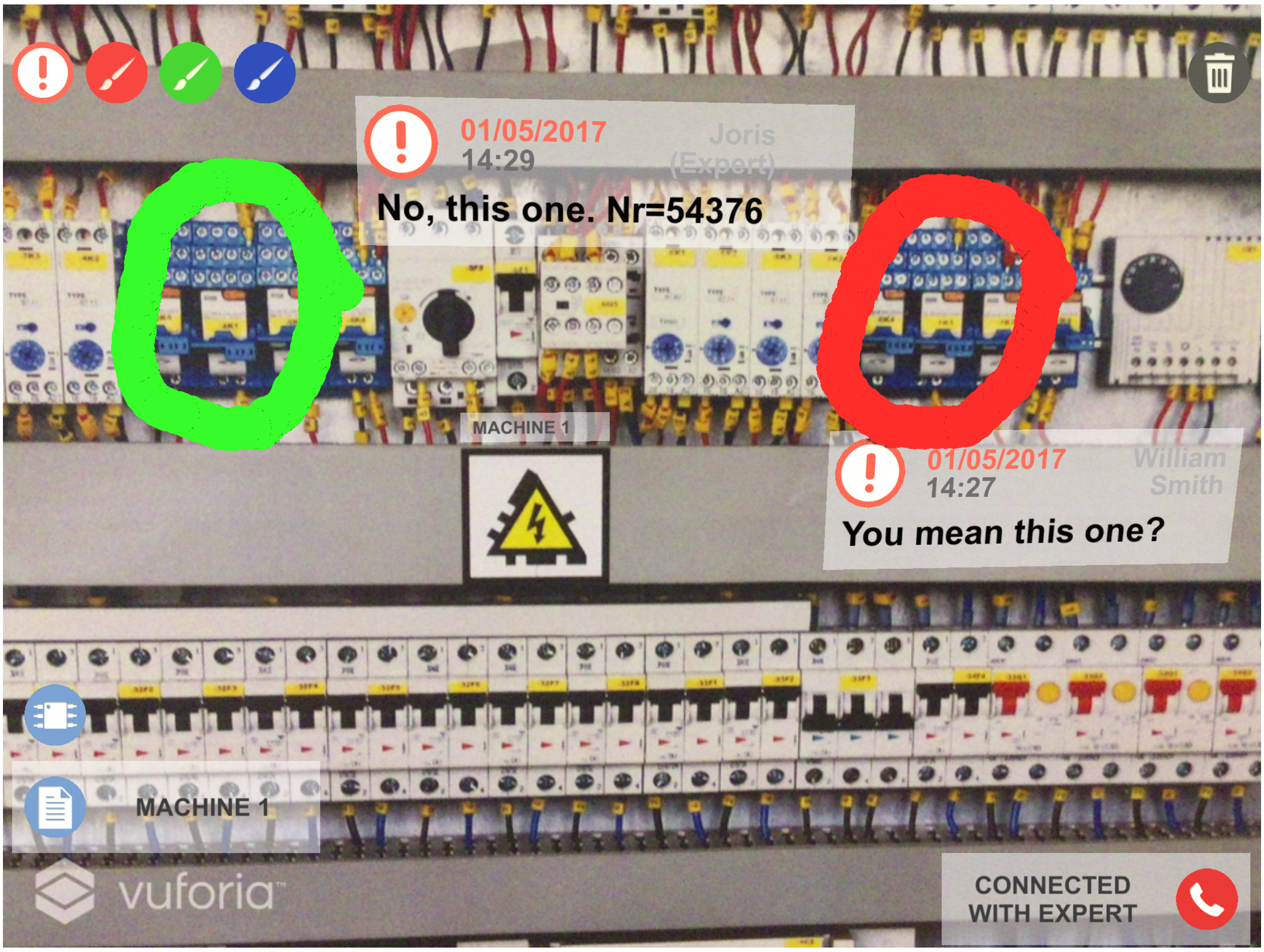


Fig. 6.21- Screen of the application including virtual objects from the user and the assisting technician

6.2.4 Networking

Network builder
Unity3D offers multiple options for networking, of which the "Network Manager", a visual interface for networking and multiplayer games, is currently the recommended option by Unity3D [28]. However, this option for networking is based on games that includes multiple players to walk around in the same 3D environment. Although theoretically a multiplayer could only consist of a camera that is not moving, no way was found in which this Network Manager could successfully operate together with the Vuforia library. Therefore, an older method for network communication between Unity3D applications is chosen. This method includes a simple Server-Client protocol. One application opens a server at its IP-address and a free port, where other applications connect to with opening a client port at the device. Although Unity3D reports this networking method as

obsolete, it offers a stable network connections with enough functionality for building a prototype. Whenever both the regular and the assisting application are connected to the internet, regardless of their location, a functioning network connecting can be established between the two.

Network messages
When the network connection is set, messages can be sent between the connected devices, which asks for a messaging method. A widely used messaging method in other Unity3D applications using the Server-Client protocol described in the previous paragraph is 'Remote Procedure Calls' (RPCs). As explained in [29], it is possible to call a function on the connected device and to transfer multiple variables over the network. These variables will be used by the called function, and can include integer, floats, strings, bytes, byte arrays, NetworkPlayers, NetworkViewIDs, Vector3s and Quaternions. A restriction to this method is that whenever an application calls a function on the other device, the function must also be present within the script that is instantiating the RPC.

As described within the black box of the networking part, four calls over the network are required to build the prototype: screen resolution, video image, new object and the expert name. Beneath, it is per network call explained how these are build as a RPC. The full C# code of the networking class, including the server-client functionalities, can be found in appendix C.3.

Screen resolution
In order to ensure that new objects are positions at the same location on the receiving application as the transmitting application, the screen resolutions should be synchronized. When a client connects to the server, the server sends an RPC including its own screen resolution.

```
public void OnPlayerConnected(NetworkPlayer player) {
        nView.RPC ("SynchronizeScreenResolution", RPCMode.Others, float.Parse(Screen.width.ToString()),
        float.Parse(Screen.height.ToString()));
}
```

The receiving applications, receives this RPC with to floats as variables. This client will save both vertical and horizontal factors, which represents how much larger the screen of the transmitting application's device is.

```
[RPC]
void SynchronizeScreenResolution(float xResolution, float yResolution) {
        xResolutionFactor = xResolution / Screen.width;
        yResolutionFactor = yResolution / Screen.height;
}
```

Video image
Since it is not possible to transmit a full image, an image from the camera view should be converted to a byte array first. Whenever a full camera image is rendered on the regular application, `OnPostRender()` is called. If the application detects a network connection, the software will capture the texture from the screen of the application (including user interface details), and stores this image in a temporary texture. Then the function reads the given image quality from a user interface element. It encodes the image with as a JPG in a byte array with the set image quality. In contrast to PNG, the JPG file format does not store the alpha-channel, which causes the file to be smaller and easier to transmit over the network. Last, the video image is transmitted towards the assisting application with a RPC. The temporary texture and the byte array are removed and emptied in order to save memory.

```
void OnPostRender() {
    if (Network.isClient && sendImage) {
        sendImage = false;
        Texture2D screenshotTexture = new Texture2D (Screen.width, Screen.height, TextureFormat.RGB24, false);
        screenshotTexture.ReadPixels (new Rect (0, 0, Screen.width, Screen.height), 0, 0);
        screenshotTexture.Apply ();
        int imageQuality = 4;
        if (ImageQualityField.GetComponent<InputField> ().text != "") {
                imageQuality = int.Parse (ImageQualityField.GetComponent<InputField> ().text);
        }
        byte[] screenshot = screenshotTexture.EncodeToJPG (imageQuality);
        nView.RPC ("SynchronizeWebcamVideo", RPCMode.Others, screenshot);
```

```
        DestroyImmediate (screenshotTexture);
        screenshot = null;
    }
}
```

When the assisting application receives a RPC on this function, it loads the received byte array as an image on the texture of the large background plane. Additionally, the script empties the received byte array to save memory, and the software removes all virtual objects that are drawn by the user.

```
[RPC]
void SynchronizeWebcamVideo(byte[] videoImage) {
        backgroundTexture.LoadImage (videoImage);
        GetComponent<Renderer> ().material.mainTexture = backgroundTexture;
        videoImage = null;
        buildEnvironmentScript.RemoveAllServer ();
}
```

New object

The assisting application is able to send the coordinates and content of new objects towards the regular application via a RPC. In the `Update()` function, the script examines whether the application is connected and whether more virtual objects are drawn then objects that are sent. If so, the script sends a different RPC for different colors of drawn circles including the horizontal and vertical position of the users finger on the screen. Because a circle is immediately sent after the user draws these, this position is the same as the position of the circle. When the drawn object is a message, the script waits until the keyboard with which the user types the message is not active anymore before sending the object over the network.

```
void Update() {
    if (Network.isServer && connected) {
        if (this.gameObject.transform.childCount > amountChildsSend) {
            if (this.gameObject.transform.GetChild (amountChildsSend).tag == "RedCircle") {
                nView.RPC ("SetNewRedCircle", RPCMode.Others, Input.mousePosition.x, Input.mousePosition.y);
            }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag == "BlueCircle") {
                nView.RPC ("SetNewBlueCircle", RPCMode.Others, Input.mousePosition.x, Input.mousePosition.y);
            }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag == "GreenCircle") {
                nView.RPC ("SetNewGreenCircle", RPCMode.Others, Input.mousePosition.x, Input.mousePosition.y);
            }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag == "MessageObject") {
                if (!buildEnvironmentScript.IsKeyboardActive ()) {
                    Transform messageObj = this.gameObject.transform.GetChild (amountChildsSend);
                    Transform messageDescription = this.gameObject.transform.GetChild (amountChildsSend + 1);
                    SendWarningMessageToClient (messageObj.localPosition.x, messageObj.localPosition.y,
messageDescription.gameObject.GetComponent<TextMesh> ().text);
                }  else {
                    amountChildsSend = amountChildsSend - 1;
                }
            }
            amountChildsSend++;
        }
    }
}
```

The regular receives one of the RPCs to instantiate a new virtual object, and notices the object builder of the augmented reality tracking part to create this object. It addresses the right prefab to the RPC, and passes this as variable towards the object builder script. The two other variables that are passed towards the object builder script are the horizontal and vertical position of the object on the screen. Since screen resolutions can differ between the regular and the assisting application, these positions are multiplied by the factors that are calculated at the start of the network communication.

```
[RPC]
void SetNewRedCircle(float xpos, float ypos) {
    buildEnvironmentScript.DrawObjectFromRCP (RedCircle, (xpos/xResolutionFactor), (ypos/yResolutionFactor));
}

[RPC]
void SetNewGreenCircle(float xpos, float ypos) {
    buildEnvironmentScript.DrawObjectFromRCP (GreenCircle, (xpos/xResolutionFactor), (ypos/yResolutionFactor));
}

[RPC]
void SetNewBlueCircle(float xpos, float ypos) {
    buildEnvironmentScript.DrawObjectFromRCP (BlueCircle, (xpos/xResolutionFactor), (ypos/yResolutionFactor));
}

[RPC]
```

```
void SetNewMessage(float xpos, float ypos, string message) {
    buildEnvironmentScript.DrawMessageFromRCP ((xpos/xResolutionFactor), (ypos/yResolutionFactor), message);
}
```

Expert name

The name of the expert is transmitted at the start of the network connection via a RPC. When the regular application receives this name, stored in a string, the script stores this information globally.

```
public void OnPlayerConnected(NetworkPlayer player) {
        nView.RPC ("SetNameOfExpert", RPCMode.Others, authorName);
}

[RPC]
void SetNameOfExpert(string name) {
        buildEnvironmentScript.SetExpertName (name);
}
```

7. Evaluation

During this phase of the project, the product that is created in the realization phase is examined towards goals and requirements that are set in the specification phase. In addition, goal of the evaluation is to answer the research's fourth subquestion. The evaluation phase consists of two parts. Functional testing, examining whether all 'Must' requirements are fulfilled by the prototype, and user testing, examining the added value for the technicians of Apollo Vredestein when using the prototype in real use situations.

7.1 Functional Testing

7.1.1 Internal requirements testing

Before the product requirements are evaluated with end-user, functional requirements are evaluated internally, meaning by the researcher itself. Goal of this test is to see whether the most important functionalities of the product, described in the specification phase, are included in the prototype. Additionally, this test shows which functionalities are interesting for future work. In table 7.1 all product requirements are listed, together with an indicator whether the requirement is achieved by the prototype. This indicator is measured by using the prototype, evaluating whether the requirement is met.

No.	Requirement	Prio.	Achieved
R1	The user is able to see its environment through the device, possibly by camera view	Must	Yes
R2.1	The software recognizes the user's environment and real world perspectives	Must	Yes
R2.2	The user is instructed to scan an object that the 3D engine can recognize	Must	Yes
R2.3	Feedback is provided when the user's environment is recognized	Must	Yes
R3.1	The user is be able to draw figures, like lines and circles, on the view	Must	Yes
R3.2	The user is able to add objects with three different colors: red, green/yellow and blue	Must	Yes
R3.3	The drawer figures stay aligned with the user's environment	Must	Yes
R4.1	The user is be able to add text-messages to the view	Must	Yes
R4.2	When a text-message is displayed, date and time of creation is given together with the name of the author	Must	Yes
R4.3	The added text-messages stay aligned with the user's environment	Must	Yes
R5	The user is be able to remove figures and text-messages from the view	Must	Yes
R6.1	The software is be able to identify specific parts of the machine	Must	Yes
R6.2	The software only shows figures and text-messages that are relevant for the identified part of a machine	Must	Yes
R7.1	The user is be able to share the camera view with the user of another device	Must	Yes
R7.2	The user of the other device is be able to draw figures, add text-messages and remove these to/from the views of both devices	Must	Partially

R7.3	The user is able to stop sharing the camera view with the user of the connected device	Must	Yes
R8	The devices automatically synchronize the figures and text-messages, so that all information on parts of machines is the same for both users	Must	Yes
R9	Both users should be able to talk to each other, like being on the phone	Must	Partially
R10.1	When a machine-part is recognized, the software can show a list including all components	Should	Partially
R10.2	In the list, for all components relevant information can be shown, including name/type, manufacturer, warehouse availability, technical scheme and workings of the component	Should	No
R11	The user is able to obtain relevant working documents to a component or machine part	Should	Yes
R12.1	The software is able to identify different components within a part of the machine	Could	No
R12.2	The software can show the components names	Could	No
R12.3	Other relevant component information can be shown, including manufacturer, warehouse availability, technician scheme and workings of the component	Could	No
R13	The user is able to order a component from the application	Could	No
R14.1	The software shows live information from the machine	Could	No
R14.2	The live information of the machine is aligned the the right components in the machine	Could	No

Table 7.1 - Evaluation functional product requirements

From the results of table 7.1 it can be concluded that all requirements prioritized with ‘Must’ are mostly implemented in the prototype. The exceptions to this are R7.2 and R9. During the design process it became clear that it would be of no advantage if it was possible for the helping technician to remove all objects from the other technicians view. Especially in cases in which this would happen accidentally, it would be a disadvantage. The other part of the requirement, including drawing objects and messages on the view of the other device is included in the prototype. R9 is partially achieved, since the FaceTime application on the iPads handle the speech communication. Because of time limits of this project it was decided to implement functionality that is not easily available by third party applications. During the user evaluation the technicians can hear each other over FaceTime communication.

R10.1 and R10.2 are partially or not included in the prototype, since these played no role in the user scenarios described in the specification phase. In addition, such functionality is technically not challenging, and therefore it is chosen to focus on the augmented reality and networking part of the system. Although R11 is not technically challenging, it appeared a ‘Must’ instead of a ‘Should’ priority during the realization process. R12.1 - R14.2 are technically challenging, but appeared not to fit within the scope of this project. Research on the topics ‘tracking machine on component level’ and ‘display a machine’s live information with augmented reality’ have large potential for future work.

7.1.2 Requirements evaluation with technician

Since this project involves participatory design, it is considered important to evaluate the functionality together with the technician that participated in the product design (co-designer). In this evaluation the co-designer was asked about the completeness of the present functionality as well as about missing functionality. His reference for this discussion are the conclusions from earlier interviews and the list of product requirements. Since the development of the prototype is completed, the co-designer can rely his comments on experience with the functionality rather than imagination of functionalities. Goal is to have an 'expert-eye' looking at functional requirements testing, and whether the prototype is completed.

This part of the evaluation is executed with a semi-structured interview with the technician that participated in the design during the specification and realization phase. It this way, this technician has a good understanding of the design choices that are already made. Before the interview, the prototype's functionalities are briefly explained and the technician has ten minutes to try these functionalities with the prototype. Topics during this interview are 'present functionality', 'missing functionality' and 'priority of the missing functionalities'.

The co-designer confirms that most important functionalities are implemented in the prototype. He did not have any comments on the functioning of the available functionalities. However, the technician gave eight ideas of new functionality, that could be included in the product. These suggestions are listed, including their priorities, in table 7.2.

Functionality	Prio.
The expert technician has the possibility to 'freeze' a frame, so that this user does not have to draw on a shaky camera view.	Must
The ability to remove (a part of) one drawing, instead of all elements, also for the expert. Otherwise, communication will become messy with a large number of drawings and messages.	Must
The system should be linked to the DMS-system for documentation. It his way, documentation is always synchronized with the current system.	Should
The software can recognize a closet when this is not opened yet, so that the user can be sure that he or she is standing in front of the right closet for a certain maintenance task.	Should
An overview of all messages that are drawn in a machine area can be found under a 'history' tab. this gives technicians a wider context to defects.	Should
The user can search with a search field on a specific component in the machine area.	Should
To improve the efficiency of the technician finding the right component in a machine, row-number of the component can be displayed in the software. On the machine stickers can be pasted including these row-numbers.	Could
When the infrastructure of the internet is poor, the system could only send the machine-ID to the expert. Then, the expert could draw on a static image of that part of the machine, which will be transformed and displayed on the other device, aligned to the machine.	Could

Table 7.2 - Functionalities suggested by the end-user with their priorities

When looking at the results from this evaluation with the co-designer, it can be concluded that the prototype is almost ready to be tested in real use situations: the user evaluation. The six suggested functionalities that received a 'should' or 'could' prioritization, will be included in future work. These are not necessary for the user evaluation, and do not serve the main focus of this research. However, the fourth suggested functionality about the recognition of closets can already be tested with the prototype. When using the scan-objects as identifiers for all different closets, the software can identify whether the user is scanning the right or the wrong closet for a task.

Next to six functionalities that are prioritized as 'should' or 'could', two 'must' functionalities are suggested by the co-designer. First, a 'freeze' functionality with which the helping technician can freeze a frame of the camera footage from the other device. This causes that the helping technician can draw on a static image, instead of drawing over a shaky and moving video. Since testing this functionality showed that this functionality is necessary for creating clear drawings, it is implemented for the user evaluation. Second, the co-designer suggested a functionality in which both users could remove specific parts of the drawings and messages. When one of the users makes a mistake, it should be possible to remove this to prevent messy and unclear virtual drawings. However, when only a functionality to remove all objects is available, also important information might be removed. Although this functionality is prioritized as a 'must' for the product, it is decided to be not implemented in the prototype. Reason for this decision is that the development of a functionality that let the user remove specific objects would ask for a structural change. Because drawings exist of many circular objects, a functionality must then group all circles that form one larger object together, e.g. an arrow. It would be available to remove a larger object at ones. It is decided that the implementation of this functionality is too time consuming to be finished before the user evaluation.

7.2 User Evaluation

Now the prototype is identified as ready for an evaluation in real use scenarios, internally and together with the co-designer, the user evaluation can be executed. In preparation for this evaluation, the required 'freeze' functionality is built in the prototype. How this is done is explained first. Afterwards, the evaluation plan is discussed. This plan includes a description of the chosen participants, the scenarios that these participants should execute, a set-up on how all elements in the evaluation relate to each other, and a plan on how to obtain results from the user evaluation. All these components are described in different chapters beneath.

7.2.1 'Freeze' Functionality

The functional test concluded that a 'freeze' functionality must be included in the prototype, before a good user evaluation can be executed. Therefore, it is decided to implement this functionality. In the view of the helping technician an additional button is created, which is only visible when the software is receiving camera footage of the other technician's iPad. When clicking this button, the device will notify the other iPad to stop sending images. This causes that the image received last will hold. Both iPads are now holding the same image, and also the tracking of Vuforia is stopped. The helping technician can draw and add messages to the view, which will immediately be send to the other device. Because the tracking of Vuforia is stopped, the virtual objects will be placed at the right position in the 3D-environment. In figure 7.1 it can be seen how this functionality looks.

Fig. 7.1 - The 'freeze' button, when the screen is unpinned and pinned

When the helping technician is done adding drawings and messages, the 'freeze-button' is clicked a second time. The software notifies the other device to start sending new images and to start the Vuforia tracking. While the 'freeze' functionality is on, the 'freeze-button' is colored blue in order to prevent misunderstandings about the freeze of the last image. When the 'freeze' functionality is turned off, the 'freeze-button' is colored grey. C# code of this functionality can be found in appendix C.4.

7.2.2 Evaluation Plan

In order to perform a reproducible user evaluation, a detailed evaluation plan has been defined. Variables can be controlled by the researcher then, so that only the necessary information is evaluated. Additionally, the evaluation is then verifiable. First, the decision for the chosen participants is described. Second, the scenarios are discussed that will be executed during the user evaluation. Third, an overview of all elements within the evaluation, and the relations between them, are given in a description of the set-up of this experiment. Lastly, methods are discussed on how to measure results from the user evaluation.

Participants
This research focusses on a qualitative approach rather than a quantitative approach. Also in the selection of participants this approach is present. It is decided as more valuable to evaluate the prototype with users that fully understand the product than with users that has few understanding of the product, even when this results in a smaller sample size. Important is that the participants of the user evaluation represent the larger group of end-users of the product, and that the way the participants use the product represent the normal scenarios of working of the end-users. Therefore, it is chosen to select two technicians that are working at Apollo Vredestein to participate in the user evaluation.

First, the most knowledgable technician of the 'Quadruplex' machine in the factory, which is also the technician that collaborated in design choices during earlier phases of this project, is participating in the user evaluation. It is important that he is participating, since the user evaluation is executed at the Quadruplex machine. He represents the group of users that are experienced to the software. Additionally, during the user evaluation it can be evaluated how much this participant feels as an 'owner' of the prototype. When the participatory design is performed successfully, this technician should feel a form of ownership towards the prototype. Second, a technician that has no pre-knowledge about this project or the developed prototype is participating in the user evaluation. He represents the group of users that are not experienced with the prototype, which can evaluate the learnability. In contrast to the first participant, this user is not biased about the prototype since he is not known to the imaginary advantages of it.

Scenarios
Before the execution of scenarios, both participants will have an introduction on what the goal of the experiment is, after which they get an introduction on which functionalities are present in the prototype, where these functionalities can be found, and how they should use these. After the introduction, both participants get ten minutes to become comfortable with the prototype. It is expected that after ten minutes the participants feel comfortable enough to start using the prototype into user scenarios. As described earlier, it is important that this evaluation should get as close to a real use situation, and therefore should get close to the predefined user scenarios. These scenarios, based on the observations in the Apollo Vredestein factory, were titled:

1. Fast expert assistance over distance
2. Report issues clearly to colleagues
3. Report task rejection with visual information

This means that the participants would be asked to perform tasks in three different scenarios. However, since overlap is present in the scenarios 2 and 3, it is decided to combine these into one scenario for the technicians. Because the evaluation should be performed quickly, while the participants should focus on their tasks, it is chosen to form the two scenarios into an easily readable task list (see appendix B.1 and B.2). Before executing the tasks, the participants are asked to read a short introduction to the scenarios. Afterwards, both technicians are asked to follow the steps that are described on the instruction paper, including the task list. These papers were printed on A4 and designed with visual elements, so that it would be easy to understand for the technicians. This will allow them to focus on the execution of the tasks.

During the execution of the first scenario, both participants are asked to perform the tasks simultaneously. Both are using a different iPad to complete the steps. While the technicians cannot see each other, they can hear each other via an audio connection between their iPads. The second scenario is executed with the experienced technician first. When this technician completed his tasks, the other technician was asked to start with his list of tasks.

Set-up
The set-up of all elements during the first scenario of the user evaluation can be seen in figure 7.2. Both the expert technician, which is going to help his colleague, and the technician that is being

helped are working with an iPad. Only the technician performing a PIPO-task is standing in front of the machine. The other is standing behind a wall or screen, so that both cannot hear or see each other. This gives the feeling that both technicians could be at a total different place. The iPads are connected via a WiFi hotspot created by an iPhone. The iPhone is connected with 4G, in order to make the hotspot-service possible and to download PDF-files of the machine over mobile internet. Both technicians are listening to audio earplugs that are connected to their iPads. In parallel to the developed software, the FaceTime application is started and connected to each other. In this way, both technician are able to hear their colleague during the whole experiment. Only the iPad that is in front of the machine uses the video camera of the device. The helping technician receives this camera footage over the WiFi connection. During the second scenario, the technicians perform their tasks one after the other, both standing in front of the machine. Both participants use the same iPad to work with. This iPad remains connected to the iPhone, in order to download PDF-documents of the machine.

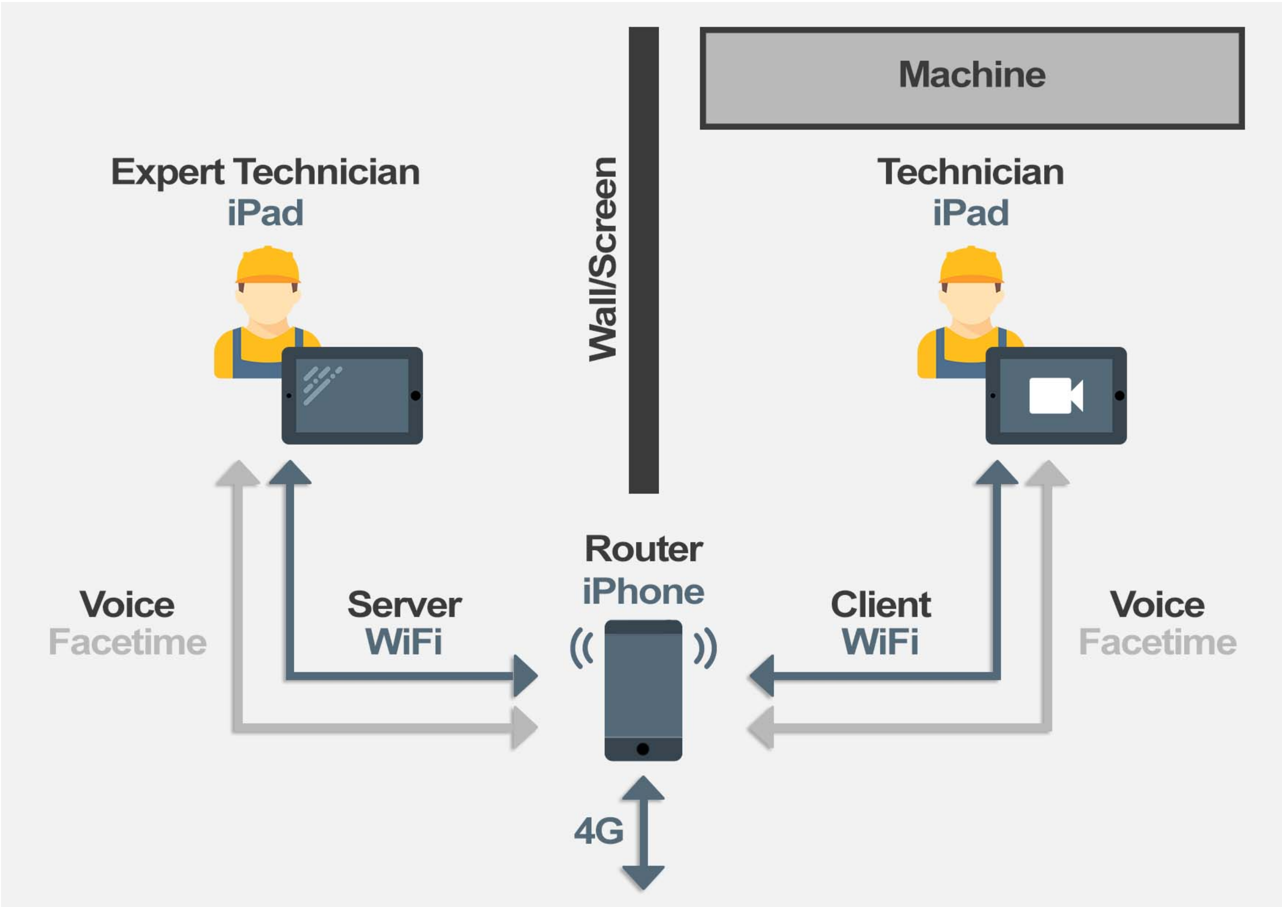


Fig. 7.2 - Set-up of the evaluation, scenario 1

Interview
In order to obtain results from the user evaluation, both technicians are participating in a semi-structured interview that is performed immediately after the execution of the two scenarios. The topics that are discussed during this interview are: participatory design, improvement of personal skills- and knowledge, improvement of quality of work, usability of the system and learnability of the system.

Time Schedule
All parts of the user evaluation are scheduled as displayed in table 7.3.

Time	Activity
00:00	Introduction of the user evaluation and explaining the functionalities in the software
00:10	Getting comfortable with the software
00:20	Setting up the devices for the execution of the scenarios
00:30	Start of experiment scenario 1
00:45	Start of experiment scenario 2
01:00	Semi-structured interview
01:30	End of evaluation

Table 7.3 - Time schedule of the user evaluation

7.2.3. Results

The results of this user evaluation are described following the topics of the semi-structured interview that is conducted with the two technicians from Apollo Vredestein. Before the execution of this interview, both technicians participated in two scenarios in which they used the prototype. The participants executed tasks following the evaluation plan, as described in 7.2.2. In the figures 7.3 and 7.4, it can be seen how the technicians performed the scenarios at the control units of the Quadruplex machine in the Apollo Vredestein factory.

Fig 7.3 - Technician using the prototype

Fig 7.4 - Expert-technician helping his colleague

Participatory design
The expert technician, which participated in design choices of the product, confirmed the feeling of contributing to the product. He tells that during the interviews he saw functionalities suggested by himself coming back into the product. Examples of such functionalities are the ‘freeze’ functionality when the expert technician is helping a colleague, or the list of machine components that show up when a target is scanned. He also tells that not all suggested functionalities are within the product, and that he understands that some suggested functionalities were technically too ambitious for an early prototype or did not have enough priority. Examples of these can be found in table 7.3. Besides his own confirmation of the feeling of contributing to participatory design, the feeling of ownership of the product is seen during the introduction phase of the user evaluation. The expert technician started explaining the functionalities to his colleague technician.

Improving personal skills- and knowledge
The technicians see many advantages to the ability of having documentation available on a tablet at the machine. Especially when the software automatically suggests suitable information when a machine area is scanned, saves the technicians time for searching the right documentation. Since the technicians only can find this information at the desktop PC's, such a solution is a 'quick win' for these workers. Additionally, the technicians expect to learn faster with such software whenever working with a machine they never worked with before, since they are expected to solve an issue faster on their own.

Improving quality of work
Software enabling the preparation of PIPO's with visual communication is expected to decrease the amount of uncertainty within the technicians' work, according the interviewed technicians. Especially in cases in which a task needs to be executed in a machine area that looks similar to others. The product helps in excluding the wrong similar machine areas, and including the right ones. The technicians tell that they think this has an effect in efficiency, but also in less mistakes and therefore quality of their work.

Both technicians think that a tool like the prototype can earn itself back easily. One technician tells that whenever a machine is stopped unexpected, this costs the company a large amount of money per hour. Especially when the machine is interrupted so long that other machines are waiting on production of this machine. Additionally, the technicians tell that it happens frequently that machines are interrupted for a period. This is a high expense for Apollo Vredestein. Whenever the augmented reality tool could cause substantially quicker repairs of defect machines, the profits generated from machines running earlier will outweigh the costs for such a tool.

Usability
Since the technicians were experienced with touch-screen interaction, they considered the prototype as quite usable. Interactions, like clicking a button for switching color, removing objects and opening documentation, or scanning a scan-target, were easily executed by the participants. However, the technician that has not seen the software before had suggestions to increase the software's usability:

1. Buttons should give immediate feedback when pressed. For example, a colored circle around the color that is selected, and the button changing color for a short amount of time when pressed.
2. When the expert help 'freezes' the screens, an icon should tell the user of the calling device that the colleague freezed the screen. Otherwise, the user would think that the application has stopped.
3. The quality of augmented reality tracking should be improved. Especially when a machine component is not near a scan-target, the software possibly makes mistakes with tracking.

Functionality
While most basic functionality was present within the prototype, according to the reactions of the technicians, they do have desires for additional functionalities. In order to increase usability of the system, the technicians suggested six functions to the product. These are given in table 7.4.

Functionality	Prio.
The possibility to enable/disable the 'drawing tool'. When this tool is enabled, the user can draw objects in the augmented reality environment. When this tool is disabled, the user can click on the augmented reality objects, to see more details.	Should
When clicked on an augmented reality message, this message should come forward on the screen with more details, so that the user can read the text clearly with all information that is available about the message.	Should
Functionality should be added to link an online PDF-document to a scan-object in the application.	Should
The user is able to change the thickness of the brush. Also when the users zooms in with the device, it is possible to draw circles that are not too large. In this way the user is able to draw circles around very small components in the machine.	Should
The software detects the distance to the machine, and automatically changes the thickness of the brush.	Could
When the user makes a 'zoom' gesture with two fingers on the touch screen, the augmented reality camera should zoom in/out.	Could

Table 7.4 - Desired functionalities by the participating technicians with their priorities

Learnability
Before the user evaluation, both technician experimented with the prototype application for ten minutes. Since the expert technician participating in the design choices already felt comfortable with the software, he started explaining his colleague technician about how to use the functionalities and when to use these. While all three scenarios were explained, both technicians tried to execute these as preparation for the real user evaluation. During these ten minutes, the technician for who the software was new was able to find and understand all features that were included in the scenarios.

During the user evaluation, the 'new' technician had the expectation that the software would include functionality that was not present. For example, the technician tried to zoom-in on the camera view by moving apart two fingers on the screen. However, this only made the application drew colored circles everywhere the technician touched the screen. Functionalities that were present in the application were already understood by the ten minutes experimenting phase before the evaluation started. The 'new' technician stated that the software was easy to understand because of the use of icons instead of plain text.

7.2.4 Conclusion

From the functional testing it can be concluded that the most important functionalities are present in the prototype. Besides an internal functional evaluation, the end-user confirms that the prototype includes the 'Must' functionalities, except for the functionality that enables the user to remove virtual objects partially. From the user evaluation it can be concluded that both participating end-users see great advantages of the use of an evolved version of the prototype in their daily work. During the experiment, it became clear that the prototype fitted within their practices. Overall, the prototype was experienced as usable and easy to understand. During the functional testing as well as during the user evaluation, functionalities were suggested that could be implemented in future versions of the product. This answers the fourth subquestion of this study.

8. Discussion

This section summarizes the report briefly, while formulating the answers to the main research question and the related subquestions. Afterwards, future work is described that continues on the performed work during this study.

8.1 Conclusions

When searching for existing concepts or products using augmented reality technology in the domain of maintenance and repair, three categories were found. First, products exist with the aim to train or educate with augmented reality objects or 3D models. These could ‘float’ in space and could virtually be taken apart. Second, products are found with the category of task instructions. This could be autonomous instructions with virtual objects showing examples of how to perform a task. Or this could be people that assist virtually over distance. Third, products that warns users for possible causes of accidents.

A stakeholder identification towards the Tip4Support product at the factory of Apollo Vredestein, resulted in a chosen target group for this study: technicians at this factory, performing maintenance and repair tasks to complex machines. As a result of a two day observation of these technicians, three PACT user stories were written that describe the actual way of working of this user group. It became clear that the way technicians are communicating, especially with the expert-technician, could be optimized. Expert-technicians spent a large amount of time on assisting colleagues, while they experience a high workload. Additionally, most important information was only available at statically placed computers.

A product that could assist technicians in communicating more efficiently, together with making the important information mobile, was expected to bring value to the technicians while their work would become more efficient. Additionally, this product concept did not involve any costs for 3D modeling, which is generally used in augmented reality products. This resulted in a feasible product concept.

During the specification phase, more detailed design choices were made with the use of the participatory design process, involving an end-user within these choices. It was decided to create an iPad application, that can recognize warning-stickers individually and can build a virtual 3D environment around it. This causes that technicians can include virtual drawings and messages that enables them to communicate visually. In order to reduce the workload of the expert-technician, a colleague can ask this technician for help via this product. Over internet both can communicate visually and with voice to assist more quickly, without the expert-technician requiring to walk through the factory. Additionally, when the software recognized a specific machine area, it provides the user of machine information that is only relevant for that area. In this way, the documentation is mobilized and automatically sorted on relevance. Both functionalities are expected to make the work of the technicians at the factory of Apollo Vredestein more efficient.

After a prototype was realized, including most of the pre-defined ‘Must’ and ‘Should’ requirements, an evaluation with this product is performed. Within this evaluation, two technicians were asked about their expectations towards a change in efficiency and effectiveness of their work when using the product. Additionally, questions are asked about the perceived value towards the prototype. One technician participated within the participatory design process, the other had no pre-knowledge of this project. Both technicians valued the prototype highly. They felt comfortable with using the software, and could imagine situations in which they would like to use the product in their work. Additionally, they expected that the product could decrease the amount of uncertainty in maintenance tasks. This could both improve effectivity and efficiency of the technicians’ work. Also, they expect that an expert-technician assisting from distance could be financially interesting. Especially when machines are defect unscheduled.

Although the opinions towards the augmented reality prototype are promising, the product should be developed further before it can be implemented in the technicians working processes. During the evaluation, a number of limitations were mentioned by the technicians. Both the situations, synchronous assistance and asynchronous communication, include these limitations. When a technician is assisted by the expert-technician with the product synchronously, it is questioned how large the perceived value is of the augmented reality in comparison with a regular video call, e.g. Skype or Google Hangouts. This could be encountered in a follow-up research. In asynchronous communication, the augmented reality tracking should increase in reliability. During the experiment, the software miscalculated the tracking a couple of times, resulting in the software highlighting another machine component than the first technician has highlighted. This could bring safety issues, when technicians are executing tasks wrongly, and could result in the technicians loosing their trust in the product.

Along the research, it became clear that the prototype is not including all functionalities that it should or could have. This includes functionality that was intentionally left out of scope of the prototype, or functionality that is suggested during the evaluation by technicians. These functionalities are given in table 7.2 and 7.3. One 'Must' requirement that is suggested by the technicians is not present in the prototype. This requirement includes an elaborated function for deleting virtual objects. Instead of deleting all objects of a machine area when pressing the delete button, it must be possible to remove a set of virtual objects that belong to each other.

Together, the four answers formulate an answer towards the main research question of this study:

> *RQ: "Can an augmented reality solution improve technicians' efficiency in executing proactive maintenance tasks to complex machines in factories?"*

It is concluded that the way of working of the technicians asks for development, especially in domains of communication and efficiency. Main cause of these needs are the growing workload, experienced by the technicians themselves, over the last years. Although it is not statistically proven that an augmented reality solution is able to improve the technician's efficiency in executing proactive maintenance tasks, the technicians say they believe it could. Since they used a realistic augmented reality prototype in two real use situations, this prototype did not require the technicians to use much of their imagination. This means that their opinion towards the prototype is believed to be close to the opinion towards a fully developed augmented reality tool. Additionally, the maintenance expert mentioned in interviews the same opinion of the technicians towards the solution. However, to be sure that a developed version of the prototype could improve technicians' efficiency, a study should be performed in which the actual way of working and working with the prototype are compared.

8.2 Future Work

Next to a comparative study between the current way of working of the technicians and the way of working of the technicians including the augmented reality product, other future work can be executed to follow up on this study. As mentioned in the conclusions, it can be studied what the effect of augmented reality is by comparing technicians using the prototype with technicians using existing ways of video communication. Also, a list of requirements not yet implemented in the product can be developed. Other suggestions for future work are described below.

In order to professionalize the prototype to a usable product, a database structure should be constructed that stores virtual objects and distributes these amongst all connected devices. Additionally, the speech functionality that is now performed by FaceTime should be integrated within the application to remain the application an efficient work process.

One of the main incentives of this project is to be able to examine whether an augmented reality tool could enhance the Tip4Support application. This product includes functionality on task handling and machine specific documentation. Interviewed technicians told they would value such a product implemented in their work. Therefore, it is expected that when both Tip4Support and the augmented reality product would be combined, a strongly valued product is constructed for technicians of Apollo Vredestein. Therefore, it should be examined whether it is possible to implement a Unity3D application within the existing framework of Tip4Support. Additionally, the combination between the prototype and Tip4Support could be interesting for technicians in other factories as well. In order to examine wether this product can be introduced at maintenance work in other domains, an observation study was performed at two car garages. This study observes car technicians in both a specialized and an all-round car garage, and is present in appendix A.2-A.4. Conclusions of this study note that communication between car technicians and specialists is very slow. Where the all-round garage sends components towards specialists for repair instead of repairing the components themselves, the specialized car garage is communicating via text with experts. It is expected that the synchronous part of the prototype can effect the efficiency and effectivity of this communication positively. An user evaluation with car technicians using the prototype could examine this hypothesis.

For feasibility and acceptation reasons it is chosen to build an augmented reality application for the iPad. Eventually, this system could be converted to applications working on other types of mobile devices. However, the technicians tell in the evaluation that they would value a scenario in which they could remain the augmented reality content while being able to work with both hands. Holding a tablet obstructs this scenario. Therefore, it should be studied how the prototype would work on an augmented reality head-mounted display. [39] describes how the Microsoft Hololens can be controlled with hand gestures. When clicking with a hand (pinching the thumb and index finger together) in front of the device, absolute and relative movement of the hand can be tracked. These movements can be used similarly as the touchscreen input currently used in the application for user interaction like selecting, scrolling and drawing. Additionally, it is interesting to know if the technicians at Apollo Vredestein would accept this experience.

Bibliography

[1] J. Carmigniani, B. Furth, M. Anisetti, P. Ceravolo, E. Damiani and M. Ivkovic, "Augmented Reality Technologies", *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 341-377, 2010.

[2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier and B. MacIntyre, "Recent Advances in Augmented Reality", *IEEE Computer Graphics and* Applications, vol. 21, no. 6, pp. 34-47, 2001.

[3] A. Tang, C. Owen, F. Biocca and W. Mou, "Comparative Effectiveness of Augmented Reality in Object Assembly", 2003.

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

[5] H. Sharp, A. Finkelstein and G. Galal, "Stakeholder Identification in the Requirements Engineering Process", 1999.

[6] H. Lasi, P. Fettke, H. Kemper, T. Feld and M. Hoffmann, "Industry 4.0", *Business & Information Systems Engineering*, vol. 6, no. 4, pp. 239-242, 2014.

[7] A. Dix, J. Finlay, G. Abowd and R. Beale, "Human-computer interaction", *Software: Practice and Experience*, vol. 28, no. 12, pp. 1357-1358, 1998.

[8] P. Milgram and F. Kishino, "A Taxonomy of Mixed Reality Virtual Displays", *IEICE Transactions of Information Systems*, vol. E77-D, no. 12, 1994.

[9] "Inglobe presents an Industrial Augmented Reality case study with Huawei", *HyperIndustry*, 2016. [Online]. Available: http://hyperindustry.inglobetechnologies.com/inglobe-presents-an-industrial-augmented-reality-case-study-with-huawei/. [Accessed: 20 May 2017].

[10] S.A. McLeod, "Observation Methods", *SimplyPsychology*, 2015. [Online]. Available: https://www.simplypsychology.org/observation.html. [Accessed: 20 May 2017].

[11] M.V. Angrosino, "Naturalistic observation", *Qualitative Essentials*, 1. Walnut Creek, CA: Left Coast Press, 2007.

[12] "Field Services with Augmented Reality", *FieldBit*, 2016. [Online]. Available: https://fieldbit.net. [Accessed: 20 May 2017].

[13] "CN2 Augmented Reality", *CN2Tech*, 2016. [Online]. Available: http://www.cn2tech.com/home. [Accessed: 20 May 2017].

[14] "Japan Airlines creates yet another first-class experience, this time using Microsoft Hololens", *Microsoft Blogs*, 2016. [Online]. Available: https://blogs.microsoft.com/transform/2016/07/11/japan-airlines-creates-yet-another-first-class-experience-this-time-using-microsoft-hololens/. [Accessed: 20 May 2017].

[15] S. Henderson and S. Feiner, "Augmented Reality for Maintenance and Repair (ARMAR). *Columbia University Computer Graphics and User Interfaces Lab*, 2016. [Online]. Available: http://graphics.cs.columbia.edu/kenny/project/armar/. [Accessed: 20 May 2017].

[16] "Sensorama", *Wikipedia, the free encyclopedia*, 2016. [Online]. Available: https://en.wikipedia.org/wiki/sensorama. [Accessed: 20 May 2017].

[17] J. Lazar, J.H. Feng and H. Hochheiser, "Ethonography", *Research Methods in Human-Computer Interaction*, first ed. United Kingdom, pp. 232 - 236, 2010.

[18] K. Asai, H. Kobayashi and K. Tomotsugu, "Augmented Instructions - A Fusion of Augmented Reality and Printed Learning Materials", *IEEE*, 2005.

[19] B. Schwald and B. Laval, de, "An Augmented Reality System for Training and Assistance to Maintenance in the Industrial Context", *WSCG*, 2003.

[20] D. Benyon and C. Macaulay, “Scenarios and the HCI-SE design problem”, *Interacting with Computers*, vol. 14, no. 4, pp. 397-405, 2002.

[21] “Vuforia Object Scanner”, *Vuforia Developer Library*, 2017. [Online]. Available: https://library.vuforia.com/articles/Training/Vuforia-Object-Scanner-Users-Guide. [Accessed: 20 May 2017].

[22] “Building your Unity game to an iOS device for testing”, *Unity3D*, 2016. [Online]. Available: https://unity3d.com/learn/tutorials/topics/mobile-touch/building-your-unity-game-ios-device-testing. [Accessed: 14 April 2017].

[23] “After mixed year, mobile AR to drive $108 billion VR/AR market by 2021”, *Digi-Capital*, 2017. [Online]. Available: http://www.digi-capital.com/news/2017/01/after-mixed-year-mobile-ar-to-drive-108-billion-vrar-market-by-2021/. [Accessed: 20 May 2017].

[24] H. Vliet, van, “Software Engineering: Principles and Practice”, *Wiley*, p. 226, 2008.

[25] C. Spinuzzi, “The methodology of participatory design”, *Technician communication*, 2004.

[26] C. Bossen, R.C. Smith, A.M. Kanstrup, J. McDonnell, M. Teli and K. Bødker, “Participatory Design in an Era of Participation”, *PDC 2016*, 2016.

[27] K. Fakhroutdinov, “UML Use Case Diagrams”, *UML-Diagrams*, 2017. [Online]. Availalbe: http://www.uml-diagrams.org/use-case-diagrams.html. [Accessed: 10 April 2017].

[28] “The Network Manager”, *Unity3D*, 2017. [Online]. Available: https://unity3d.com/learn/tutorials/topics/multiplayer-networking/network-manager. [Accessed: 17 April 2017].

[29] “RPC Details”, *Unity3D*, 2017. [Online]. Available: https://docs.unity3d.com/Manual/net-RPCDetails.html. [Accessed: 18 April 2017].

[30] “Chroma Key”, *Wikipedia, the free encyclopedia*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Chroma_key. [Accessed: 23 April 2017].

[31] N. Bilton, “Broken Glass”, *The New York Times*, p. E1, 5 February 2015.

[32] M. Fitzsimmons, “Hands on: Microsoft Hololens review”, 2017. [Online]. Available: http://www.techradar.com/reviews/wearables/microsoft-hololens-1281834/review. [Accessed: 20 May 2017].

[33] “Moverio Smart Eyewear”, *Epson*, 2017. [Online]. Available: https://epson.com/moverio-augmented-reality-smart-glasses. [Accessed: 18 April 2017].

[34] “Introducting the Vuzix Blade 3000 Smart Sunglasses”, *Vuzix*, 2017. [Online]. Available: https://www.vuzix.com. [Accessed: 18 April 2017].

[35] “The Gateway to Augmented Reality”, *Daqri*, 2017. [Online]. Available: https://daqri.com/products. [Accessed: 18 April 2017].

[36] M. Gurman, “Apple’s Next Big Thing: Augmented Reality”, *Bloomberg Technology*, 2017. [Online]. Available: https://www.bloomberg.com/news/articles/2017-03-20/apple-s-next-big-thing. [Accessed: 20 May 2017].

[37] M. Isaac, “Mark Zuckerberg Sees Augmented Reality Ecosystem in Facebook”, *The New York Times*, p. B1, 19 April 2017.

[38] J. Coa, “Magic Lep Valued at $4.5 Billion With Funding From Alibaba”, *Bloomberg Technology*, 2016. [Online] Available: https://www.bloomberg.com/news/articles/2016-02-02/magic-leap-raises-793-5-million-from-alibaba-google-qualcomm. [Accessed: 20 May 2017].

[39] “Gestures”, *Microsoft Dev Center*, 2017. [Online]. Available: https://developer.microsoft.com/en-us/windows/mixed-reality/gestures. [Accessed: 25 June 2017].

Appendices

A. Observation Notes

A.1 Apollo Vredestein

Environment

1.1	While the machine is running, not all tasks can be executed. Therefore, the machine is stopped during the PIPO's.
1.2	A technician has to walk a two to four times to the warehouse daily, which is around 250 meters from the work place.
1.3	The technician walks a significant amount of time to the PC's on a day.
1.4	External companies are hired in for exceptional and expert maintenance tasks.
1.5	Machines are highly dependent on each other. For example, a machine knows at what times it can expect certain products of other machines. When a machine is stuck and not producing while other machines are waiting for the products, it can become costly for the company quickly. This follows a "Just-In-Time" management.

Tools

2.1	The technicians uses its own senses very frequently to obtain information (sight, hearing, feeling - e.g. temperature).
2.2	Measure devices, wrenches, screwdrivers, hammers, saws, tape and cable ties are tools that technicians use.
2.3	Like the car technicians, every technician owns its own tool box on wheels.
2.4	While performing a maintenance task, the technician always uses its both hands.

Technician

3.1	A technician can loose a significant amount of time on unscheduled work for other technicians, when additional help or knowledge is necessary.
3.2	Most technicians tell that overall there is a shortage to the amount of available technicians in order to complete all maintenance work in time.
3.3	A technician tells that it is a possibility that operators will perform a number of easy technician tasks in the future, in order to reduce the workload for the technicians.

Actions

4.1	PIPO's are executed on a frequent basis, varying between once every two weeks to once every month.
4.2	When ordering machine parts, the delivery of the parts can take between two and five weeks.
4.3	With decisions that involves orders with large amounts of money, the technicians communicate with their supervisor in order to make a decision.
4.4	Before and after a PIPO, one technician and one operator should fill in a checklist. Especially the end-time of the PIPO is important to the maintenance manager.
4.5	The technician mentions that it is very important to think about safety and safety-rules before and during the work.

4.6	After the PIPO, the expert-technician fills in a document on Excel in which all executed tasks are reported together with a small description or particularities. This will be send to the supervisor.

Learning

5.1	At least one specialist per machine that has deep knowledge about the workings of the machine, both electrical and mechanical, and about the history of the machine, is very important for repairing quickly.
5.2	Knowledge about machines is obtained by doing, asking colleagues and using documentation.

Information sources

6.1	SAP: since it is hard for the organization to make changes in SAP, technicians decide on their own which tasks they will perform and which not. Hereby, they give priority to disruptions of machines.
6.2	Logbook machine: since operators have to fill in the important events to the machine in many different forms, they tend to fill in no or very few information in this document.
6.3	Intranet: technicians complain about the disability of Intranet to write to SAP. It is only possible to read data from SAP.
6.4	DMS: technician suggest to include more images to visualize where parts are placed within a machine.
6.5	DMS: it is possible for the technician to adjust and add objects to this system.
6.6	DMS: very few technicians create work instructions, which documents how a certain task should be executed including pictures, drawings and reference links to other pages in DMS.
6.7	Machine: the machine can display a lot of information and data from sensors to the operators and technicians, including error messages and raw data.
6.8	SAP: one technician mentions it would be great to have a 'status' of every maintenance task, including e.g. that anyone is working on the task, or that the task should wait for any reason.
6.9	SAP: technicians complain about the loading time of SAP: it is always time consuming. Where some has to wait for a couple of minutes to load a list, others are waiting for 20 minutes. Reading as well as writing requires time.
6.10	SAP: includes mistakes in which Monday is considered as the last day of the week, resulting in wrong lists for the supervisor.
6.11	When technicians need additional information when executing a maintenance task, they ask over the phone or print out documents like schemes and work instructions from DMS.
6.12	In most cases DMS contains full information. However, it occurs that drawings, documents and parts are missing in the system.
6.13	Machine displays: the machine can generate alerts with three types (notification, warning and error). Also a history of all alerts is available on this screen.
6.14	Logging of a sensor can be used as information source to find the cause of an issue.
6.15	Additional information for an electrician during its job mainly are the electrical schemes.
6.16	Machine displays: the machine knows very specifically what happens to the machine (e.g. which emergency stop is pressed).

Technology

7.1	The technician mentions that there is a large amount of passwords for all systems, which are only saved on two of the five PC's.

7.2	An operator tells to be overloaded to the amount of different programs and screens that should be used during its work. Even for six screens the amount of programs is to large to remain usable.
7.3	Five PC's are available, and it occurs that all are occupied and the technician cannot do its work.

Communication and planning

8.1	Since PIPO's are scheduled with certain frequencies, a year-overview can be made of the total workload to the technicians per week.
8.2	When a technician encounters problems, the supervisor is asked for advise over email. This can include pictures.
8.3	On the PC's, the technicians have an extensive phonebook in addition to the contacts in their work phone.
8.4	Not all operators and technicians report their issues to the supervisors, because they have the idea that nothing will happen from this.
8.5	A technician suggests that working with more parttime technicians will make it easier to make suitable planning.

A.2 Auto ter Riet

Environment

1.1	Working hours are between 08:00 and 17:00, Monday till Friday
1.2	Not a specialized car garage (previously a focus on Daihatsu); repairing many different car brands and types
1.2.1	Technician tells that repairing multiple car brands cause that repairs will be more time consuming that those of specialists in 1 brand. Especially with unfamiliar brands or types for the technician, it takes more time to figure out where certain parts are placed in the car.
1.3	Within the workplace 4-6 technicians are working
1.4	Large machines include: break-tester (APK), 5 car bridges, machine for balancing tires
1.5	All technicians have a personal toolbox, including tongs, screwdrivers, wrenches etc. This toolbox has around 6/7 drawers and can be moved around in the workplace on wheels.

Tools

2.1	Technicians use gloves for some of the repairing tasks. One technician mentioned that it is not possible to perform all repairs with gloves.
2.2	Most tools are included in the personal toolbox, and are tools meant for detaching or attaching parts of the car from/to each other.
2.3	The technicians knew where to find their tools immediately.
2.4	Larger machines include a machine testing a car's breaks, a machine to balance tires, bridges and a OBD-system (see observation 7.1).

Technician

3.1	According to the technician, over the last two decades less technicians are working at the garage.
3.2	A technician tells that it is difficult to find new good technicians, because car technician is not attractive work because of the low payment and the high amount of physical work combined with the large amount of thinking while completing a repairing task.

3.3	A technician tells that technicians feel lots of pressure to work quickly, coming from multiple sources: the customer, the management and/or a dealer.
3.4	Multiple technicians confirm that they have back complaints.

Actions

4.1	The technician confirms that most of the repairing tasks include the exchange of an old part from the car with a new part.
4.2	Complex repairing tasks are handled by a specialist, which is an external party to the garage.
4.3	When the technician analyzed a defect within the engine, the engine will be send to a specialist/ revision garage since the repairing is too complex and will take too much time for the technician.
4.4	Before executing a repairing task, the technicians always analyse the defect first using its own senses, using test-machines or using the OBD-system (see observation 7.1).
4.5	When the technician finds the cause of a defect, the technician communicates with the reception about the price of the repair and whether to perform the repairing task immediately.
4.6	The technician or the receptionist communicates with the customer about the repair and the prices.
4.7	When a technician starts a repairing task, the technician clocks the starting time and the end time of the repair, since can influence the price of the repair.
4.7.1	The technician clocks in or out by scanning its personal barcode with the computer in the garage.

Learning

5.1	In order to be licensed for conducting APK on cars, a technician has to perform an exam every two years which includes both a theoretical and practical test.
5.1.1	The technician says he feels a lot of pressure while performing this test for the APK license, since failing possibly ends up in a decreased loan.
5.2	In the Netherlands, the MBO education for car technician varies between two and four years.
5.3	The MBO education includes weeks of supervised practice at several garages, varying from a length of ten weeks in the first years to 24 weeks in the last year.
5.4	Completing the four years during MBO education includes a “revision exam” in which the technician student has to build-up a completely disassembled engine.

Information sources

6.1	The technicians receive most of the information through its own senses: looking, feeling, hearing, smelling, and (probably in less frequency) tasting.
6.2	In order to know whether they correctly performed a repairing tasks, the technicians conducts a test drive with the repaired car.
6.2.1	During a test drive it is possible to digitally “read” the car’s sensors for additional information.
6.3	The car includes a lot of symbols, mostly telling what not to do / what the dangers are.
6.4	Different closures of liquid tanks (like engine coolant or wiper fluid) have different colours.
6.5	On parts that are replaced by a garage, stickers are placed including the garage that performed the repair of that part and including the date of the replacement.

Technology

7.1	The garage owns a Autel MaxiSys, which is an application on a tablet that can read out all sensors in a car. This technology is called “On-board diagnostics” (OBD).
7.1.1	Autel’s OBD system has possibilities for reading cars of most known car brands.
7.1.2	Since Autel’s system is cheaper than the OBD system of Bosch, the garage exchanged their Bosch system for the MaxiSys.
7.1.3	The software is able to show raw data of all sensors in the car, as well as diagnostics including what issues are found in the car.
7.1.4	A technician tells that they frequently use the OBD system, and always use this system in a case of an electrical malfunction in the car.
7.1.5	A technician tells that at garages specialized in one car brand, also the software is mostly more specialized to the cars of the brand.
7.1.6	The software is executed on a tablet, which is well protected for falling, communicating over Bluetooth with a device that can be connected through a plug to the car.
7.1.7	The technician using the MaxiSys software had no struggles using it or finding functions.

A.3 Munsterhuis Renault

Environment

8.1	Working times varies from 08:00 till 16:30 on less busy days to 08:00 till 17:30 on busy days.
8.2	Technicians have to search for themselves where the car that they will take into repair next is parked around the garage.
8.3	Originally, every technician owns its own working place, including a bridge. But since there exist two kinds of bridges in the garage, they sometimes repair a car at the working place of another technician.

Tools

9.1	Every technician owns, and is responsible for, its own toolbox on wheels.
9.2	Multiple technicians use ear protection when working on a repairing task.
9.3	Larger tools include a OBD-laptop (see observation 14.5), bridges, machines to balance tires, automatic drills and extraction hoses.

Technician

10.1	Previously there worked eighteen car technicians in the garage, which has been decreased to approximately ten technicians.
10.1.1	A possible cause for the decrease of technicians is that the quality of the cars have improved a lot over the years, resulting in less repairs and less technicians that are necessary.
10.2	Since there are less repairs on the bottom side of the cars than before, technicians nowadays mostly work beneath them. This causes back complaints for a large amount of mechancis.

Actions

11.1	In case of repairing an electric car, the technician has to perform a few time consuming tasks like fencing the car with warning pillars and disabling the engine from any voltage.

11.1.1	It is not allowed to turn on the electrical engine while repairing the car, meaning that when a technician has to test the engine, he should conduct time consuming actions to enable the voltage again.
11.2	For every operation, even when only exchanging summer tires for winter tires, the technician does a quick inspection of the condition of the car, including things like inspecting the breaks quickly.
11.3	When making a test drive, the technician uses its senses (visual and hearing) to analyse the condition of the car. One technician mentioned not to use a OBD system when test driving.
11.4	According to technicians, a time consuming task is moving a set of tires.
11.5	A technician mentioned that car technicians are attaching and detaching parts of the car 90 percent of their time.

Learning

12.1	When a new car is introduced by Renault, all technicians have to perform a three phase course: 1. learning about the car and its parts with online video material and documentation, 2. conducting an online test, which is mandatory to pass in order to get invited for the next phase, 3. travel to the importer in order to get a class in which the technicians can physically work with the new car, including a mandatory test that the technicians should pass in order to remain a 'Renault' licensed technician.
12.2	In order to maintain a 'Renault' certification, all technicians have to conduct a test every four years. This certificate is necessary to keep working at Renault Munsterhuis. According to a technician, this causes a lot of pressure for them.
12.3	A technician confirms that they perform an APK test every two years, in order to maintain their certification in conducting APK's.
12.4	A technician tells that parts in Renaults are mostly placed in a similar way for multiple types of cars. However, Renault is using different engines for multiple types.

Information sources

13.1	Renault offers a database for technicians which includes issues that possibly occur to their cars together with a way to solve these issues step-by-step. This database is online available on the computer at the reception. The technician gets acces by plugging in his personal USB-stick in the computer and by filling in a password.
13.2	Renault offers a service in which the technicians can chat/email with the importer named KIM. In this system it is possible to send text messages to the helpdesk of the importer which are available during working hours.
13.2.1	A technician mentioned that it can take days to solve the issue with the use of KIM. Especially when the helpdesk is not available after 17:00, the technicians have to wait till the next workday.
13.2.2	KIM is used for advice, but also for permission to order new parts when a car still has warranty.
13.2.3	It is possible to add pictures to a text message.
13.3	In order to get certified for repairing electrical cars, there are courses available. At Munsterhuis there is one technician certified for this.

Technology

14.1	Within a year, the garage will exchange the instruction papers of the repairs with a digital program on tablets. The technicians will then receive a personal tablet.
14.2	Registration of operations and start- and end times of repairs is performed on a touch-screen computer at the entrance of the garage. Two technicians complaint that the touch-screen is sometimes not responsive, which annoys them.

14.3	Multiple technicians confirm that technology in the garage should be very simple, it should be possible to accidentally drop these devices on the ground and it should not be a problem if the technology gets dirty.
14.4	One technician mentioned that he liked the idea of an iPad, but was already annoyed by putting on safety goggles.
14.5	Munsterhuis uses a OBD system that is made by Renaults, specialized for their cars. This software is installed on a laptop that can via a cable and plug be attached to a car.
14.5.1	One technician is says it irritates him that he has to wait a long time till the car is analyzed on defects by the OBD system.
14.5.2	The software visually shows which sensors are where in the car, including their actual data.
14.5.3	The softwares enables the technician to look into the history of the sensors' data. In addition, the system can analyze what the defect is in a car and can tell the technician how to repair the defect step-by-step.

Expert communication

15.1	The technicians frequently communicate with the importer or manufacturer for mainly two purposes: 1. specialistic questions, when a detected issue is too complex for the technician, and 2. to ask for permission to order new parts for a car when the car still has its warranty.
15.2	The communication with the importer and manufacturer goes via a static computer, that is located at the reception of the garage. Two technicians mention that they are walking a lot from the computer to the car, and back, when they are communicating with the importer/manufacturer.
15.3	Additional images can be send to the importer/manufacturer by uploading an image. These are mostly made with a mobile phone. A technician says that this is a time consuming task.
15.4	When an unusual issue is communicated with the importer/manufacturer, it is possible that one of these parties ask to send the defect parts for research or it is possible that people from these parties visit the garage to investigate the unusual issue. Renault does this for product improvement.

A.4 Conclusions of Observations at Car Garages

First observations of car technicians are performed at the car garage 'Auto ter Riet' in Enschede. During five hours, car technicians are observed as well as actively asked about their way of working in a number of non-formal interviews. Questions and observations were related towards maintenance tasks that are regularly executed by the technicians. The second observations are performed at the car garage 'Munsterhuis Renault' in Enschede. These observations were executed similarly as during the first day, and took four hours of observing. An additional category is added to the categories listed in 4.3 after these observations: expert communication. All observations addressed to situations in which technicians ask for assistance to an expert are categorized into this category. The full lists of observations are given above, in appendix A.2 and A. 3. Beneath, the results from these observations are discussed.

Auto ter Riet

From the list of observations, four conclusions are written down that are worth noticing to the researcher.

- The car technicians mostly know very good where to find specific parts in the car, and mostly use their own senses to locate these parts. They do not search for additional information in a handbook or instruction book of the car. It is plausible that this expertise is gained during the large amount of practical work on the MBO education (see observation 5.3). However, since a technician gets a higher variation in car brands and types, it can be time consuming to find a specific part when the technician is not known to the car brand or type. Additionally, most

information that is externally obtained comes from the OBD-system (see observations 6.1 and 6.2).
- When a repairing task is analyzed as too complex for the all-round technician, like a defect within the engine or a defect within the air conditioning parts, the defect parts are send to a specialist, like a revision garage (see observations 4.2 and 4.3). The reason that these parts are send to a third party is that the specialists perform the repairing tasks quicker, and therefore cheaper.
- Autel's OBD system is a frequently used technology in the garage, providing the technicians with additional information of the car. The technicians use this tool in order to analyze the defect or to verify whether a repair is executed successfully (see observations 7.1, 7.1.3 and 7.1.4). Because of the software's simplicity and the tablet's protection for falling down, the system is easy to use for the technicians (see observations 7.1.6 and 7.1.7). Additionally, it can be concluded that for this garage and the technicians that were observed, a tablet is a way to use technology in the technicians' work.
- Since the technicians' work is both physically and mentally challenging, while this work is also paid poorly, it has become difficult to find good new car technicians. One technician mentioned that a number of students stop their education even in later phases of their study, because they see this disadvantages to the job.

Munsterhuis Renault
Analyzing the observations at Munsterhuis Renault results in three conclusions.

- After visiting both an all-round and a dealer garage, it can be concluded that there is large difference between the two. Where an all-round technician has a more wide knowledge to multiple cars and types, a dealer technician is specialized, and has a deeper knowledge, into one brand. This causes that an all-round garage sends parts to a specialist quicker than a dealer garage, which can execute the repairs mostly on their own. Also, since the specialized technician has a quick and easy access to even further specialized people through KIM, they can perform more complex tasks (see observation 15.1).
- Although the communication with the importer can be very helpful for the technicians, they also think that this is something that could be improved. The technician has to walk to the computer for every text message, and is waiting for an answer from the KIM system. Also, when the technician discovers something very unusual, a team from the importer or manufacturer can visit the garage for further investigation. These are time consuming processes (see observation 15.1 - 15.4).
- Especially technicians working at a dealer have to perform a large amount of tests during the years. These tests include APK certification (once per two year), Renault certification (once per four year), additional courses for every new car type (this happens a couple of times in a year) and, not mandatory, a course for electrical cars. Studying for these tests takes time, as well as traveling to the right location to perform the tests. In addition, these tests puts a lot of pressure on the technicians, since they are mandatory to pass the tests in order to maintain their job at Munsterhuis (see observations 12.1 - 12.3).

From the conclusions drawn after the observations at the two garages, five challenges are listed that the technicians noted to be a part of their daily work. This list is given in table beneath, including what the priority of a solution would be for the technicians, interpreted by the researcher.

No.	Description	Priority
1	With new cars or car types it can be challenging to find where parts are located in the car.	Low
2	Tasks that are too specialistic for general technicians are given to the specialists, costing the garage money.	Medium
3	The job 'car technician' can seem as very challenging work, because of physical and mental load. This causes that it is hard to find good new car technicians.	High

4	Communication with specialists through an online system, like KIM, is mostly very time consuming. In addition, miscommunication occurs with this system.	Medium
5	Car technicians, especially working at a dealer garage, have to perform a lot of tests during the year, which also includes a large pressure for passing them.	High

B. Interview with Users

A.1 Discussion Results Observations

Approximately, how many time does a technician spend on the PC during a day?
It depends highly on what a technician is doing on a day. When in preparations for a PIPO, it could be several hours a day. When carrying out work for a PIPO, it could be no time at all. On average, a technician spends one hour a day behind the PC.

Approximately, how many sessions on the PC does a technician have during a day?
Like explained before, it varies a lot. There are times that a technician is working at the PC for a week, while an technician can also have ten sessions on a day.

Approximately, how many percent of a session on the PC is a technician waiting?
Issues with waiting for SAP are solved. Now the waiting lasts for two minutes. In order to prevent waiting these two minutes, the technician tells that he always opens his email during this time.

Has the organization thought about mobile information sources for the technicians?
Apollo Vredestein thought about experimenting with tablets for technicians. However, this is not carried out and is now not in the picture. Concerns to the tablets are the costs of €1500,- per person, and prices for the organizational changes that has to be made. They realized that tablets could help technicians with reactive repair work. In the future, some operators are going to work with laptops with which they can operate the machine.

Is the way that the expert technician is used with the Quadruplex machine representative for all machines in the factory?
Yes, this technician is called a ‘first technician’ of a machine. For every machine in the factory, there is one first technician, which functions as a first contact for colleague technicians about its machine.

Approximately, how many times is an expert technician asked for help during a day?
The first technician of the Quadruplex tells that, on average, he helps out other technicians eight or ten times a day.

Approximately, how many time does helping colleague normally cost? How does this affect his workload?
This varies between a two minute call, to a ten minute inspection, to helping and carrying out repair work for an hour. The effect is that his work will shift to the future, which will cause that he cannot perform all of his tasks. Results of this is that tasks like writing work documents cannot be done.

What happens if the expert technician is not available or present during an unexpected issue with the machine?
Sometimes the first technician is being called at home, which normally happens when the issue is about a new machine. When the first technician is not present, there is no other technician that takes this role. This means that when an incident with the machine happens at such a time, the technicians in reactive repair shifts mostly figure it out themselves.

What is the effect when a PIPO is not finished in time, and a machine cannot start producing? Does this affect other machines?

A PIPO has a strict timeframe, since a PIPO is seen as the technicians borrowing the machine for a specific time from the operators. Not finishing in time, can cause large amount of losses. However, for small time delays, every machine has a small buffer. When a PIPO takes longer than expected, this is being discusses with the maintenance engineer and supervisors.

Does misunderstandings exist about parts of the machines, tasks or spare parts during maintenance work? If yes, how often? What is the effect of this?

Misunderstandings exist when a description of a task or a component is not correctly done or not complete. This can happen when an operator describes an issue to vague or not complete, and is then not present during the execution of the task. Schemes are there to prevent misunderstandings. These are then printed from the PC, cut in the way that only the relevant information is left, and supported with additional information that is written down on the paper.

A.2 Discussion Augmented Reality Solution

Advantages

- Could be an interesting solution for technicians executing reactive work. This technician has the ability then to look an issue at a machine while not even standing in front of it.
- Prevents misunderstandings about which component is meant in a machine. This happens during technicians' work.
- With such a system, that cannot make any changes in the machine, hacking is less of an issue than with a remote system than can operate a machine.
- A first technician can help other technicians from home very easily.
- The largest profit will be for technicians executing reactive work. If they could identify the cause of a defect while not at the machine, the operator could already make a fix with help of the technician, or could already pick up a new part at the warehouse. When the technician then arrives at the machine, the new part is already picked up, and the technician can start working right away.
- Advantages are seen when an organization works with different locations in the world, with the same machine.
- Since machines include more electrical parts these days, the machines get complexer. This means that the need for experts will be increasingly necessary.

Disadvantages and Worries

- Starting up an application which requires tablets will be very costly. The organization will not only have to pay for the personal tablets, but also for organizational changes to supply the tablets.
- Not all information can be transferred over video or a microphone. Sometimes a technician should really feel, hear and see a defect in order identify the cause.
- The speed of the connectivity between tablets should be good enough to result in fluent video share.
- Since a significant percentage of technicians are not frequently working with technology like smartphones and tablets, software should be clear and very easy to understand.

Additional Wishes and Ideas

- An option to show all components within the machine, and to obtain specific information per component: scheme, warehouse availability and information about its working.
- The tracking of electricity warnings could be moved to the plates on which the serial number of the electrical control panel cabinet is present.
- Include DMS is the system.
- It can be hard to find the specific name of an object. This tool could help with that.
- New machines can identify an issue on micro scale, meaning it knows which component is defect. The application could show this information if available in augmented reality.
- It would be easy if the system could display who the author is of a message.

B. User Evaluation

B.1 Instructions: scenario 1 - fast expert assistance over distance

Scenario 1
Frans Warringa

Tijdens de PIPO van de Quadruplex, moet je een defect component vervangen. Omdat niet duidelijk is waar deze precies in de kast zit, wil je overleggen met de eerstelijns monteur.

1. Druk op om te overleggen met de eerstelijns monteur.
2. Leg uit naar welk component je op zoek bent in de machine. Scan het dichtsbijzijnde .
3. De eerstelijns monteur kan nu aanwijzingen maken op jouw scherm, waar het component zich bevindt.
4. Als de aanwijzingen duidelijk zijn, druk je op om het gesprek te sluiten.
5. Nu kun je het juiste component vervangen.

Scenario 1
Dennis Bakker

Gisteren heb je de PIPO van de Quadruplex voorbereid. Je word zometeen gebeld door een monteur die tijdens deze PIPO een defect component wil vervangen, maar je bent zelf ergens anders in de fabriek aan het werk.

1. Wacht totdat de andere monteur jou belt. Zijn beeld verschijnt vanzelf op jouw scherm.
2. Je kunt met spraak overleggen wat zijn vraag is.
3. Als de monteur vraagt waar de component zich bevindt, klik je op om het beeld vast te zetten.
4. Klik op om blauw te selecteren.
5. Zet een cirkel om de defecte component.
6. Klik op om het beeld weer los te laten. De monteur zal nu ophangen als hij genoeg weet.

B.2 Instructions: scenario 2 and 3 - report issues clearly to colleagues and report task rejection with visual information

Scenario 2
Frans Warringa

Je bent bezig in de PIPO van de Quadruplex. Eén van je taken is het vervangen van een defect component in een elektriciteitskast van de machine.

1. Scan alle op de kasten, om de juiste kast te vinden.
2. Open de kast, en scan de totdat je extra informatie van de PIPO voorbereidende monteur ziet.
3. Klik op om documentatie van de machine te zien.
4. Je voert nu 'zogenaamd' de taak uit. Ga terug naar de applicatie.
5. Je ziet dat een ander component er ook slecht uitziet. Klik eerst op om alle meldingen te verwijderen.
6. Teken nu een rode cirkel om de slecht uitziende component.
7. Klik op om een melding te kunnen maken.
8. Tik op het scherm in de buurt van de rode cirkel, en typ op het toetsenbord "Component in slechte staat".
9. Klik op "Done", en maak een screen-shot om op te sturen naar de eerstelijns monteur.

Scenario 2
Dennis Bakker

Je bent een PIPO aan het voorbereiden van de Quadruplex. Eén van de taken is het vervangen van een defect component.

1. Scan een om de machine te herkennen.
2. Zet nu een rode cirkel om de defecte component.
3. Klik op om een melding te kunnen maken.
4. Tik in het scherm, in de buurt van de rode cirkel, om een melding te maken.
5. Typ nu op het toetsenbord "Component is defect, svp vervangen". Klik nu op "Done".
6. De melding is nu beschikbaar voor de uitvoerend monteur van morgen.

C. Unity3D C# Code

C.1 User Interface

General UI Classes

```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class SetTransparency : MonoBehaviour {
	void Start () {
		//Set object transparency to 40%
		Color color = gameObject.GetComponent<Image> ().material.color;
		color.a = 0.4f;
		gameObject.GetComponent<Renderer> ().material.color = color;
	}
}
```

Drawing Classes

```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class AccountSettings : MonoBehaviour {
	public GameObject firstNameField;
	public GameObject lastNameField;
	public GameObject settingsBackground;

	void Start () {
		firstNameField.SetActive (false);
		lastNameField.SetActive (false);
		settingsBackground.SetActive (false);
	}

	public void ActivateSettings () {
		if (firstNameField.activeSelf) {
			firstNameField.SetActive (false);
			lastNameField.SetActive (false);
			settingsBackground.SetActive (false);
		}  else {
			firstNameField.SetActive (true);
			lastNameField.SetActive (true);
			settingsBackground.SetActive (true);
		}
	}

	public string GetFirstName () {
		return firstNameField.GetComponent<InputField> ().text;
	}

	public string GetLastName () {
		return lastNameField.GetComponent<InputField> ().text;
	}
}

using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class SignIn : MonoBehaviour {
	public GameObject welcomeText;
	public GameObject emptyNameWarning;
	public GameObject userImage;
	public GameObject ipAddressField;
	public GameObject nameField;
	public GameObject expandButton;
	public GameObject warningButton;
	public GameObject redButton;
	public GameObject greenButton;
	public GameObject blueButton;
	public GameObject pinButton;
	public GameObject removeButton;
	public GameObject waitingText;
	public NetworkMenu networkMenuScript;
	private string ipAddress = "127.0.0.1";

	public void Expand () {
		if (ipAddressField.activeSelf) {
			ipAddressField.SetActive (false);
			expandButton.transform.localEulerAngles = new Vector3 (0, 0, 0);
		}  else {
			ipAddressField.SetActive (true);
```

```
                        expandButton.transform.localEulerAngles = new Vector3 (0, 0, 180);
                }
        }

        public void SignInClicked () {
                if (nameField.GetComponent<InputField> ().text == "") {
                        emptyNameWarning.SetActive (true);
                }  else {
                        waitingText.SetActive (true);
                        welcomeText.SetActive (false);
                        userImage.SetActive (false);
                        emptyNameWarning.SetActive (false);
                        ipAddressField.SetActive (false);
                        this.gameObject.SetActive (false);

                        string ipAddressFromField = ipAddressField.GetComponent<InputField> ().text;
                        if (ipAddressFromField != "") {
                                ipAddress = ipAddressFromField;
                        }
                        networkMenuScript.SetIPAddress (ipAddress);
                        networkMenuScript.InitializeServer ();
                }
        }

        public void SetButtonsActive() {
                warningButton.SetActive (true);
                redButton.SetActive (true);
                greenButton.SetActive (true);
                blueButton.SetActive (true);
                pinButton.SetActive (true);
                removeButton.SetActive (true);
                waitingText.SetActive (false);
        }

        public void SetButtonsInActive() {
                warningButton.SetActive (false);
                redButton.SetActive (false);
                greenButton.SetActive (false);
                blueButton.SetActive (false);
                pinButton.SetActive (false);
                removeButton.SetActive (false);
                waitingText.SetActive (true);
        }

        public string GetAuthorName() {
                return nameField.GetComponent<InputField> ().text;
        }
}
```

Machine Information Classes

```
using UnityEngine;
using System.Collections;
using Vuforia;

public class TargetExample : MonoBehaviour, ITrackableEventHandler {
        public GameObject targetExampleText;
        public MachineDocumentation machineDocumentation;
        private TrackableBehaviour mTrackableBehaviour;

        void Start () {
                mTrackableBehaviour = GameObject.Find("VuMark").GetComponent<TrackableBehaviour> ();
                if (mTrackableBehaviour) {
                        mTrackableBehaviour.RegisterTrackableEventHandler (this);
                }
        }

        public void OnTrackableStateChanged(TrackableBehaviour.Status previousStatus, TrackableBehaviour.Status
newStatus) {
                if (newStatus == TrackableBehaviour.Status.DETECTED || newStatus ==
TrackableBehaviour.Status.TRACKED) {
                        this.gameObject.SetActive (false);
                        targetExampleText.SetActive (false);
                        machineDocumentation.SetDocumentationAvailable ();
                }  else if (newStatus == TrackableBehaviour.Status.NOT_FOUND) {
                        this.gameObject.SetActive (true);
                        targetExampleText.SetActive (true);
                        machineDocumentation.SetDocumentationNotAvailable ();
                }
        }
}

using UnityEngine;
using System.Collections;

public class GetMachineName : MonoBehaviour {
        public VuMarkID vuMarkIDScript;
        private int previousVuMarkID = 0;
        private int newVuMarkID = 0;
```

```
	void Update () {
		newVuMarkID = vuMarkIDScript.GetVuMarkID ();
		if (newVuMarkID != previousVuMarkID) {
			if (newVuMarkID == 1) {
				this.GetComponent<TextMesh> ().text = "MACHINE 1";
			}  else if (newVuMarkID == 2) {
				this.GetComponent<TextMesh> ().text = "MACHINE 2";
			}  else if (newVuMarkID == 3) {
				this.GetComponent<TextMesh> ().text = "MACHINE 3";
			}  else if (newVuMarkID == 8) {
				this.GetComponent<TextMesh> ().text = "MACHINE 8";
			}  else {
				this.GetComponent<TextMesh> ().text = "NOT RECOGN.";

			}
		}
		previousVuMarkID = newVuMarkID;
	}

	public string GetName() {
		int machineID = vuMarkIDScript.GetVuMarkID ();
		return "MACHINE " + machineID.ToString();
	}
}

using System.IO;
using UnityEngine.UI;

public class MachineDocumentation : MonoBehaviour {
	public Material documentationNotAvailable;
	public Material documentationAvailable;
	public GetMachineName machineNameScript;
	public GameObject machineName;
	public GameObject componentsButton;
	private Image image = null;
	private bool machineTargeted;

	void Start() {
		image = gameObject.GetComponent<Image> ();
		componentsButton.SetActive (false);
	}

	public void OpenDocumentation() {
		if (machineTargeted) {
			if (machineNameScript.GetName ().Contains ("1")) {
				Application.OpenURL ("http://docdro.id/lKTz8wI");
			}  else {
				Application.OpenURL ("http://docdro.id/IJZFDuY");
			}
		}
	}

	public void SetDocumentationAvailable() {
		image.material = documentationAvailable;
		machineTargeted = true;
		SetMachineName (machineNameScript.GetName());
	}

	public void SetDocumentationNotAvailable() {
		image.material = documentationNotAvailable;
		machineTargeted = false;
		SetMachineName ("NO MACHINE FOUND");
		componentsButton.SetActive (false);
	}

	private void SetMachineName(string name) {
		machineName.GetComponent<Text> ().text = name;
		componentsButton.SetActive (true);
	}
}

using UnityEngine;
using System.Collections;

public class MachineComponents : MonoBehaviour {
	public GameObject componentList;
	private bool opened = false;
	private Hashtable upwards = new Hashtable();
	private Hashtable downwards = new Hashtable();

	void Start() {
		displayHeader (false);
		upwards.Add ("y", 0.555);
		upwards.Add ("time", 2);
		downwards.Add ("y", -0.555);
		downwards.Add ("time", 2);
	}

	public void toggleBar() {
```

```
			if (!opened) {
				iTween.MoveAdd (this.gameObject, upwards);
				displayHeader (true);
				Invoke ("activateComponentList", 1.5F);
				opened = true;
			}  else {
				iTween.MoveAdd (this.gameObject, downwards);
				displayHeader (false);
				deactivateComponentList ();
				opened = false;
			}
		}

		private void displayHeader(bool activate) {
			foreach (Transform child in this.transform) {
				child.gameObject.SetActive (activate);
			}
		}

		private void activateComponentList() {
			componentList.SetActive (true);
		}

		private void deactivateComponentList() {
			componentList.SetActive (false);
		}
}
```

Network Settings Classes

```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class CallSettings : MonoBehaviour {
		public GameObject ipAddressField;
		public GameObject settingsBackground;
		public GameObject frameRateField;
		public GameObject imageQualityField;

		void Start () {
			ipAddressField.SetActive (false);
			settingsBackground.SetActive (false);
			frameRateField.SetActive (false);
			imageQualityField.SetActive (false);
		}

		public void ActivateSettings () {
			if (ipAddressField.activeSelf) {
				ipAddressField.SetActive (false);
				settingsBackground.SetActive (false);
				frameRateField.SetActive (false);
				imageQualityField.SetActive (false);
			}  else {
				ipAddressField.SetActive (true);
				settingsBackground.SetActive (true);
				frameRateField.SetActive (true);
				imageQualityField.SetActive (true);
			}
		}

		public string GetIPAddress () {
			return ipAddressField.GetComponent<InputField> ().text;
		}
}
```

Network Information Classes

```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;

public class ChangeMaterial : MonoBehaviour {
		public Material openCall;
		public Material endCall;
		public GameObject settingsButton;
		public GameObject accountButton;
		private Image image = null;

		void Start() {
			image = gameObject.GetComponent<Image> ();
		}

		public void setOpenCallMaterial() {
			image.material = openCall;
			settingsButton.SetActive (true);
			accountButton.SetActive (true);
		}
```

```
	public void setEndCallMaterial() {
		image.material = endCall;
		settingsButton.SetActive (false);
		accountButton.SetActive (false);
	}
}
```

C.2 Augmented Reality Tracking

Machine number identification

```
using UnityEngine;
using System.Collections;
using Vuforia;

public class VuMarkID : MonoBehaviour {
	private VuMarkManager vuMarkManager;
	private VuMarkTarget vuMarkTarget;

	void Start () {
		vuMarkManager = TrackerManager.Instance.GetStateManager ().GetVuMarkManager ();
	}

	public int GetVuMarkID() {
		int vuMarkID = 0;
		foreach (var bhvr in vuMarkManager.GetActiveBehaviours()) {
			vuMarkTarget = bhvr.VuMarkTarget;
			vuMarkID = (int)vuMarkTarget.InstanceId.NumericValue;
		}
		return vuMarkID;
	}
}
```

Object builder (including objects from network)

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

public class BuildEnvironment : MonoBehaviour {

	public GameObject redCircle;
	public GameObject warningText;
	public GameObject warningObject;
	private GameObject selectedObject;
	private float drawingDistance;
	private float yPositionDrawing;
	private int amountWarnings;

	public GameObject parentObject;
	public Camera cam;
	public NetworkMenu networkMenuScript;
	public GUISkin warningDescriptionFieldSkin;
	private string expertName;
	private TouchScreenKeyboard keyboard;
	private string pcKeyboardText;
	private bool pcKeyboard;
	private bool instantiatePCKeyboard;
	private Scene scene;

	public VuMarkID vuMarkIDScript;
	private int previousVuMarkID = 0;
	private int vuMarkID = 0;

	void Start () {
		selectedObject = redCircle;
		amountWarnings = 0;
		scene = SceneManager.GetActiveScene ();
		pcKeyboard = false;
		instantiatePCKeyboard = false;
		pcKeyboardText = "Enter message...";
	}

	void Update () {
		if (scene.name == "AR-Assistance") {
			vuMarkID = vuMarkIDScript.GetVuMarkID ();
			if (vuMarkID != previousVuMarkID) {
				SetObjectsOfVuMarkActive (vuMarkID);
				SetObjectsOfOtherVuMarksInActive (vuMarkID);
				previousVuMarkID = vuMarkID;
			}
		}
```

```
		if (Input.GetMouseButtonDown(0)) {
			if (scene.name == "AR-Assistance") {
				drawingDistance = CalculateDistanceFromCamera ();
			}  else {
				drawingDistance = 2;
			}
		}
		if (Input.GetMouseButton(0)) {
			if (!IsPointerOverUIObject()) {
				if (selectedObject.name == "WarningObject" && Input.GetMouseButtonDown (0)) {
					amountWarnings++;
					DrawObjectAtMousePosition (selectedObject, true, false);
					DrawObjectAtMousePosition (warningText, true, true);
					keyboard = TouchScreenKeyboard.Open ("Enter message...",
TouchScreenKeyboardType.Default);
					if (SystemInfo.deviceType == DeviceType.Desktop) {
						pcKeyboard = true;
						instantiatePCKeyboard = true;
					}
				}  else if (selectedObject.name != "WarningObject") {
					DrawObjectAtMousePosition (selectedObject, false, false);
				}
			}
		}
		KeyboardTextToWarning ();
	}

	private void OnGUI() {
		GUI.skin = warningDescriptionFieldSkin;
		if (pcKeyboard && instantiatePCKeyboard) {
			GUI.SetNextControlName ("WarningDescriptionField");
			pcKeyboardText = GUI.TextField (new Rect (70, Screen.height - 80, 300, 50),
pcKeyboardText, 100);
			GUI.FocusControl ("WarningDescriptionField");
			instantiatePCKeyboard = false;
		}  else if (pcKeyboard) {
			pcKeyboardText = GUI.TextField (new Rect (70, Screen.height - 80, 300, 50),
pcKeyboardText, 100);
			if (GUI.GetNameOfFocusedControl() != "WarningDescriptionField") {
				pcKeyboard = false;
				if (Network.isServer) {
					Vector3 screenPos = cam.WorldToScreenPoint (GameObject.Find
(string.Concat (vuMarkID.ToString(), "WarningDescription", amountWarnings)).transform.position);
					networkMenuScript.SendWarningMessageToClient (screenPos.x, screenPos.y,
pcKeyboardText);
				}
				pcKeyboardText = "Enter description here...";
			}
		}
	}

	public void SetSelectedObject(GameObject drawObject) {
		selectedObject = drawObject;
	}

	public void DrawObjectFromRCP(GameObject drawObject, float xpos, float ypos) {
		drawingDistance = CalculateDistanceFromCamera ();
		Vector3 p = cam.ScreenToWorldPoint (new Vector3 (xpos, ypos, drawingDistance));
		GameObject tempObj = Instantiate (drawObject, p, Quaternion.Euler (-90, 0, 0)) as GameObject;
		tempObj.name = string.Concat (vuMarkID.ToString (), tempObj.name);
		tempObj.transform.SetParent(parentObject.transform, true);
	}

	public void DrawMessageFromRCP(float xpos, float ypos, string message) {
		Debug.Log ("draw message at: " + xpos + ", " + ypos + ": " + message);
		drawingDistance = CalculateDistanceFromCamera ();
		Vector3 p = cam.ScreenToWorldPoint (new Vector3 (xpos, ypos, drawingDistance));
		GameObject tempObj = Instantiate (warningObject, p, Quaternion.Euler (-90, 0, 0)) as GameObject;
		tempObj.transform.GetChild(5).GetComponent<GetAuthor> ().SetNewName (expertName, "(Expert)");
		tempObj.name = string.Concat (vuMarkID.ToString (), tempObj.name);
		tempObj.transform.SetParent(parentObject.transform, true);
		tempObj.transform.localEulerAngles = new Vector3 (180, 0, 0);

		GameObject tempTextObj = Instantiate (warningText, p, Quaternion.Euler (-90, 0, 0)) as
GameObject;
		tempTextObj.name = string.Concat (vuMarkID.ToString (), tempTextObj.name);
		tempTextObj.transform.SetParent(parentObject.transform, true);
		tempTextObj.GetComponent<TextMesh> ().text = message;

		tempTextObj.transform.localEulerAngles = new Vector3 (90, 0, 0);
		float shiftX = tempTextObj.transform.localPosition.x - 0.18F;
		float shiftZ = tempTextObj.transform.localPosition.z - 0.3F;
		tempTextObj.transform.localPosition = new Vector3 (shiftX, tempTextObj.transform.localPosition.y,
shiftZ);
	}

	private void DrawObjectAtMousePosition(GameObject drawObject, bool warning, bool text) {
		Vector3 p = cam.ScreenToWorldPoint (new Vector3 (Input.mousePosition.x, Input.mousePosition.y,
drawingDistance));
		GameObject tempObj = Instantiate (drawObject, p, Quaternion.Euler (-90, 0, 0)) as GameObject;
		tempObj.transform.SetParent(parentObject.transform, true);
```

```
                tempObj.name = string.Concat (vuMarkID.ToString (), tempObj.name);

                if (warning && text) {
                        if (scene.name == "AR-Assistance") {
                                tempObj.transform.localEulerAngles = new Vector3 (90, 0, 0);
                                float shiftX = tempObj.transform.localPosition.x - 0.18F;
                                                                //shift X-position of message to the right
                                float shiftZ = tempObj.transform.localPosition.z - 0.3F;
                                                                //shift Y-position of message up
                                tempObj.transform.localPosition = new Vector3 (shiftX,
tempObj.transform.localPosition.y, shiftZ);         //add calc. X & Y position to the message
                        }  else {
                                tempObj.transform.localEulerAngles = new Vector3 (90, 180, 0);
                                float shiftX = tempObj.transform.localPosition.x + 0.17F;
                                                                //shift X-position of message to the right
                                float shiftZ = tempObj.transform.localPosition.z + 0.35F;
                                                                //shift Y-position of message up
                                tempObj.transform.localPosition = new Vector3 (shiftX,
tempObj.transform.localPosition.y, shiftZ);
                        }
                        tempObj.name = string.Concat(vuMarkID.ToString(), "WarningDescription", amountWarnings);
                        TextMesh textObject = GameObject.Find(string.Concat(vuMarkID.ToString(),
"WarningDescription", amountWarnings)).GetComponent<TextMesh>();
                        textObject.text = string.Concat("Warning No. ", amountWarnings);
                }  else if (warning && !text) {
                        if (scene.name == "AR-Assistance") {
                                tempObj.transform.localEulerAngles = new Vector3 (180, 0, 0);
                        }  else {
                                tempObj.transform.localEulerAngles = new Vector3 (180, 180, 0);
                        }
                        tempObj.name = string.Concat (vuMarkID.ToString(), "WarningObject", amountWarnings);
                }
        }

        public int GetAmountWarnings() {
                return amountWarnings;
        }

        public bool IsKeyboardActive() {
                if (keyboard != null) {
                        return keyboard.active;
                }
                return false;
        }

        public void RemoveAllClient() {
                foreach (Transform tempObj in parentObject.transform) {
                        if (tempObj.name.StartsWith (vuMarkID.ToString ())) {
                                Destroy (tempObj.gameObject);
                        }
                }
        }

        public void RemoveAllServer() {
                GameObject imageTarget = GameObject.Find ("NetworkElement");
                int amountChildren = imageTarget.transform.childCount;
                int pcKeyboardIsOpen = 0;
                if (pcKeyboard || (keyboard != null && !keyboard.done)) {
                        pcKeyboardIsOpen = 2;
                }
                for (int i = 0; i < (amountChildren - pcKeyboardIsOpen); i++) {
                        Destroy (imageTarget.transform.GetChild (i).gameObject);
                }
                networkMenuScript.ResetAmountChildsSend ();
        }

        public void SetExpertName(string name) {
                expertName = name;
        }

        private void KeyboardTextToWarning() {
                if (keyboard != null && !keyboard.done) {
                        TextMesh textObject = GameObject.Find (string.Concat (vuMarkID.ToString (),
"WarningDescription", amountWarnings)).GetComponent<TextMesh> ();
                        textObject.text = keyboard.text;
                        if (textObject.text.Length >= 20) {
                                GameObject plane = GameObject.Find (string.Concat (vuMarkID.ToString (),
"WarningObject", amountWarnings)).transform.GetChild (1).gameObject;
                                plane.transform.localScale = new Vector3 (0.3F + ((textObject.text.Length - 19)
* 0.015F), 0.1F, 0.1F);
                                plane.transform.localPosition = new Vector3 (1.2F + ((textObject.text.Length -
19) * 0.075F), 0F, 0.2F);
                        }
                }  else if (pcKeyboard) {
                        TextMesh textObject = GameObject.Find (string.Concat (vuMarkID.ToString (),
"WarningDescription", amountWarnings)).GetComponent<TextMesh> ();
                        textObject.text = pcKeyboardText;
                        if (textObject.text.Length >= 20) {
                                GameObject plane = GameObject.Find (string.Concat (vuMarkID.ToString (),
"WarningObject", amountWarnings)).transform.GetChild (1).gameObject;
                                plane.transform.localScale = new Vector3 (0.3F + ((textObject.text.Length - 19)
```

```
* 0.015F), 0.1F, 0.1F);
                                        plane.transform.localPosition = new Vector3 (1.2F + ((textObject.text.Length -
19) * 0.075F), 0F, 0.2F);
                                }
                        }
        }

        private float CalculateDistanceFromCamera() {
                Vector3 heading = parentObject.transform.position - cam.transform.position;
                float distance = Vector3.Dot (heading, cam.transform.forward);
                return distance;
        }

        private bool IsPointerOverUIObject() {
                PointerEventData eventDataCurrentPosition = new PointerEventData (EventSystem.current);
                eventDataCurrentPosition.position = new Vector2 (Input.mousePosition.x, Input.mousePosition.y);
                List<RaycastResult> results = new List<RaycastResult> ();
                EventSystem.current.RaycastAll (eventDataCurrentPosition, results);
                return results.Count > 0;
        }

        private void SetObjectsOfVuMarkActive(int vuMarkNumber) {
                foreach (Transform tempObj in parentObject.transform) {
                        if (tempObj.name.StartsWith (vuMarkNumber.ToString ())) {
                                tempObj.gameObject.SetActive (true);
                        }
                }
        }

        private void SetObjectsOfOtherVuMarksInActive(int vuMarkNumber) {
                foreach (Transform tempObj in parentObject.transform) {
                        if (!tempObj.name.StartsWith (vuMarkNumber.ToString ()) && tempObj.tag !=
"NotMachineDependent") {
                                tempObj.gameObject.SetActive (false);
                        }
                }
        }
}
```

Classes used when instantiating a new message

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

public class GetAuthor : MonoBehaviour {
        public string firstName = "Default";
        public string lastName = "Default";
        public Scene scene;
        private bool authorNameAlreadySet = false;

        void Start () {
                scene = SceneManager.GetActiveScene ();
                if (scene.name == "AR-Assistance" && !authorNameAlreadySet) {
                        AccountSettings accountSettingsScript = GameObject.Find ("Account Settings
Button").GetComponent<AccountSettings> ();
                        firstName = accountSettingsScript.GetFirstName ();
                        lastName = accountSettingsScript.GetLastName ();
                        SetNameToTextMesh ();
                }  else if (scene.name == "AR-Assistance-Client") {
                        firstName = GameObject.Find ("NetworkElement").GetComponent<NetworkMenu>
().GetAuthorName();
                        lastName = "";
                        SetNameToTextMesh ();
                }
        }

        public void SetNewName(string newFirstName, string newLastName) {
                authorNameAlreadySet = true;
                firstName = newFirstName;
                lastName = newLastName;
                SetNameToTextMesh ();
        }

        private void SetNameToTextMesh() {
                this.GetComponent<TextMesh> ().text = firstName;
                this.transform.parent.transform.FindChild ("LastName").GetComponent<TextMesh> ().text = lastName;
                Debug.Log ("New name: " + firstName + lastName);
        }
}

using UnityEngine;
using System.Collections;
using System;

public class GetDate : MonoBehaviour {
        void Start () {
                TextMesh textObject = this.GetComponent<TextMesh> ();
```

```
                textObject.text = DateTime.Now.ToString("dd/MM/yyyy"); //Set object text to date
        }
}
using UnityEngine;
using System.Collections;
using System;

public class GetTime : MonoBehaviour {
        void Start () {
                TextMesh textObject = this.GetComponent<TextMesh> ();
                textObject.text = DateTime.Now.ToString("HH:mm"); //Set object text to time in hours and minutes
        }
}
```

C.3 Networking

```
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;
using System.Collections;
using System;
using System.IO;
using Vuforia;

public class NetworkMenu : MonoBehaviour {
        public string connectionIP = "127.0.0.1";
        public int portNumber = 7777;
        public NetworkView nView;
        public GUISkin textSkin;
        public GameObject Vumark;
        public GameObject RedCircle;
        public GameObject BlueCircle;
        public GameObject GreenCircle;
        public GameObject CallBackground;
        public GameObject CallInformation;
        public GameObject FrameRateField;
        public GameObject ImageQualityField;
        public ChangeMaterial changeMaterialExpertButtonScript;
        public BuildEnvironment buildEnvironmentScript;
        public SignIn signInScript;
        public CallSettings callSettingsScript;
        public PinScreen pinScreenScript;

        private bool connected = false;
        private bool sendImage = false;
        private int amountChildsSend = 0;
        private int amountFrames = 0;
        private Scene scene;
        private string authorName = "";

        private Texture2D backgroundTexture;
        public int textureWidth = 1440;
        public int textureHeight = 900;
        private float xResolutionFactor = 1;
        private float yResolutionFactor = 1;

        void Start() {
                nView = GetComponent<NetworkView> ();
                scene = SceneManager.GetActiveScene ();
                if (scene.name == "AR-Assistance") {
                        CallBackground.SetActive (false);
                        CallInformation.SetActive (false);
                }
        }

        void Update() {
                if (Network.isServer && connected) {
                        if (this.gameObject.transform.childCount > amountChildsSend) {
                                if (this.gameObject.transform.GetChild (amountChildsSend).tag == "RedCircle") {
                                        nView.RPC ("SetNewRedCircle", RPCMode.Others, Input.mousePosition.x,
Input.mousePosition.y);
                                }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag ==
"BlueCircle") {
                                        nView.RPC ("SetNewBlueCircle", RPCMode.Others, Input.mousePosition.x,
Input.mousePosition.y);
                                }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag ==
"GreenCircle") {
                                        nView.RPC ("SetNewGreenCircle", RPCMode.Others, Input.mousePosition.x,
Input.mousePosition.y);
                                }  else if (this.gameObject.transform.GetChild (amountChildsSend).tag ==
"MessageObject") {
                                        if (!buildEnvironmentScript.IsKeyboardActive ()) {
                                                Transform messageObj = this.gameObject.transform.GetChild
(amountChildsSend);
                                                Transform messageDescription =
this.gameObject.transform.GetChild (amountChildsSend + 1);
                                                SendWarningMessageToClient (messageObj.localPosition.x,
messageObj.localPosition.y, messageDescription.gameObject.GetComponent<TextMesh> ().text);
                                        }  else {
```

```
                                        amountChildsSend = amountChildsSend - 1;
                                    }
                                }
                                amountChildsSend++;
                        }
                }  else if (Network.isClient && connected) {
                        int amountFramesToWait = 10;
                        if (FrameRateField.GetComponent<InputField> ().text != "") {
                                amountFramesToWait = int.Parse (FrameRateField.GetComponent<InputField>
().text);
                        }
                        if (WaitedAmountFrames(amountFramesToWait)) {
                                sendImage = true;
                        }
                }
        }

        void OnPostRender() {
                if (Network.isClient && sendImage) {
                        sendImage = false;
                        Texture2D screenshotTexture = new Texture2D (Screen.width, Screen.height,
TextureFormat.RGB24, false);
                        screenshotTexture.ReadPixels (new Rect (0, 0, Screen.width, Screen.height), 0, 0);
                        screenshotTexture.Apply ();
                        int imageQuality = 4;
                        if (ImageQualityField.GetComponent<InputField> ().text != "") {
                                imageQuality = int.Parse (ImageQualityField.GetComponent<InputField> ().text);
                        }
                        byte[] screenshot = screenshotTexture.EncodeToJPG (imageQuality);
                        nView.RPC ("SynchronizeWebcamVideo", RPCMode.Others, screenshot);
                        DestroyImmediate (screenshotTexture);
                        screenshot = null;
                }
        }

        public string GetAuthorName() {
                return authorName;
        }

        public void SetIPAddress(string ipAddress) {
                connectionIP = ipAddress;
        }

        public void InitializeServer() {
                if (!Network.isServer) {
                        Network.InitializeServer (2, portNumber, true);
                        Debug.Log ("Server initialized (" + connectionIP + ", " + portNumber + ")");
                        authorName = signInScript.GetAuthorName ();
                }  else {
                        Network.Disconnect ();
                        connected = false;
                }
        }

        public void ConnectToServer() {
                if (!Network.isClient) {
                        changeMaterialExpertButtonScript.setEndCallMaterial ();
                        string ipAddressToConnect = callSettingsScript.GetIPAddress ();
                        if (ipAddressToConnect == "") {
                                ipAddressToConnect = "127.0.0.1";
                        }
                        Network.Connect (ipAddressToConnect, portNumber);
                        CallBackground.SetActive (true);
                        CallInformation.SetActive (true);
                        CallInformation.GetComponent<Text> ().text = "CALLING...";
                }  else {
                        Network.Disconnect ();
                        connected = false;
                        changeMaterialExpertButtonScript.setOpenCallMaterial ();
                        CallInformation.GetComponent<Text> ().text = "ENDING CALL...";
                }
        }

        public void OnPlayerConnected(NetworkPlayer player) {
                signInScript.SetButtonsActive ();
                nView.RPC ("SynchronizeScreenResolution", RPCMode.Others, float.Parse(Screen.width.ToString()),
float.Parse(Screen.height.ToString()));
                nView.RPC ("SetNameOfExpert", RPCMode.Others, authorName);
        }

        public void OnPlayerDisconnected(NetworkPlayer player) {
                signInScript.SetButtonsInActive ();
                resetBackgroundPlane ();
        }

        public void SendWarningMessageToClient(float xpos, float ypos, string message) {
                nView.RPC ("SetNewMessage", RPCMode.Others, xpos, ypos, message);
        }

        public void ResetAmountChildsSend () {
                amountChildsSend = 0;
```

```
	}

	private void OnConnectedToServer() {
		connected = true;
		Debug.Log ("Connected to Server (" + connectionIP + ", " + portNumber + ")");
		CallInformation.GetComponent<Text> ().text = "CONNECTED WITH EXPERT";
	}

	private void OnServerInitialized() {
		backgroundTexture = new Texture2D (textureWidth, textureHeight, TextureFormat.RGB24, false,
false);
		backgroundTexture.filterMode = FilterMode.Point;
		connected = true;
	}

	private void OnFailedToConnect() {
		changeMaterialExpertButtonScript.setOpenCallMaterial ();
		CallInformation.GetComponent<Text> ().text = "NO EXPERTS AVAILABLE";
	}

	private void OnDisconnectedFromServer() {
		connected = false;
		if (scene.name == "AR-Assistance") {
			changeMaterialExpertButtonScript.setOpenCallMaterial ();
			CallBackground.SetActive (false);
			CallInformation.SetActive (false);
		}  else {
			signInScript.SetButtonsInActive ();
		}
	}

	private bool WaitedAmountFrames(int frames) {
		if (amountFrames < (Time.frameCount - frames)) {
			amountFrames = Time.frameCount;
			return true;
		}
		return false;
	}

	private void resetBackgroundPlane() {
		Color32 resetColor = new Color32 (255, 255, 255, 0);
		Color32[] resetColorArray = backgroundTexture.GetPixels32 ();

		for (int i = 0; i < resetColorArray.Length; i++) {
			resetColorArray [i] = resetColor;
		}

		backgroundTexture.SetPixels32 (resetColorArray);
		backgroundTexture.Apply ();
	}

	[RPC]
	void SynchronizeScreenResolution(float xResolution, float yResolution) {
		xResolutionFactor = xResolution / Screen.width;
		yResolutionFactor = yResolution / Screen.height;
	}

	[RPC]
	void SynchronizeWebcamVideo(byte[] videoImage) {
		if (!pinScreenScript.isScreenPinned ()) {
			backgroundTexture.LoadImage (videoImage);
			GetComponent<Renderer> ().material.mainTexture = backgroundTexture;
			videoImage = null;
			buildEnvironmentScript.RemoveAllServer ();
		}
	}

	[RPC]
	void SetNewRedCircle(float xpos, float ypos) {
		Debug.Log ("POSITION OF NEW CIRCLE: " + (xpos/xResolutionFactor) + ", " + (ypos/
yResolutionFactor));
		buildEnvironmentScript.DrawObjectFromRCP (RedCircle, (xpos/xResolutionFactor), (ypos/
yResolutionFactor));
	}

	[RPC]
	void SetNewGreenCircle(float xpos, float ypos) {
		buildEnvironmentScript.DrawObjectFromRCP (GreenCircle, (xpos/xResolutionFactor), (ypos/
yResolutionFactor));
	}

	[RPC]
	void SetNewBlueCircle(float xpos, float ypos) {
		buildEnvironmentScript.DrawObjectFromRCP (BlueCircle, (xpos/xResolutionFactor), (ypos/
yResolutionFactor));
	}

	[RPC]
	void SetNewMessage(float xpos, float ypos, string message) {
		buildEnvironmentScript.DrawMessageFromRCP ((xpos/xResolutionFactor), (ypos/yResolutionFactor),
message);
```

```
        }

        [RPC]
        void SetNameOfExpert(string name) {
                buildEnvironmentScript.SetExpertName (name);
        }
}
```

C.4 Freeze functionality

PinScreen Class

```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;

public class PinScreen : MonoBehaviour {
        public NetworkMenu networkMenuScript;
        public Material pin;
        public Material pinned;
        private Image image = null;
        private bool screenPinned = false;

        void Start () {
                image = gameObject.GetComponent<Image> ();
        }

        public void pinPressed() {
                if (!screenPinned) {
                        image.material = pinned;
                        networkMenuScript.SynchronizeFreezeWithClient (true);
                        screenPinned = true;
                }  else {
                        image.material = pin;
                        networkMenuScript.SynchronizeFreezeWithClient (false);
                        screenPinned = false;
                }
        }

        public bool isScreenPinned() {
                return screenPinned;
        }
}
```

Addition to NetworkMenu Class

```
public void SynchronizeFreezeWithClient(bool freezeScreen) {
                nView.RPC ("SynchronizeFreeze", RPCMode.Others, freezeScreen);
        }

[RPC]
void SynchronizeFreeze(bool freezeScreen) {
        if (freezeScreen) {
                ObjectTracker objTracker = TrackerManager.Instance.GetTracker<ObjectTracker> ();
                objTracker.Stop ();
                CameraDevice.Instance.Stop ();
                CameraDevice.Instance.Deinit ();
        }  else {
                CameraDevice.Instance.Init (CameraDevice.CameraDirection.CAMERA_DEFAULT);
                CameraDevice.Instance.Start ();
                ObjectTracker objTracker = TrackerManager.Instance.GetTracker<ObjectTracker> ();
                objTracker.Start ();
        }
}
```

UNIVERSITY
OF TWENTE.