Optimizing Pallet Racking using Mixed-Reality

Final report of bachelor thesis

Submitted by Niels Killaars

In fulfilment of the requirements for the degree
Bachelor of Science
To be awarded by the
Fontys University of Applied Sciences

Venlo, June 11, 2018

Information Page

Fontys University of Applied Sciences
Postbus 141, 5900 AC Venlo

Bachelor Thesis

Name of student:	Niels Killaars
Student number:	2521806
Course:	Informatics - Software Engineering
Period:	February 2018 - July 2018
Company name:	LOGwear
Address:	Kazernestraat 15
Postcode, City:	5928 NL Venlo
Country:	The Netherlands
Company coach:	Monsieur, Geert G.J.Y.J.
Email:	`g.monsieur@fontys.nl`
University coach:	Langenhuizen, Marco M.L.J.A.
Email:	`m.langenhuizen@fontys.nl`
Examinator:	Dorssers, Thijs M.M.C.
External domain expert:	Janssen, Jeroen
Non-disclosure agreement:	No

Summary

English

This report exists to describe the process of the entire bachelor thesis including the period covered by the interim report. The goal of this report is to give an overview of the complete project and a conclusion about the work that has been done. This report is also delivered as a requirement for the degree Bachelor of Science.

This project is done for LOGwear, a research project focussing on documenting the use of Wearables in logistics. One of the project's partner companies is Kuijken Logistics Group (KLG) Europe, a logistics company that among other things provides storage solutions in their warehouses for customers. A lot of space is wasted in the pallet racks where goods are stored. This is because forklift operators have to decide and estimate themselves where to place the pallets in the rack. As a solution a Microsoft HoloLens application can help to measure the height of pallets using Three-Dimensional (3D) Spatial mapping technology. This device is a mixed-reality headset that is also able to display holograms in the view of the user, while still being able to see the physical environment. After measuring the height of a pallet, it guides the forklift operator to the most optimal location in the pallet rack by marking the physical locations using holograms. These locations must be determined by calculating which location will have the least wasted space when placing the pallet. A product backlog and planning has been created to plan and manage the activities within this project. This HoloLens application exchanges data with an already existing simulation environment, including a database. A forklift operator starts using the application by scanning a pallet barcode with a ring scanner. The operator can then move a virtual object and place it on the physical environment surface because of the 3D Spatial mapping. This object is then used as reference point to determine the height of the physical object where the virtual object is placed on. The forklift operator can then see the most optimal location in the rack based on the height of the pallet and the empty locations in the rack. This can be extended later to include for example considering the time a pallet will be stored. Field-testing the application concluded that the usefulness and optimization provided was recognized by the testers.

The proof-of-concept that has been developed will be used as a demo to show to KLG's customers and companies interested in the LOGwear project. The application demonstrates an innovative optimization for the pallet racking process in warehouses. A research executed during this project concluded that there are multiple use cases for smart glasses in storage warehouses. Think about logistical processes like order picking that require a lot of data for an operator to process.

Nederlands

Dit rapport bestaat om het gehele proces van de bachelor scriptie te beschrijven, inclusief de periode waarop het tussentijdse rapport betrekking heeft. Het doel van dit rapport is om een overzicht en conclusie te geven van het volledige project over het werk dat is gedaan. Dit rapport wordt ook afgeleverd als een vereiste voor het behalen van de Bachelor of Science.

Dit project is gedaan voor LOGwear, een onderzoeksproject gericht op het documenteren van het gebruik van Wearables in de logistiek. Een van de partnerbedrijven van het project is KLG Europe, een logistiek bedrijf dat onder andere opslagoplossingen biedt voor hun klanten in opslagmagazijnen. Er wordt veel ruimte verspild in de palletstellingen waar goederen worden opgeslagen. Dit komt omdat vorkheftruck bestuurders zelf moeten beslissen en inschatten waar ze de pallets in het rek plaatsen. Als oplossing kan een Microsoft HoloLens-applicatie helpen de hoogte van pallets te meten met behulp van 3D Spatial mapping technologie. Dit apparaat is een mixed-reality headset die ook in staat is om hologrammen weer te geven in het zicht van de gebruiker, terwijl hij toch de fysieke omgeving kan zien. Na het meten van de hoogte van een pallet leidt deze de vorkheftruckbestuurder naar de meest optimale locatie in het palletrek door de fysieke locaties te markeren met behulp van hologrammen. Deze locaties moeten worden bepaald door te berekenen welke locatie de minst verspilde ruimte zal hebben bij het plaatsen van de pallet. Er is een product backlog en planning gecreëerd om de activiteiten binnen dit project te plannen en te beheren. Deze HoloLens applicatie wisselt gegevens uit met een al bestaande simulatieomgeving, inclusief een database. Een heftruckbestuurder begint de applicatie te gebruiken door een palletbarcode te scannen met een ringscanner. De operator kan vervolgens een virtueel object verplaatsen en dit op het fysieke oppervlak plaatsen vanwege de 3D Spatial mapping. Dit object wordt vervolgens gebruikt als referentiepunt om de hoogte van het fysieke object waarop het virtuele object wordt geplaatst te bepalen. De heftruckbestuurder kan vervolgens de meest optimale locatie in de stelling bekijken op basis van de hoogte van de pallet en de lege locaties in de stelling. Deze bepaling kan later worden uitgebreid om bijvoorbeeld rekening te houden met de tijd dat een pallet wordt opgeslagen. Uit de praktijktest van de applicatie kwam naar voren dat de bruikbaarheid en optimalisatie die werd geboden door de testers werden herkend.

Het proof-of-concept dat ontwikkeld is zal gebruikt worden als een demo om te laten zien aan klanten van KLG en bedrijven die geïnteresseerd zijn in het LOGwear-project. De applicatie demonstreert een innovatieve optimalisatie voor het logistieke process waarbij pallets in stellingen worden geplaatst. Een onderzoek dat tijdens dit project werd uitgevoerd, concludeerde dat er sprake is van meerdere use cases voor smart glasses in opslagmagazijnen. Denk aan logistieke processen zoals orderpicking waarbij veel gegevens nodig zijn die door de operator moeten worden verwerkt.

Statement of Authenticity

I, the undersigned, hereby certify that I have compiled and written the attached document / piece of work and the underlying work without assistance from anyone except the specifically assigned academic supervisors and examiners. This work is solely my own, and I am solely responsible for the content, organization, and making of this document / piece of work.

I hereby acknowledge that I have read the instructions for preparation and submission of documents / pieces of work provided by my course / my academic institution, and I understand that this document / piece of work will not be accepted for evaluation or for the award of academic credits if it is determined that it has not been prepared in compliance with those instructions and this statement of authenticity.

I further certify that I did not commit plagiarism, did neither take over nor paraphrase (digital or printed, translated or original) material (e.g. ideas, data, pieces of text, figures, diagrams, tables, recordings, videos, code, ...) produced by others without correct and complete citation and correct and complete reference of the source(s). I understand that this document / piece of work and the underlying work will not be accepted for evaluation or for the award of academic credits if it is determined that it embodies plagiarism.

Name:	Niels Killaars
Student Number:	2521806
Place/Date:	Venlo, June 11, 2018

Signature:

Contents

List of Figures

List of Tables

Listings

Glossary

Air-tap With HoloLens "Elements"—or any virtual application or button—are selected using an air tap method, similar to clicking an imaginary computer mouse. Also see figure 5.6. 5, 13, 14

Bitbucket Bitbucket is a web-based version control repository hosting service owned by Atlassian, for source code and development projects that use either Mercurial (since launch) or Git revision control systems.. 9

Git Git is a version control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows.. 9

Ground-plane/floor-plane The ground-plane or floor-plane is a virtual object that is positioned and scaled to exactly overlay the floor in a room.. 5, 13–15

Personal computer Can be a desktop personal computer or notebook personal computer. xii, 28, 79

Raycast A raycast in Unity engine casts a ray from a point in the scene with coordinates Vector3(x,y,z). The ray is casted in a direction, with possibly a maxDistance as length, against all colliders in the scene.. 13, 14

Spatial mapping Spatial mapping provides a detailed representation of real-world surfaces in the environment around the HoloLens, allowing developers to create a convincing mixed reality experience. By merging the real world with the virtual world, an application can make holograms seem real. Applications can also more naturally align with user expectations by providing familiar real-world behaviors and interactions.. iii, iv

Version Control A component of software configuration management, version control, also known as revision control or source control, is the management of changes to documents, computer programs, large web sites, and other collections of information. Changes are usually identified by a number or letter code, termed the "revision number", "revision level", or simply "revision". For example, an initial set of files is "revision 1". When the first change is made, the resulting set is "revision 2", and so on. Each revision is associated with a timestamp and the person making the change. Revisions can be compared, restored, and with some types of files, merged. . xiii, 9

Version Control System Version control systems (VCS) most commonly run as stand-alone applications, but revision control is also embedded in various types of software such as word processors and spreadsheets, collaborative web docs[2] and in various content management systems, e.g., Wikipedia's page history. Revision control allows for the ability to revert a document to a previous revision, which is critical for allowing editors to track each other's edits, correct mistakes, and defend against vandalism and spamming.[**?**]. xiii

Wearable Smart electronic devices (electronic device with micro-controllers) that can be worn on the body as implants or accessories.. iii, iv, vi, viii, 1, 10, 26, 28–30, 60, 79–84

Acronyms

3D Three-Dimensional. iii, iv, 3, 5, 9, 14, 17, 18, 31

API Application Programming Interface. 2, 3, 9–11, 13, 25, 31

AR Augmented Reality. 28

CPU Central Processing Unit. viii, 23–25

CSV Comma-Separated Values. ix, 17, 23, 59

ER Entity-Relationship. viii, 10, 60

ETL Event Trace Log. 23

FPS Frames per Second. 24, 39

GPU Graphics Processing Unit. 25

GUI Graphical User Interface. 10

HTTP Hypertext Transfer Protocol. x, 11, 13, 25

ID Identifier. 5, 10, 13, 28–30, 79–81, 83

IT Information Technology. 2, 27

KLG Kuijken Logistics Group. iii, iv, vi, viii, 1–3, 5, 6, 9, 10, 13, 17–20, 23, 25, 26, 31, 35, 60

MRTK MixedRealityToolkit. 12, 13

PC Personal Computer . 28–30, 79–84, *Glossary:* Personal computer

QR code Quick Response Code. 18, 23

REST Representational State Transfer. 2, 3, 9–11, 13, 25, 31

SoC System On Chip. 23, 25, *Glossary:* SoC

UI User Interface. 13, 20, 23

UML Unified Modeling Language. 9

URL Uniform Resource Locator. 4, 15, 20

VC Version Control, also known as revision control or source control. *Glossary:* Version Control

VCS Version Control System. *Glossary:* Version Control System

VR Virtual Reality. 28

WMS Warehouse Management System. 9, 10, 13, 25, 28–30, 79–84

Chapter 1

Introduction

This report exists to describe the process of the entire bachelor thesis including the period covered by the interim report. The goal of this report is to give an overview of the complete project and a conclusion about the work that has been done. This report is also delivered as a requirement for the degree Bachelor of Science.

1.1 Situation Description

1.1.1 Companies Involved

This project has two companies that are involved. The first company, LOGwear [LOGwear, 2018], is actually a research project and researches how Wearables can be used in logistics processes, e.g. order picking, to improve the work process. To achieve this, they carry out applied research together with other companies. KLG Europe is one of these companies, the main objective is to research how Wearables can support the existing work processes. KLG Europe offers worldwide logistics services. From transport by road, water or rail to value added services in their modern warehouses. For warehousing, they support the storage, transshipment, repackaging, order picking, assembly and labelling processes. They also offer warehouse management systems, transport management systems, advanced planning & scheduling, track & trace, electronic invoices and automated client management reports. They provide a customer zone where customers can log on to the secure customer portal. This way a customer will gain insight on the status of their orders, goods and stocks. [KLG, 2018]

1.1.2 KLG Europe High Racking

As mentioned before KLG Europe provides a lot of different logistic solutions for their customers. For their storage solution they have pallet racking warehouses. These warehouses consist of high pallet racks and narrow aisles between them. Figure 1.1 shows the high racking warehouse in Venlo. The pallets are stored in the rack using a man-up very-narrow-aisle forklift truck that can't turn while in an aisle and brings the operator up with the pallet. An example of this kind of forklift truck is shown in chapter D. Every customer has one or more rows of storage in the warehouse. Once a pallet has been assigned to a specific row in the warehouse, the forklift operator decides the final location in rack within this row. The different pallet storage locations in a row only vary in height per level. The total height of a rack is almost the same for every row, some rows have more levels than others based on the difference in heights per level. First the forklift operator scans the pallet ID barcode using a handheld barcode scanner to register it in the system. Once the pallet has been placed in the rack the operator scans the location barcode in the rack to register the location of the pallet in the system.

Figure 1.1: KLG High Racking

1.2 Problem Outline

Project partner KLG Europe has a very specific problem in their pallet racking storage system. The pallets with products that have to be stored are varying widely in their total height. This means that the operator that is responsible for storing the pallet in the rack has to choose a location which is best fit in terms of height. The goods are stored in racks from which the different locations are also varying in size going higher up the rack. When the operator chooses to put the pallet in a location that is much higher than the actual height of the pallet a lot of space is wasted. When the operator makes this mistake multiple times, eventually there might be pallets that cannot be stored because all the empty locations are too small in terms of height. This could have been prevented if the smaller pallets were placed in the most optimal locations. This problem exists for only a few customers of KLG that have a lot of variation in the total height of pallets with products that need to be stored.

1.3 Assignment Description

The assignment consists firstly of analysing the problems KLG has with their storage warehouse. A proof-of-concept has to be implemented that uses the Microsoft HoloLens, a mixed-reality headset, to solve the problems KLG has with the pallet racking system in their warehouse. To solve the problems at KLG the HoloLens shall measure the height of the complete pallet. This information shall then be used to determine the best location to store the pallet. The operator shall then be navigated to the location. The data that is required for this process shall be simulated and added to the simulation database from LOGwear. The communication with the database shall go through the LOGwear Representational State Transfer (REST) Application Programming Interface (API), this API shall be modified to add the required requests. Research shall be done about using smart glasses for common logistical processes. During the project, the following competences from the domain description bachelor of Information Technology (IT) will be worked on. Design and implementation of user interaction and software, a complete new way of operators interacting with their working environment must be designed and implemented. Business processes must be analysed to make an accurate design of the application. Design of the software architecture will be done by keeping in mind the requirements of the new application and the existing architecture.

1.4 Scope

The project is limited to developing a proof-of-concept HoloLens application which supports operators during the storage process at KLG. The project will not focus on connecting the proof-of-concept with the live environment of KLG but will have simulated data. Allocation optimization does not have the focus, but the smallest fit will be. Which pallets need to be moved first is not part of choosing a location. There will not be worked on other products or services that are owned and developed by KLG. The project includes modifying the existing KLG demo environment database and REST API for the storage process. The conducted research will be limited to other logistical processes and how smart glasses can assist or be a useful addition for these processes and why. The research will not include other processes or services that are not related to logistics and not listed in the research design.

1.5 HoloLens Background Information

Microsoft HoloLens is a mixed-reality headset allowing the user to see holograms and the physical environment at the same time. These virtual objects can be interacted with in real time. The HoloLens has four environment understanding cameras and a depth camera. These sensors combined create the possibility of creating a 3D spatial mapping of the room. The device is always aware where it exactly is in this environment. This makes it possible to add virtual objects to the physical environment. Because this spatial mapping has a 1 to 1 ratio scale with the physical world, calculations can be done to determine the height of physical objects according to their virtual mapping.

1.6 Report Structure

The next chapter after the introduction describes the project approach and activities. After that, the requirements are explained followed by prototypes that have been created and the tools and technologies used during the project. Next, the design of the different components in this project is discussed. Chapter seven goes in depth about the implementation of the requirements and is followed by the quality management chapter. In chapter nine, research on using smart glasses in warehouses is described. The last chapter contains the conclusions and evaluation.

Chapter 2

Project Approach and Activities

The work in this project is managed using the Scrum framework. The requirements will change during the project and new ones might be added at any time. It is important to have frequent stakeholder meetings to determine if requirements must be added or changed. The goal was to have sprint meetings every two weeks, after a sprint is completed. The activities that are worked on in this project are listed using a product backlog. This list contains all the tasks and new features that must be completed and implemented for a successful ending of the project. The individual products are either part of a use case or are a general requirement for the project.

The activities from the product backlog are planned with a Trello board. [Trello, 2018] This board has been configured to consist of a few different categories. Firstly, the product backlog is a list of tasks that must be executed for every sprint. The current sprint shows the number and time frame of the sprint that is currently active. The "To Do" category lists the tasks from the product backlog that must be started for this sprint. "In Progress" contains all the tasks that are currently being worked on. Next, the "Waiting for customer approval" category shows the tasks that have been completed this sprint, but still need to be approved by the customer (stakeholders) before they are put in the "Done" category. The waiting for customer approval list is emptied after the sprint meeting, when it has been determined if the task is done or must be revised. Lastly, stakeholder meetings are listed and planned in the last column. The board does not only contain new features. When bugs or issues appear during the project they are added to the board as well. The board has different labels to support this system. The new features, user stories, issues and bugs are all prioritized to determine which tasks are worked on first. The Trello board can be accessed via the following link (LOGwear employees only): `https://trello.com/b/m4my520H/klg-pallet-racking`. In the appendix table A.1 and A.2 is a two part export of the Trello board from 31/05/2018, this can be used to view the board if there is no access to the board via the Uniform Resource Locator (URL).

Chapter 3

Requirements

The first meeting at KLG was mostly about noting down the problem definition at KLG's high racking warehouse. It was also a brainstorm session about how the problems could be solved. The next two weeks after this meeting were all about analysing and designing solutions for the problems defined in the first meeting. The solutions were first translated into functional requirements, describing which interactions between the system and user are required. The use cases defined in this chapter describe these interactions.

This section gives an overview of important interactions between a user and the system for specific scenarios. The complete requirements specification can be found in Appendix B, this document has been updated throughout the project.

Use Case	**Operator registers a pallet for storage**		
Code	UC-1		
Package	KLG		
File	**undefined**.tex		
Actors	Operator with forklift working at pallet racking system		
Description	Register a pallet for storage in the rack, scan the Identifier (ID) and measure the height		
Precondition(s)	A pallet with products is at starting location, operator is wearing the HoloLens with the application running		
Scenario	1. Operator scans the pallet's barcode to register the pallet ID 2. System displays the scanned pallet's ID and asks the user to select the pallet. 3. Operator selects the pallet using the Air-tap gesture. 4. System makes a 3D mapping of the environment, including the pallet and shows this to the user. 5. Operator selects the highest point of the products that are on the pallet using the Air-tap gesture. 6. System places a hologram on the highest point according to the 3D mapping, displays the height and asks user to verify this being the highest point. 7. Operator selects" 'register height' 8. System registers the height of this pallet in the database.		
Extensions	**6.1** System reports that a Ground-plane/floor-plane has not been created yet 1. System displays: Please scan the ground. 2. Operator scans the ground. 3. Use case proceeds at step 5.		
Result	The pallet's ID and height is registered in the system.		
Version	2.0	Author	Niels Killaars

Use Case	**Operator moves a registered pallet to the optimal location in the pallet rack**		
Code	UC-2		
Package	KLG		
File	`undefined.tex`		
Actors	Operator with forklift working at pallet racking system		
Description	Move a pallet to storage location according to the most optimal location		
Precondition(s)	A pallet which has been registered is at starting location, operator is wearing the HoloLens with the application running		
Scenario	1. System calculates the optimal storage location in the pallet rack based on the height of the pallet. 2. System displays arrow to indicate the direction to which the operator should move and marks the location. 3. Operator follows instructions to indicated location		
Extensions	**4.1** Operator ignores direction of arrow. 1. Operator moves pallet to different location as indicated. 2. Use case ends here.		
Exceptions	**2.1** No fitting locations are left. Use case ends here.		
Result	The pallet is moved to a location in the pallet rack for storage		
Version	2.0	Author	Niels Killaars

Use Case	**Operator registers location of pallet**		
Code	UC-3		
Package	KLG		
File	`undefined.tex`		
Actors	Operator with forklift working at pallet racking system		
Description	Register a pallet's location in the system		
Precondition(s)	A registered pallet has been moved to a location in the pallet rack, operator is wearing the HoloLens with the application running		
Scenario	1. System is waiting for operator to register the storage location of the pallet. 2. Operator scans a location barcode. 3. System displays that the location is registered to this pallet.		
Exceptions	**4.1** The scanned location is already registered to another pallet in the system. Use case ends here.		
Result	The pallet is registered in the system to a location in the pallet rack		
Version	2.0	Author	Niels Killaars

Chapter 4

Prototyping

When the first versions of the use cases were ready, prototypes were created to help the customer understand the interactions between the user and the system. The prototypes also made it easier to explain the proposed solutions to the customer. The scene in this chapter is a visual representation of the interactions between the system and the user. Figure 4.1 shows the prototype of the first use case. The figure shows an operator in a forklift truck at the starting location for pallet racking. The forklift truck in this figure should be seen as a man-up forklift truck like in chapter D. The starting location has place for pallets that need to be stored. The operator is wearing the Microsoft HoloLens. The green lines indicate the view of the operator and area that is being analysed by the HoloLens. To determine the height of the pallet the operator places a virtual selection object on top (or the top side) of the physical pallet. The application then calculates and displays the distance between this virtual selection object and the physical ground (the height of the pallet). If the operator is satisfied with the height calculation he can proceed by registering the pallet. The pallet barcode must be scanned and a new entry is made in the database. Information is saved about the pallet including the measured height.

Figure 4.1: Prototype Register Pallet for Storage

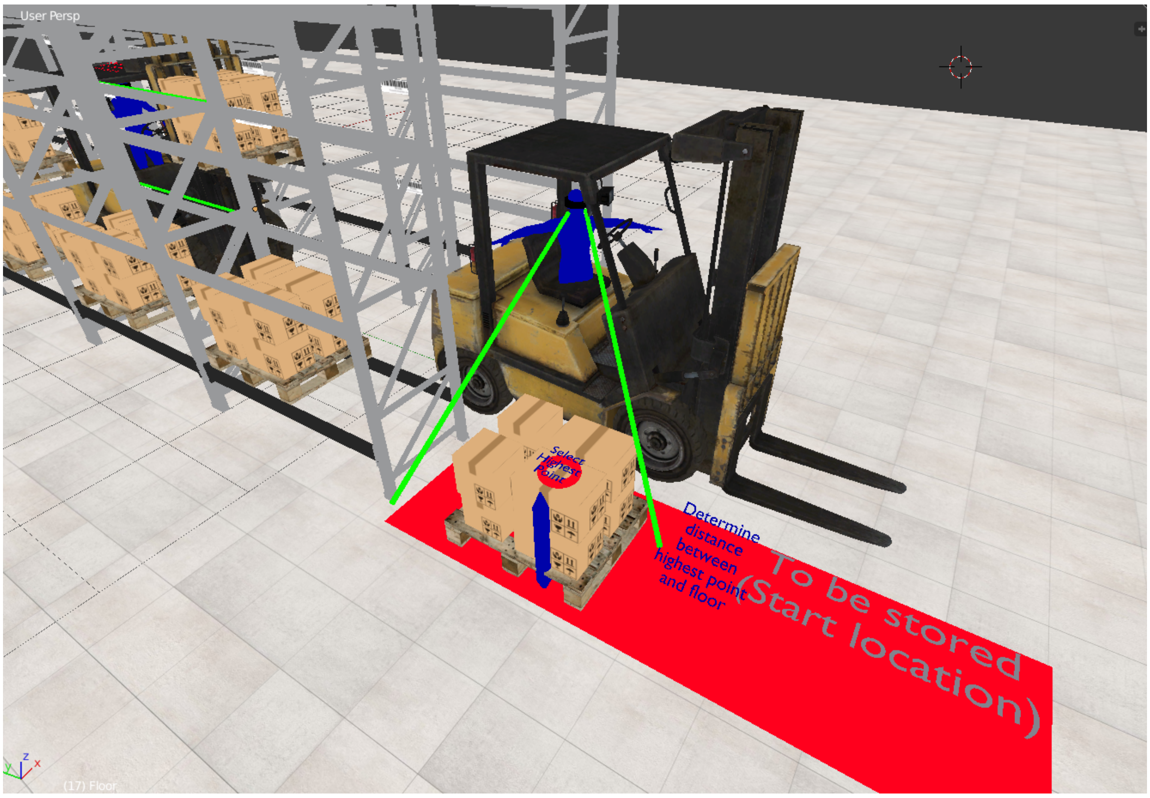
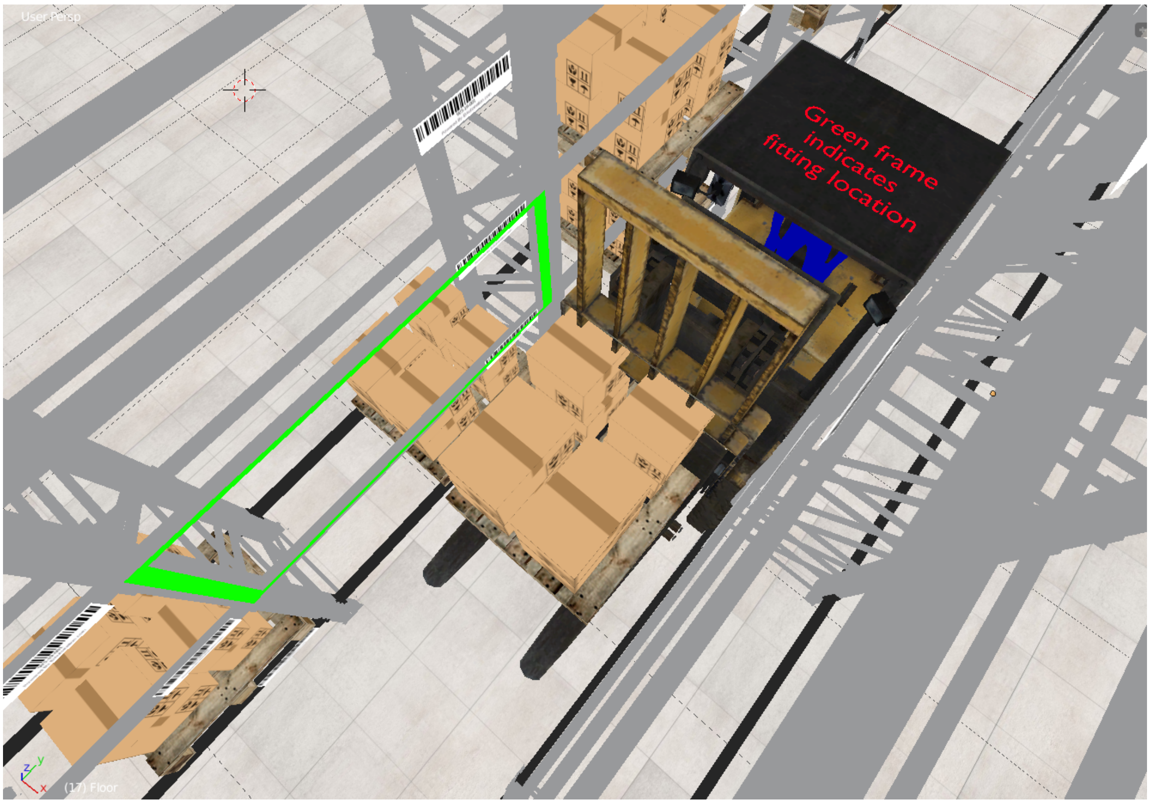


Figure 4.2 displays the prototype of use case two. An operator is transporting a height registered pallet to a location in the pallet rack. The most optimal locations are marked with a green virtual border displayed by the HoloLens. This location is most optimal because it would waste the least amount of space in terms of height.

The last prototype is of use case three. This prototype is shown in figure E.1. The operator has arrived at the assigned location and has placed the pallet in this location. The red lines indicate the scanning of a barcode to register the location in the system. After the location is scanned a confirmation must be given by the operator and the location is then saved to the pallet entry in the database.

After showing the prototypes to the customer during the second meeting a product backlog was created together with the client. The products were planned over the duration of the project. The planning is based on the priorities that were set and discussed with the customer.

Figure 4.2: Prototype Determine Optimal Location



Chapter 5

Tools and Technologies

The following tools and technologies are used during this project. The list includes a description of what it was used for.

- Microsoft HoloLens. A Mixed Reality headset to displays holograms that overlay the physical environment.
- Windows 10 laptop (at least creators update) for development with the Universal Windows Platform and Windows SDK, required for building Unity holographic applications for HoloLens.
- Blender. 3D modelling program used to create assets and prototypes.
- Unity 2017.2.1p2. The Unity engine is the most used engine for building holographic apps for HoloLens. Other engines are not directly supported. The HoloLens documentation assumes you are using Unity for development.
- Visual Studio Enterprise 2017 Version 15.5.7. Used as external script editor for the C# scripts that are created in Unity. Furthermore, Visual Studio is used for building and deploying the holographic app on the HoloLens.
- Netbeans 8.2. IDE used for work on the Java API for RESTful Web Services (JAX-RS) REST API Warehouse Management System (WMS) clone.
- Docker containers. Running the WMS and swagger documentation.
- Firebase Realtime Database. Used in the KLG demo environment to store data for, among other things, pallets and orders.
- MixedRealityToolKit-Unity. This is a collection of scripts and components intended to accelerate development of applications targeting Microsoft HoloLens and Windows Mixed Reality headsets. This collection is completely open source.
- Visual Paradigm and UMLet, used for Unified Modeling Language (UML) modelling.
- Version Control using Bitbucket and Git. Version management is important for being able to go back to a previous revision if mistakes are made. It also functions as a back-up if the files get corrupted or when the local files are unusable / unavailable.

Chapter 6

Software Architecture Design

6.1 KLG Wearable System Simulation

The existing KLG simulation for order picking and validation using Wearables has been created by the previous internship student at LOGwear Regan [2018]. This system consists of four main components. The idea is that some of these components will be altered and used during this project as well. The data needed for the pallet racking system can be stored in the already existing simulation database that has been created. To retrieve, create and update data, the existing REST API will be used. The following components are part of the KLG Wearable system simulation.

- A Firebase database, to store all order and warehouse data required for the simulation. Figure I.1 shows the ER model with all tables and required data for the database. This database will also need to store data for the KLG pallet racking system and thus needs to be altered. It will need to store data about pallets, including their ID, location in the pallet rack and height. All data needed will be simulated. The changes to the database were done at a later stage in the project, this had a low priority.
- A (simplified clone) of the WMS at KLG. This is a Java REST API application created using the JAX-RS framework. For the KLG pallet racking application this REST API needs to be altered to support creating, reading, updating and deleting data for the pallet racking system. For this project the Unity HoloLens application sends and receives data via this REST API, but it should be easy to adapt the Unity application to communicate with the live WMS at KLG. The testing of this API is done using REST-assured Haleby [2018].
- "Main system", a JavaFX Graphical User Interface (GUI) application for creating, releasing and assigning orders. There are no plans to use or change this application for this project.
- "Wearable system", an android application that can run on the Zebra WT6000 hand terminal and communicates with either a Zebra Ring-scanner or Pro Glove hand scanners. This application is obviously also not used for this project but has been demonstrated during a few workshops.

6.2 KLG Pallet Racking System

Creating a class diagram for Unity applications is hard because a lot of functionality is in separate scripts that are connected to specific game objects. The best way to describe the interaction between objects is a sequence diagram. They are used to give an overview of the objects and classes that are involved for specific scenarios.

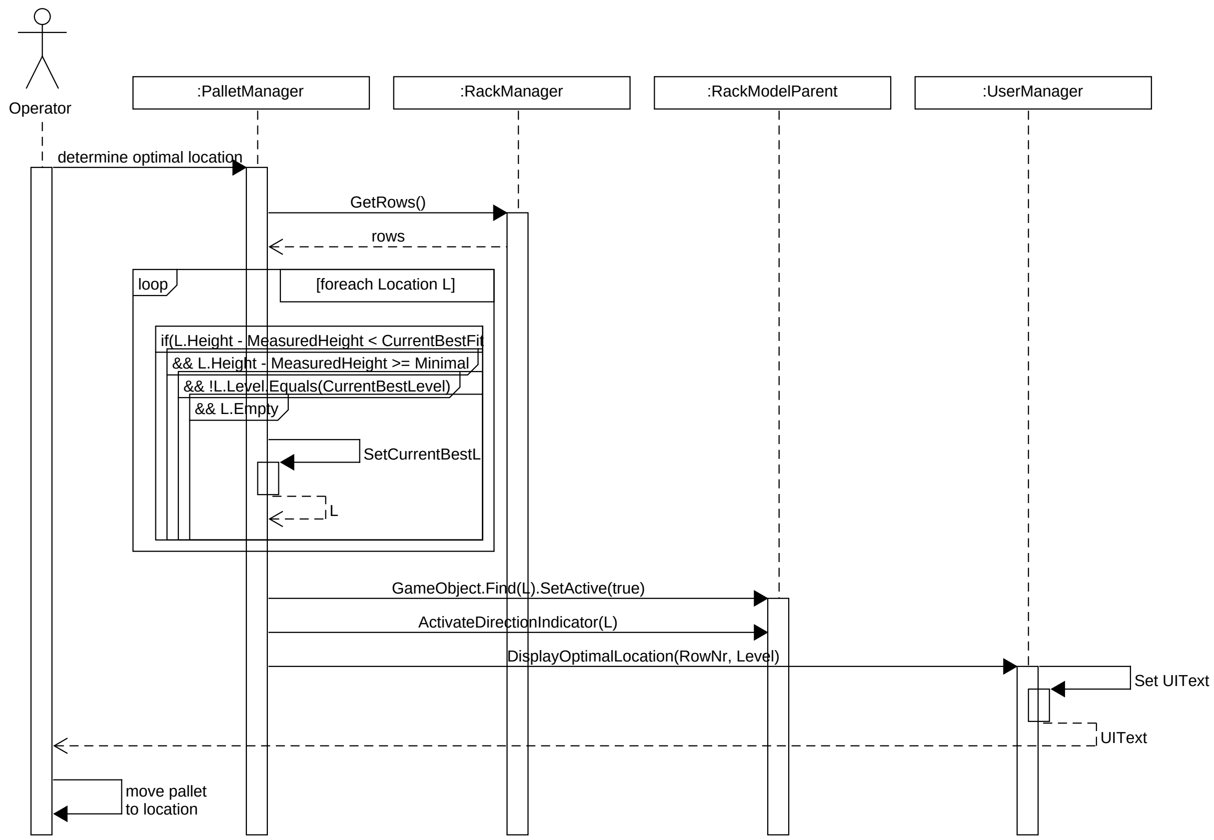
6.2.1 Register a Pallet

Figure I.2 shows a sequence diagram that describes the scripts involved in registering a pallet. The details of the implementation of this scenario is further discussed in section 7.1 of the implementation. The sequence of interaction that is required is designed using this figure.

6.2.2 Determine Optimal Location

Figure 6.1 shows the sequence diagram used to design the implementation of determining the optimal location. This scenario is further discussed in section 7.2 of the implementation. The sequence of interaction that is required is designed using this figure.

Figure 6.1: Sequence Diagram "Determine Optimal Storage Location" (UC-2)



6.2.3 Register Location

Figure I.3 shows the sequence diagram used to design the implementation of registering the location of a pallet. This scenario is further discussed in section 7.3 of the implementation. The sequence of interaction that is required is designed using this figure.

6.3 Architecture Overview

Figure I.4 shows an overview of the different components that are discussed in this chapter. The figure shows that the KLG Pallet Racking system is a Unity application, coded using C# in Visual Studio. This application is then deployed to the HoloLens which communicates with the KLG demo environment using HTTP requests. The REST API in the demo environment that receives the requests has access to a synchronized version of the data from the firebase database.

Chapter 7

Implementation

This chapter focuses on discussing the implementation of the requirements.

7.1 Register a Pallet

The third sprint included work on products from the first use case. An initial Unity project has been created which has the standard project settings required for a HoloLens application. After the application has started the operator will see a virtual spatial mapping on the physical environment. This mapping consists of a wireframe material, it represents the skeletal framework of the environment. Figure 7.1 shows an example of what the operator sees when using the application. Multiple triangles combined create a virtual layer on top of the physical world. The user can also see a small white dot in the middle of his view. This object acts as a mouse cursor and can be moved by adjusting the head rotation. It will be displayed on top of the spatial mapping in the direction the user is looking at.

Figure 7.1: Spatial Mapping

A depth camera in combination with four environment understanding cameras make the spatial mapping possible on the HoloLens. This mapping is created using the "Spatial Mapping Manager" and "Spatial Mapping Observer" scripts from the MixedRealityToolkit (MRTK) [Microsoft, 2018a]. These scripts are responsible for translating the information received from the sensors to a visual representation. The manager is configured to update the mapping every two seconds at a detail level of 500 triangles per cubic meter. These values can be changed easily to increase the update rate or level of detail in the mapping. The mapping is not created when the application starts, but the already existing mapping of the environment is used which is saved on the HoloLens. This existing mapping is updated when the application runs and after the application stops persisted in memory to be used for different applications.

For the first fifteen seconds after starting the application, the environment will be scanned. The operator is informed using User Interface (UI) text that during this time he has to wait. After this start-up the application will attempt to create a Ground-plane/floor-plane. This will be done based on the spatial mapping of the ground that was already known before starting the application, and the updates that have been made to it after the first fifteen seconds. The scripts "Spatial Processing", "Surface Meshes To Planes" and "Remove Surface Vertices" are responsible for creating the Ground-plane/floor-plane. These scripts are also provided by the MRTK. By default, the "Surface Meshes To Planes" script will generate planes for walls, ceilings, floors and tables that are recognized by the spatial mapping. It has been configured to only generate a Ground-plane/floor-plane and not render the actual plane for the user. The plane is only used as reference point to calculate the height of objects. Later during the field-tests it turned out that the maximum size of the Ground-plane/floor-plane is about $20\,\mathrm{m}^2$. This is based on the nearest spatial mapping available from the position that the application is started in the warehouse. This issue causes measurements of pallets outside this area to be invalid and had to be fixed.

To determine the height of a pallet, it needs to be registered first by scanning the pallet ID barcode. Barcode scanning is done using a ring scanner that uses blue-tooth to connect to the HoloLens. The scanner is used for this proof-of-concept only, a hand scanner is used during the normal process. The Zebra RS6000 Ring Scanner scans the barcodes and sends the input as keyboard input to the HoloLens [Zebra, 2018]. The input is processed using the "ScanInputManager" script. The function ProcessInputString() is called every frame and checks the Input.inputString variable for characters that have been received this frame. Once the first character is received ProcessInputString() is called for another second every frame. After this time the received scan input is processed by checking if it is a pallet ID or location ID scan.

As mentioned earlier, the KLG demo environment created by Courtney Regan consists among other things of a REST API and database. For this project they have been modified to support registration of pallets with their height and location. The Firebase Realtime Database part of this environment is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. In this case the REST API is the client, build using JAX-RS, a Java API for RESTful Web Services. As mentioned before the demo environment REST API functions as a clone of the WMS at KLG. The REST API had some problems getting the data in real-time though. After some research it appeared the application was using a Thread.sleep for every API request that came in before accessing the data from the database. When the Thread.sleep was removed every HTTP request coming in would return an empty result. The problem was that the application wasn't waiting for the data from the database to be available. The solution that is implemented currently uses callbacks to wait for a result to be available instead of waiting a hard-coded amount of time for every request. When for example the order list was retrieved, three other requests with sleeps where used per order in the list. The total waiting time was increasing when more orders were added to the list.

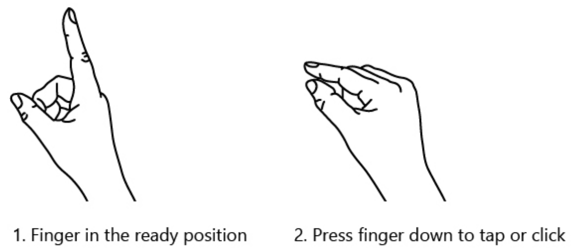
REST-assured is used to test and validate the changes to the REST API [Haleby, 2018]. A description of how these tests are executed is in chapter 8 about Quality Management.

When a pallet ID is scanned, a UnityWebRequest is used to send a HTTP POST request to the REST API. The pallet is registered with the pallet ID from the scanned barcode. When the pallet has not been registered yet in the database, a new entry is added to the list. This request is also tested using rest-assured to check for a correct response.

In figure 7.2, a cardboard box on the ground is shown with a virtual sphere on top of it. This sphere is a simple Unity GameObject and appears when a pallet has been registered. A script is controlling the behaviour of the sphere. As mentioned earlier, there is a cursor that is used to interact with the environment. When the cursor is moved on the sphere the user is able to select it by executing the Air-tap gesture (see figure 7.3). The "Tap To Place" script that controls the sphere has an OnInputClicked() method that handles a click event on the sphere. The sphere will follow the cursor when selected. Feedback from the customer was that if the sphere is out of view it is hard to find it. A custom voice command was added to fix this problem, when the user says "Reset" the sphere will be automatically picked up. This is done using the custom VoiceInputManager script and KeyWordRecognizer class from the UnityEngine.Windows.Speech library. The hit location of a Raycast in the direction that the user is looking determines the position that the sphere will be visible. This location is updated every frame. The hit location is either a hit with the spatial mapping collider or the Ground-plane/floor-plane collider. If there is no hit with a collider the sphere will appear in the direction the user is looking 2 meters away. To make the sphere appear on top of the surface the hit location is not enough. When the

Figure 7.2: Measuring a Cardboard Box

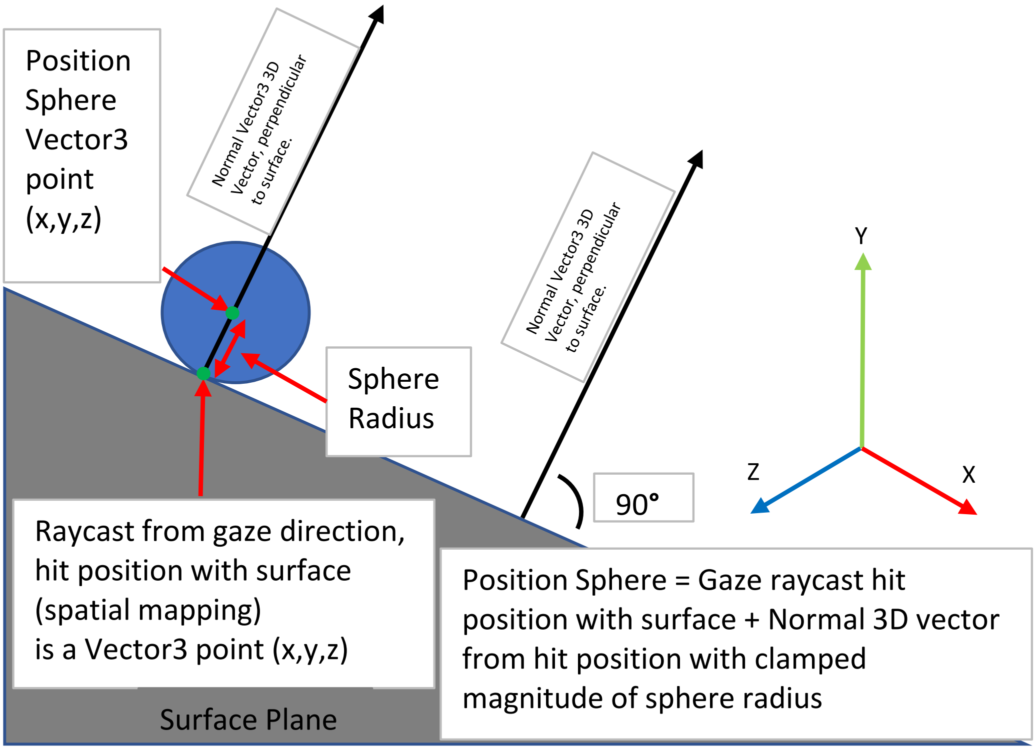
Figure 7.3: "Air-tap" Gesture



sphere is simply placed on the hit location the centre of the sphere is used (the sphere's transform position). To make the sphere appear on top of the surface firstly the normal of the hit location is needed. The normal is the vector that is perpendicular to the surface of the hit location. The final location to render the sphere is calculated by changing the normal vector's height to the radius of the sphere and then adding this vector to the hit location vector. Figure 7.4 shows how the normal is used to place the sphere exactly on the surface that the user is gazing at. See `https://docs.unity3d.com/ScriptReference/Vector3.html` for a more in depth look at how 3D Vectors and points are represented in Unity. At the start of the project a sphere was chosen as reference object. After showing this to the customer it was agreed for now to keep using a sphere as reference object. If later it is decided to use a different mesh, the code does not need to be changed. Only the sphere game object in the scene will need a different mesh filter and collider. The radius that is used for the height calculation and positioning of the object is determined the same way for spheres, cubes, planes, etc. The exact radius is determined by first retrieving the bounds from the mesh filter of the object. The bounds object is an axis aligned bounding box that fully encloses the object. This object has a property extents, which is half of the size of the bounds for in this case the X axis. Multiplying this value with the local scale of the object on the X axis gives the radius of object in meters (1 unity unit = 1 meter).

When the Air-tap gesture is executed again the object will be locked to the current position it is visible. Now a calculation is done to determine distance from that location of the sphere to the ground. This distance is the height of the physical object that the virtual object is placed on. In listing 7.1 the CalculateHeight() function is shown. To get the distance between the virtual object and Ground-plane/floor-plane a Raycast straight down from the object is needed. There can be multiple surfaces that are hit including the spatial mapping mesh, so a search is done specifically for the Ground-plane/floor-plane when analysing the list of hit locations returned from the Raycast. The hit location with the Ground-plane/floor-plane is then used in combination with the location of the object. The height is calculated by taking the distance between the current position of the object (middle of the object) and the Ground-plane/floor-plane, minus the radius of the object (middle to top). This last step is critical to get a correct calculation when the object is on top of a physical object. The accuracy of the height calculation depends on the spatial mapping and generated Ground-plane/floor-plane from this mapping. The difference between actual height and calculated height is tested in different conditions and the results are shown in table 7.1. The tests are done in a room on random objects of random heights. For every test the calculated height of the application and actual height which is measured using a measuring tape, is noted down. The absolute difference between these values is calculated and at the end the average absolute difference over all tests is calculated. As seen in table 7.1 this value is 1.98 cm. The minimal required space to place a pallet is 5 cm. The

Figure 7.4: Determine Position to Place the Object



first conclusion of the test is that the actual height is always smaller than the calculated height. There can be scenarios where a location is falsely marked as not fit because the actual height is smaller than the calculated height. This also means a location cannot be marked as optimal when it wouldn't fit, but space can be wasted when a taller location is marked because of the difference between the calculated height and actual height. The chance of this happening with an average absolute difference of 1.98 cm is quite small.

Test 7 and 15 answer the question of why there is a difference between the actual height and calculated height at all. The actual height for these two tests is zero centimetres, the object is placed on the ground. Even with the object on the ground the calculated height has a value larger than zero. This means that the Ground-plane/floor-plane has an offset from the physical ground. Later tests showed results that indicated if the spatial mapping of the ground was more accurate, the Ground-plane/floor-plane was placed more accurately and the difference between calculated height and actual height becomes smaller. Furthermore, if the HoloLens loses the spatial awareness and has to remap the surroundings, a small offset can be introduced to the position of the Ground-plane/floor-plane. Losing the spatial awareness can happen when the sensors get too close to physical objects.

In conclusion, the best height calculations are done when a correct and complete spatial mapping of the ground is completed before starting the application. Also, loss of spatial awareness can be prevented by avoiding obstructing the sensors. This in turn prevents the Ground-plane/floor-plane to be offset and ensures more accurate height calculations.

The video in appendix F (Adobe Reader and Flash Player required) shows a demonstration of measuring the height of objects. It is also possible to view the video visiting the following URL: `https://www.youtube.com/watch?v=hm5UIBH5IYI`.

Table 7.1: Height Measurement Accuracy Test

test #	Calculated Height (cm)	Actual Height (cm)	Absolute Difference (cm)
1	135.3	134.8	0.5
2	159.7	158.1	1.6
3	170.8	168	2.8
4	77.5	77	0.5
5	118.9	117.5	1.4
6	189.5	187	2.5
7	2.1	0	2.1
8	74	73	1
9	90.3	89.2	1.1
10	189.3	187.2	2.1
11	140.1	138.5	1.6
12	119.1	117.5	1.6
13	108	104.2	3.8
14	127.7	123.5	4.2
15	2.9	0	2.9
		Maximum Absolute Difference	**4.2**
		Average Absolute Difference	**1.98**

Listing 7.1: CalculateHeight() Function

```
/// <summary>
/// Raycast from object position to ground
/// Save and display the distance from object position to hit with ground
/// </summary>
private void CalculateHeight()
{
   //Array that holds all hit locations from a raycast
   RaycastHit[] hits;
   //Vector3 direction straight down
   Vector3 dir = new Vector3(0, -1, 0);
   DistanceToGround = 0f;
   //Raycast from the current position of the sphere to the ground (straight down direction)
   hits = Physics.RaycastAll(transform.position, dir, 10F);
   //For every collider that has been hit
   for (int i = 0; i < hits.Length; i++)
   {
      //Look for the ground plane
      if (hits[i].transform.name.Equals("SurfacePlane(Clone)"))
      {
         //Measure from bottom of sphere not center
         DistanceToGround = hits[i].distance - radius;
      }
   }
   //Update text to display height, in centimeters rounded with one digit
   transform.GetChild(0).GetComponent<TextMesh>().text = "Pallet height: " + "\n" +
       Math.Round(DistanceToGround * 100f, 1) + " centimeters";
}
```

7.2 Determine the Optimal Location

The high racking warehouse in Venlo provides storage solutions for customers of KLG. As mentioned before, customers buy storage locations in the rack to store their goods on pallets. The total height of the pallets can vary greatly for some customers. The average pallet height can also change which means the pallet rack needs to be changed to support the different heights. The application must determine the optimal location to store a pallet after measuring its height. It also needs to guide the operator to the location in the rack. Ultimately, this means that a dynamic 3D model of the pallet rack is required that is scaled to the physical pallet rack. This model then needs to be placed exactly overlaying the physical world to be able to indicate the optimal location and navigate to it.

7.2.1 Dynamic Pallet Rack Model

Table H.1 shows the layout of KLG's WH12 High Racking in Venlo. The table is updated when KLG makes changes to the layout of this warehouse. It shows the height in centimetres per level per row, the amount of levels per row and the total height of this row. As seen in the table and before in figure 1.1, every row has a left and right side. The aisle between the left and right side has a static width but is varying slightly per row aisle. The exact aisle width per row has been added to the WH12 layout. Furthermore, four rows have an extra pillar between them. The width is also varying per pillar and needs to be considered when creating the 3D model of the warehouse. Figure H.1 shows the width of a storage location and the space between two rows, seen from the same point of view as figure 1.1. The only data that will change is the height of the locations. When a warehouse engineer updates the heights for some rows in the warehouse, a configuration file for the application must be updated. This is a very simple comma-separated file that is basically a simplified copy of the layout table of KLG. An excel workbook template is used to make it easier to copy the data. After the new data has been copied into the template, it simply needs to be saved as CSV file on the HoloLens using the file explorer via the Windows Device Portal [Microsoft, 2018c]. Table H.2 shows the template configuration file, table H.3 is the same file after it has been saved as a CSV file.

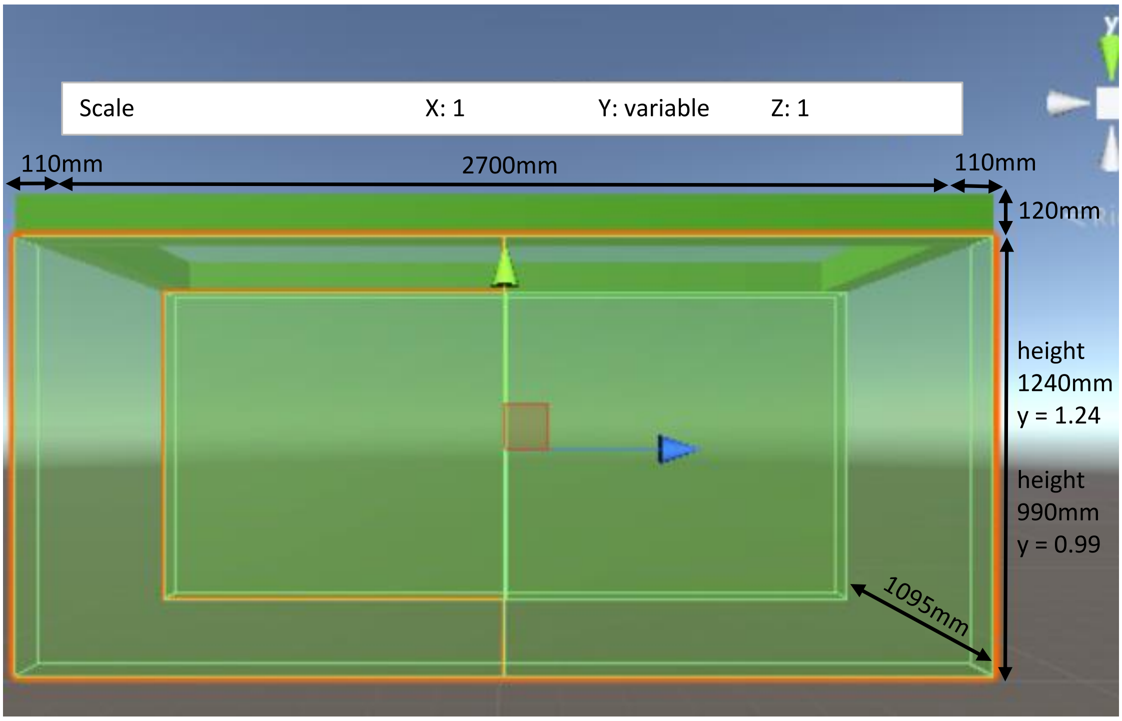
Once the application starts the RackManager script will start by reading the configuration file and processing the data. The file is located on the HoloLens in the "Windows.Storage.ApplicationData.Current.RoamingFolder.Path" folder, this is specific to where the build files are saved on the HoloLens. The System.IO.FileStream and StreamReader classes are used to create a byte stream from the file and read the characters from this stream. Using StreamReader.ReadLine() the lines in the CSV file are read one by one. Every line is then analysed by splitting the values using the comma separator and saving them in an array. The value at index zero of this array is the warehouse row name as seen in table H.2. A new warehouse row object is created and initialized, the aisle width for this row is set as a property using the value at index one. For the right side of any warehouse row, the value at index one is empty. The new warehouse row object has a property for its right side locations and left side locations. These are filled with new location objects using the values at index two until the last index as height.

The warehouse's row and location data retrieved from the configuration file is used to create a 3D model. Figure 7.5 shows a prefab of a single storage location in the rack. The width from both sides is fixed as seen in the figure. The height of the location (excluding the top bar) is variable. In figure 7.5 an empty game object is selected that is used to scale the child object, the pole of the location, towards the top bar rather than its centre. This makes it easier to scale the poles to the height of a location. The position to place the location consists of (X,Y,Z) coordinates. When creating the model think about standing in front of the pallet rack as in figure 1.1. From left to right is the X coordinate, walking through the aisle straightforward is the Z coordinate and going up the rack to different levels is the Y coordinate. The X value is determined by considering the width of one location as seen from the front side and the space between two rows. These values are static and are defined in figure H.1. If there is an aisle that must be considered, the aisle width for this row is used instead of the standard space between rows to calculate the X value. The Y value is based on the height of the location and considering the bar on top if the location is not on the first level. Lastly, the Z value is using the width when standing in front of the location in the aisle and the width of the sidebar. The Z value only changes when positioning the subsequent locations on one level.

7.2.2 Model Position and Rotation Calibration

When all the location prefabs are initialized and positioned correctly, the model looks like figure 7.6 (Screenshot taken from same point of view as figure 1.1, with different height settings). To place the virtual model exactly overlaying the

Figure 7.5: Location Prefab

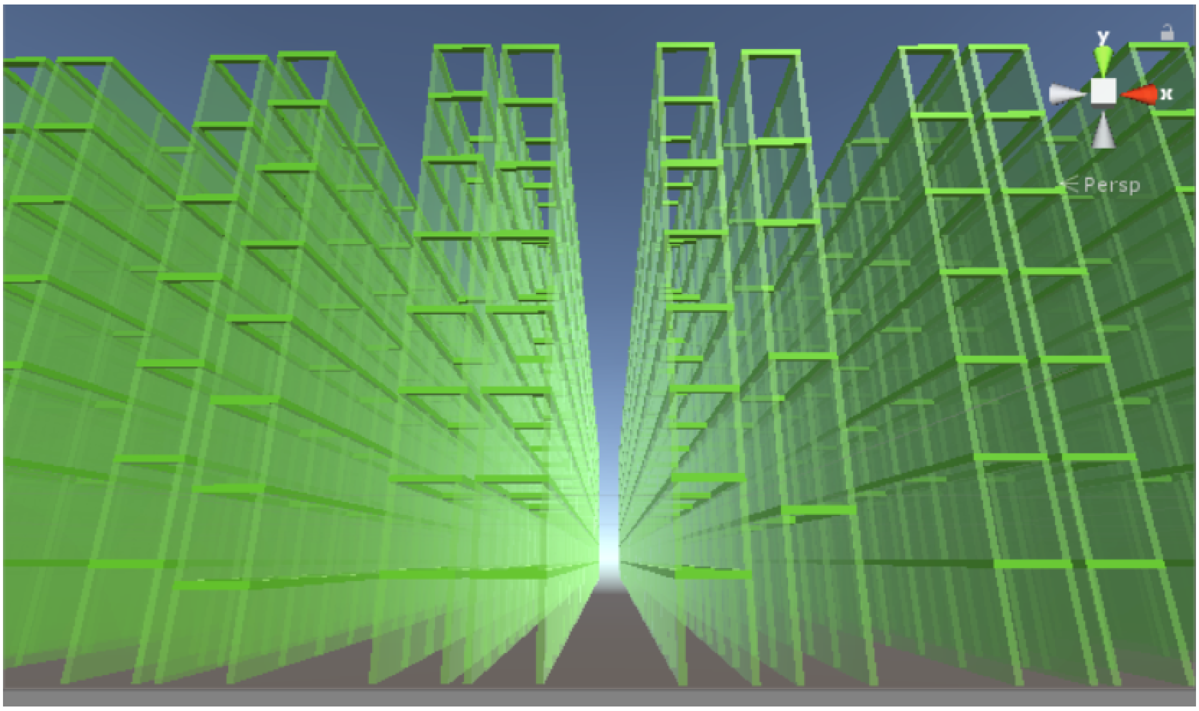


physical model, a reference point is required. The Vuforia framework [Vuforia, 2018] provides functionality to attach digital objects to specific objects. Vuforia's image targets represent images that the Vuforia SDK can detect and track. The SDK detects and tracks the features found in the image itself and compares these against a known local target resource database. In this case a Quick Response Code (QR code) will be placed on the pillar between row eight right side and nine left side. The QR code image is uploaded to the Vuforia target database on their developer portal. The target database can be downloaded as unity package file and then imported in the unity project. An image target is added to the main scene in the unity project and represents the QR code in the physical world. It must be positioned keeping in mind where it would be in the unity scene when the 3D model of the pallet rack is created. Setting the pallet rack model as a child of this image target in the scene allows Vuforia to take over control of the pallet rack model. Once the QR code is detected and tracked using the camera on the HoloLens, the image target will be positioned and rotated in the scene where it is detected in the physical world. The pallet rack model follows the position of its parent, the image target. Using extended tracking the image target will stay at the last location it was recognized when the tracking is lost. The model will always be placed horizontal, making only the changes to the Y value in rotation = Quaternion.Euler(X,Y,Z) be used to rotate the model. When the operator is not satisfied with the rotation, he gets the possibility to calibrate the rotation himself after placing the model initially. Arrows indicate the possibility to rotate the model. When the operator is satisfied with the calibration of the model compared to the physical world model, he selects a green check mark hologram to lock the model in place for this session. Always rendering the model is unnecessary and would have too much of a performance impact. The complete virtual model will thus be disabled after the calibration and single locations will be enabled when marked as optimal location.

7.2.3 Optimal Location Algorithm

When pallets arrive at KLG's high racking warehouse they are registered in the WMS and receive a barcode. After this initial process, the pallet is moved to a specific row based on the customer that owns the contents of the pallet. The HoloLens application comes in after a row has been chosen for the pallet, but still needs to know for which row

Figure 7.6: Pallet Rack Model



the optimal location must be calculated. As mentioned before the pallet rack model is locked at a specific position at this time. The spatial awareness of the HoloLens allows it to constantly have the calibration between the virtual world and physical world on point. The virtual pallet rack model is still calibrated to be exactly overlaying the physical rack. Checking which row in the virtual model is closest to the operator also gives the physical pallet rack row that is the closest. The information about which row is closest to the operator wearing the HoloLens is used to determine the optimal location for a specific row.

The forklift operator is always entering an aisle that has a left and right side with locations. The optimal location is always calculated for the left and right side of one row, the row that the operator is currently in. The optimal location for this row is currently based on a few factors. Firstly, the difference between the height of the location and the measured height of the pallet should be the smallest. Secondly, this difference should be higher than or equal to the minimal required space of 5 centimetres to be able to place the pallet. Next, the location should be empty, but there is at the of writing this report no communication with KLG's WMS to check if a location is empty, so for now all locations are marked as empty. The last check is to see if the location is not on the same side and level as the best location, this ensures the closest location to the aisle entrance is chosen on the optimal level and side. In the future this algorithm could be extended to support for example incorporating the time that the pallet will be stored. Pallets that will be stored for a longer period can be placed more to the end of the aisle and pallets that will leave the warehouse fast can be placed near the entrance of the aisle. Listing 7.2 shows the DetermineOptimalLocation() function that is called once the operator selects register height.

Listing 7.2: DetermineOptimalLocation(int RowNr) Function

```
/// <summary>
/// For one row, the left and right side, determine the optimal location.
/// Based on smallest fit, taking into account minimal required space,
/// the current best level and if the location is empty.
/// Also show a direction indicator to the optimal location.
/// </summary>
/// <param name="RowNr"></param>
private void DetermineOptimalLocation(int RowNr)
{
   Row r = rm.Rows.Find(x => x.GetRowNr() == RowNr);
   //Concat left and right of RowNr and order by level to find the best fit location closest to the floor
   var AllLocationsForThisRow = r.Left.Concat(r.Right).OrderBy(x => x.Level);
```

```
  double CurrentBestFit = 1000;
  string CurrentBestLevel = "";
  Location CurrentBestLocation = new Location("none", false, 0, "");
  foreach (Location L in AllLocationsForThisRow) {
    //Find the smallest fit empty location, keeping in mind minimal required space and the level of the location in
        the rack
    if(L.Height - MeasuredHeight < CurrentBestFit && L.Height - MeasuredHeight >= MinimalRequiredSpace &&
        !L.Level.Equals(CurrentBestLevel) && L.Empty) {
      CurrentBestFit = L.Height - MeasuredHeight;
      CurrentBestLevel = L.Level;
      CurrentBestLocation = L;
    }
  }
  //If a suitable location is found
  if (!CurrentBestLocation.ID.Equals("none")) {
    //Find the optimal location game object in the scene
    GameObject Location = GameObject.Find("RackModelParent").transform.Find(r.GetRowNr().ToString() +
        CurrentBestLevel.Last()).Find(CurrentBestLocation.ID).gameObject;
    Location.SetActive(true);
    //Activate a direction indicator to the optimal location, display optimal location info in UI
    GameObject DirectionIndicator = Location.transform.Find("DirectionIndicator(Clone)").gameObject;
    DirectionIndicator.SetActive(true);
    DirectionIndicator DirectionIndicatorInstance = DirectionIndicator.GetComponent<DirectionIndicator>();
    DirectionIndicatorInstance.OnEnable();
    um.DisplayOptimalLocation(r.GetRowNr().ToString() + CurrentBestLevel.Last(),
        CurrentBestLevel.Remove(CurrentBestLevel.Length - 1));
    sim.WaitForScanInput = true;
    OptimalLocationID = CurrentBestLocation.ID;
  } else {
    //No fitting location was found, display a warning in the UI
    um.DisplayNoFittingLocationFound(MeasuredHeight);
  }
}
```

7.3 Register Location

Once the optimal location is decided the operator can navigate to this location guided by a directional indicator. Figure 7.7 shows the directional operator pointing to the most optimal location. The operator also sees UI text that in this case indicates the row and level of the optimal location. The optimal location is marked using a green layout on the physical location. Once the operator directly looks at this location the indicator will disappear. Figure 7.8 shows a location that is marked in row 9R and level 5. The final decision in which location the pallet will be placed is made by the operator. The indicated optimal location is a guideline. Once the operator arrives at a location and places the pallet, the next step is to scan the location barcode. In the demo environment database, the location is registered to the pallet. At this point the operator is informed via the UI text that the scanned location barcode is registered to the scanned pallet barcode. The complete process is demonstrated in a short demo video filmed at KLG's high racking warehouse. View the video visiting the following URL: `https://youtu.be/6-dPYoXzBd0`.

Figure 7.7: Navigate to a Location

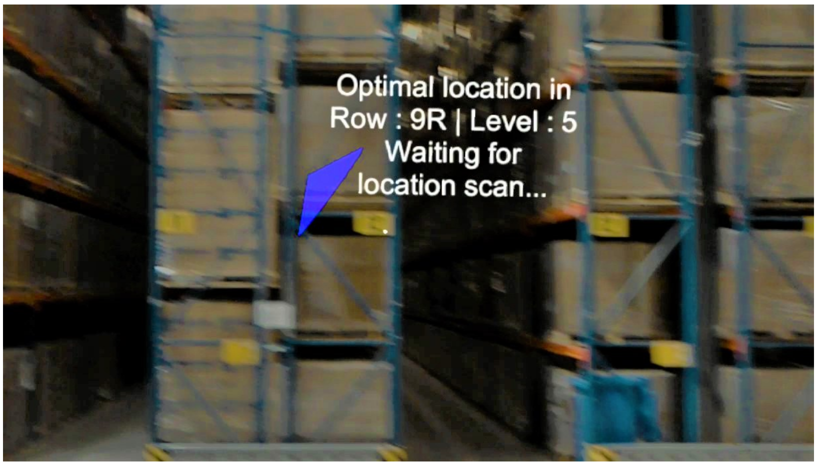
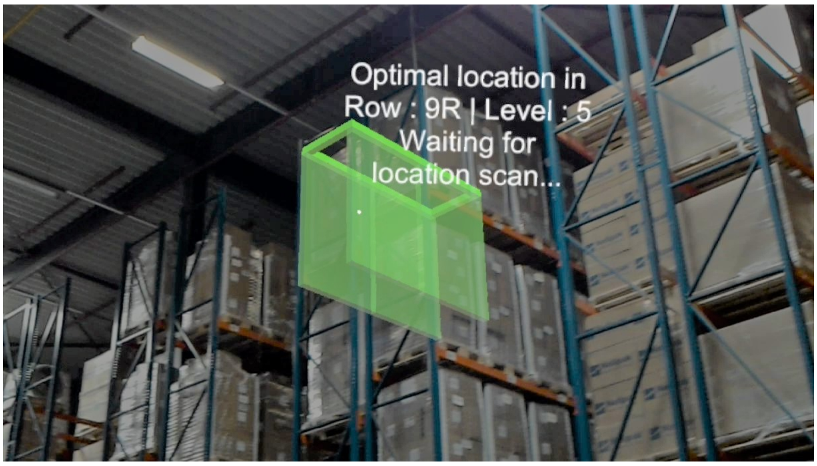


Figure 7.8: Optimal Location Indicated with Holograms



Chapter 8

Quality Management

This chapter provides details on the automated testing, user experience and usability testing and performance testing that has been done to ensure a high-quality proof-of-concept.

8.1 Functional Testing

Functional tests are described in the test document in chapter K. They describe features, the corresponding function in the application, what the test prerequisites are and the expected results and actual results.

8.2 Non-functional Testing

The non-functional tests consist of field testing, user experience testing in different conditions and performance testing.

8.2.1 Field Tests

The goal of field tests is to evaluate the adoption of product features. These tests gather data and feedback on natural, unguided use of the product over an extended period. They answer the question: Will operators use the product? In this case operators will have time to try out the application for half a day. The idea is to have a few operators try to do their normal routine when storing pallets, but in this case have access to extra guidance from the HoloLens. People with different roles have been chosen to field-test the application. Firstly, a forklift operator who is used to the normal routine where he decides himself the locations to store pallets. Next, a warehouse engineer who is responsible for the design of the pallet rack. The warehouse engineer can give feedback in how valuable the optimization is brought by this application. Lastly, a LOGwear employee who has no knowledge about the storage process at KLG and will thus focus his feedback on user experience without previous knowledge. The goal with these subjects is to have a large diversity in background and not a large quantity. After the field test the test subjects had to fill in a user experience and usability survey about the application. This survey and the results are part of the test document found in chapter K.

The first operator says the application enabled him to "see very quickly which location is optimal". He responded to all usability scales with four or five out of six, where six is the best. He did not mention any negative aspects of the application. He did mention having contact lenses caused him to have discomfort after 15 minutes of using the application. This might also simply be motion sickness, which some people experience sooner than others. For how easy to use the application is he said quite difficult to use but did not mention why.

The second field test was executed by a warehouse engineer, responsible for designing the high racking pallet rack. He responded with above average for easy to use. For all the other usability related scales he rates four or five out of six as well, the conclusion being he thinks it can be a good optimization to the current process. One of the negatives aspects of the application he mentions is that a network connection is not always easy to make. This was because during the field test there was no Wi-Fi connection available yet in the warehouse. The network connection was setup using a hotspot which did not work as well as hoped. The second negative aspect was that there is no option to scan 1D barcodes with the HoloLens itself. This is a hardware limitation and might be improved in the next version of the HoloLens. Positive aspects mentioned include not needing to measure the height physically and the accuracy of displaying the optimal location while on the move and the accuracy of knowing (the HoloLens) its own position in space.

Lastly, an employee of LOGwear tested the application. He found the application above average easy to use and rated very high on the usability scale (six out of 6 mostly). He did mention a negative aspect not seeing the highlighted location at first, but this was resolved during the second test run. The positive aspect included that operators can save time. In an extra remark he said that tests are required for motion sickness, but he did not mention himself experiencing this issue.

8.2.2 User Experience

To further analyse the user experience, the application is tested in different conditions on predefined criteria. The results of these tests conclude the following. Firstly, gestures are hard to get used to. This is unfortunate but can be bypassed using the HoloLens clicker, an external clicking device that replaces the select gesture. Next, Hologram placement is very good, even when moving parts of the body. Clipping of holograms is completely missing and should be implemented in the future. This has also been added to the product backlog. The cursor is very accurate in various conditions. The detailed results of the user experience tests can be found in section "User Experience" of the testing document in chapter K.

8.2.3 Performance Testing

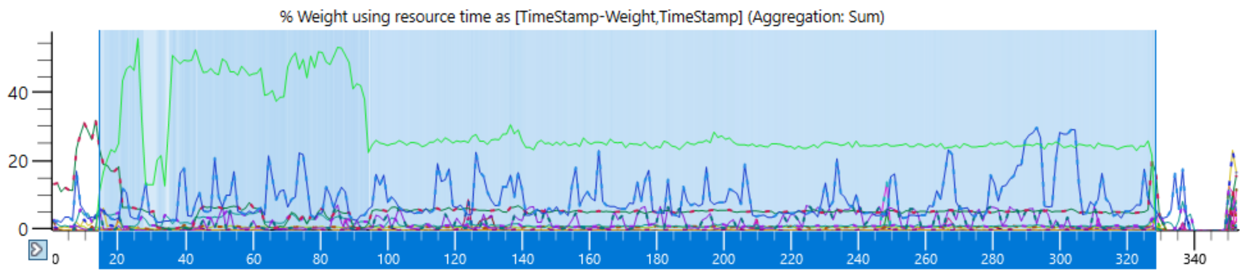
The performance tests are executed to check if the performance requirements set at the start of the project are met. This subsection discusses the individual metrics. All performance tests are done using the Windows Device Portal which allows configuration and management of a HoloLens remotely over Wi-Fi or USB. The Device Portal is a web server on the HoloLens that can be connected to from a web browser on a PC. The Device Portal includes many tools that help to manage the HoloLens and debug and optimize applications. For determining CPU and memory utilisation the performance tracing page on the Windows Device Portal is used. This page allows capturing Windows Performance Recorder traces from the HoloLens. The performance tracing is done for a full routine of activities including start-up, idle, pallet rack model calibration, measuring pallet height, optimal location calculation and guidance to a location in the rack. This complete process is recorded to get a better idea of application performance during these different activities. The result of a performance trace is an Event Trace Log (ETL) file that is then analysed using the Windows Performance Analyzer. This is a tool that creates graphs and tables from the ETL file. Frame rate and power consumption are tested using the system performance page from the Windows Device Portal. This page shows live graphs of the frame rate, system power utilization and System On Chip (SoC) power utilization.

8.2.3.1 CPU Utilisation

Figure 8.1 shows the sampled CPU usage of all processes active on the HoloLens while running the KLG Pallet Racking application. The green line in figure 8.1 represents the sampled CPU usage of the application. A few things can be noted when analysing this figure. The CPU usage for the application has its highest point during the start-up. When the application has finished booting at about 30 seconds system time, the CPU usage drops backs down to about ten percent. The high CPU usage during start-up is caused by the initialization of the pallet rack. The dimensions of the pallet rack are read during this time from a CSV configuration file. For each row and level in the rack location model prefabs are used to instantiate location model GameObjects. They are scaled and positioned based on the configuration file. In figure 8.1 the green line spikes again at about 35 seconds system time, this is when the user after start-up scans the model placement QR code. The complete pallet rack model is activated during this time to give the user the opportunity to calibrate the virtual model with the physical pallet rack. To visualize the complete pallet rack model a lot of computing power is required. After the user is satisfied with the positioning of the model it is disabled. After the 90 second mark CPU usage stays stable at an average of 25%, this includes calculating the position to display the cursor and UI text every frame, which is done constantly at this point. During this time the application is also calculating the closest row every second which takes some CPU usage. At 135 seconds system time a small spike to 30% is visible where the most optimal location in the rack is calculated for a specific pallet height. The following statistics are retrieved from the time that the application is starting until it is shut-down. The average CPU usage is 24.6%, which is even below the ideal value seen in the non-functional requirements in chapter G.

- Run-Time: 314s
- Start Time: 14s
- End Time: 328s
- Average CPU Usage: 24.6%
- Min CPU Usage: 12.1%
- Max CPU Usage: 59.2%

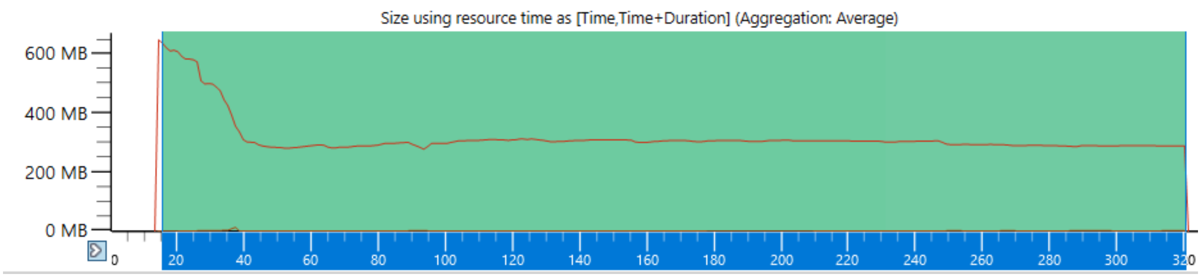
Figure 8.1: CPU Usage Sampled



8.2.3.2 Memory Utilisation

Memory usage stays consistent during a normal routine of using the application. Figure 8.2 shows that at the start more memory is used during the calibration of the rack model. The memory usage stays around 300 MB after that, in practice this means that on average 500 MB is left available. The HoloLens already utilizes about 1.2 GB of the total 2 GB available memory when in use. According to the non-functional requirements maximum memory usage can not be above 900 MB, this will cause the application to crash. The maximum memory usage according to figure 8.2 is 650 MB and thus an acceptable value in the non-functional requirements.

Figure 8.2: Memory Utilisation



8.2.3.3 Frame Rate

The frame rate in normal conditions should always be very close to 60 Frames per Second (FPS). This is achieved during the largest amount of time using the application. On start-up when the pallet rack model must be calibrated, drops in frame rate are experienced when the user tries to look at the complete model. The idea of this calibration is to see if the outer sides of the model (row 1 and 20) line up with the physical rack. Only parts of the model must be in view to validate this, but still frame rates can drop as low as 5 FPS. The calibration is only a small part of the time spent using the application and makes it acceptable in this case, especially considering the overall average frame rate is very close to 60 FPS. This is above the acceptable value seen in the non-functional requirements.

8.2.3.4 Power Consumption

In the appendix chapter J has an overview of power consumption under different circumstances. The figures about system power include the total power consumption for the HoloLens. This includes active sensors, displays, speakers, Graphics Processing Unit (GPU), CPU, memory, and all other components. The figures about SoC power utilization include only the combined power consumption of CPU, GPU, and memory. The results indicate that during the highest system power task, spatial mapping, the average power consumption only increases by about 6%. The average system power usage while idle in the application is about 76%. Considering spatial mapping increases usage of sensors, the GPU and the CPU, there is not a very large increase of power consumption. With an idle power consumption of 76% battery drain is significant enough to have estimated battery life between 130 and 150 minutes. This depends on the amount of time that the operator is busy measuring the height of pallets. This is within the acceptable value specified in the non-functional requirements.

8.3 Automated Testing

As mentioned before, the WMS REST API created by Courtney Regan has been altered to support retrieving pallets from the existing KLG demo environment. Testing and validating REST services in Java is harder than in dynamic languages such as Ruby and Groovy. REST Assured brings the simplicity of using these languages into the Java domain [Haleby, 2018]. The following JSON is returned from the HTTP request “http://demo.logwear.eu/WMS/api/wms/pallets/getAllPallets”:

Listing 8.1: JSON Result from HTTP Request getAllPallets


```
{
"082197100" : {
"customer_id" : "27",
"height" : 148.6999969482422,
"location_id" : "W202210103"
},
"082241900" : {
"customer_id" : "27",
"height" : 102.9000015258789,
"location_id" : "W209020101"
}
}
```


The test in listing 8.2 shows that the HTTP requests is executed by REST-assured. The body of the received response is checked to have the two pallets in it.

Listing 8.2: TestRetrieveAllPallets()

```
@Test
public void TestRetrieveAllPallets() {
get("http://demo.logwear.eu/WMS/api/wms/pallets/getAllPallets").then().body("id", hasItems("082197100",
    "082241900"));
}
```

After updating and testing the REST API, the changes had to be documented. The previous version of the REST API was documented using Swagger, an interactive API documentation tool. This documentation has been updated to reflect the new requests that have been added.

Chapter 9

Using Smart Glasses in Warehouses

This chapter contains research about using smart glasses in warehouse. To be more specific, the use of smart glasses for three common logistic processes in warehouses has been researched. This chapter describes the design, results and conclusion of this research.

9.1 Conceptual Research Design

This section describes the research objective, framework and research questions.

9.1.1 Research Objective

This research is executed at LOGwear during the bachelor thesis project. LOGwear researches and documents Wearables that are suitable for specific activities in logistical processes. The results are processed and added to a knowledge base that is openly accessible via the internet. The bachelor project focusses on using the Microsoft HoloLens for the pallet racking system at KLG Europe. Smart glasses could be useful for different logistical processes, this needs to be researched. Research results include the advantages and disadvantages per logistical process. The results of this research will be added to the knowledge base of LOGwear. Companies like KLG can use the results to determine future usage of smart glasses for their logistical processes.

The objective of this research is to make recommendations and provide data for logistics companies like KLG Europe on using smart glasses for three logistical processes by providing a diagnosis of the specific steps for each process and providing an overview of the advantages and disadvantages per process. The logistical processes that are analysed include "inbound of goods", "order picking" and "restocking". The reason to research these three processes is that they are commonly implemented in warehouses [Rushton et al., 2010 (Fourth edition] [Solutions, 2017]. These processes are related to the storage process that is optimized during this project, but the activities are different. This research is meant to give recommendations after this project, specifically for logistics companies like KLG that have implemented the activities of the processes that are researched. The results will thus have an impact on possible future developments of this project or closely related projects.

9.1.2 Research Framework

The different steps (activities) within three logistical processes must be analysed. These logistical processes are "inbound of goods", "order picking" and "restocking". For every step (activity) within these processes the possibilities of using smart glasses must be analysed. There are some aspects that are important when analysing the activities. Firstly, the hardware and/or software requirements for completing the activities. Another aspect to consider is the effective productivity boost that will be gained, the time it takes to complete an activity. The user experience is also an important aspect. Another aspect that is also included are the costs for executing an activity. The activities within the processes are defined according to previous research done by LOGwear and documentation found in books. The hardware and software functionality and limitations of smart glasses are considered over a wide variety of smart glasses that are currently available in the market.

9.1.3 Research Questions & Definitions

The main question is: "What added value can smart glasses have in logistical processes?". The following set of sub questions have to deliver all information that is necessary to answer the main question.

1. (For every defined process) What are the activities to follow during this process? This includes all the actions an operator executes to complete a process.
2. (For every activity found) Does this activity require any hardware and/or software to be completed? More specifically, are there any devices and/or software systems used for input, processing and output for the operator. This does not include vehicles like forklift trucks. This question is necessary to determine if there is any hardware that can be replaced or tasks that can be taken over by smart glasses. It is also meant to determine if there is any software that can be run on smart glasses instead of a PC or wearable terminal.
3. (For every activity found) What optimization can smart glasses provide to this activity? By optimization is meant what are the use cases for smart glasses for this activity and are there any advantages or disadvantages to the time efficiency and user experience of the execution of this activity.
4. (For every defined process, after all other questions have results) Which type of smart glasses are best fit for this process and what are the advantages and disadvantages of using these smart glasses during this process? Cost differences are also included in the final judgement. Looking at the results of every individual activity in this process is key.

9.2 Technical Research Design

This section describes the strategy and material for this research.

9.2.1 Research Strategy

The first part of the research where processes are analysed will be a typical qualitative and desk research. The second part about advantages and disadvantages of using smart glasses for specific processes will be a mixed research. This part will consist of literature research.

9.2.2 Research Material

This subsection defines per research question and object, which type of information is needed, where to gather it and using which methods.

1. For this question detailed information is needed from existing sources about the three logistical processes defined earlier. The sources include books about process management and IT in logistics. The method to gather the information is literature research. Furthermore, information about the activities is received from experts in logistics that are part of the LOGwear project.
2. This question is a follow-up on the first question. When the activities in the processes are clear, the activities themselves will need to be analysed. The same sources should provide this information and the research method is literature research.
3. To determine the optimization possible information is needed about the possibilities of smart glasses. The sources include documentation on smart glasses and their possibilities and limitations. The research method is literature research.

4. The last question has a literature research to determine the advantages or disadvantages based on existing data.

9.3 Results

This section contains the objective picture of the gathered material. Augmented Reality Smart Glasses are defined as Wearable Augmented Reality (AR) devices that are worn like regular glasses and merge virtual information with physical information in a user's view field. The four major features or capabilities of smart glass technology according to brainxchange [Friedman, 2016] are:

1. Hands-free still image, video and audio capture
2. Real-time sharing of wearer's point of view
3. Hands-free, on-the-spot access to information
4. Augmenting the real world

Smart Glasses are usually worn like glasses, or are devices mounted on regular glasses. Several technologies (e.g., camera, depth-camera, GPS, microphones etc.) capture physical information and augment them with virtual information that can be gathered from the internet and/or stored on the smart glasses memory, primarily accomplished through location-, object, facial, and image-based recognition technologies. This virtual information is then displayed in real-time on a display, which, in brief, is a plastic screen in front of a user's eye(s). A user can see both the offline and the virtual and the real-world through these displays. Prominent examples of smart glasses are Microsoft HoloLens [Microsoft, 2018b] or Google Glass [LLC., 2018]. It is worth noting that a related technology, virtual reality glasses (Virtual Reality (VR) Glasses), needs to be distinguished. Unlike AR smart glasses, where digital content is overlaid onto the real world, VR glasses are completely closed off from the real, physical world, and instead present only a virtual world. [Rauschnabel, Brem, and Ro, 2015]

As mentioned before three common logistical processes are used for this research. LOGwear researches these processes and documents the activities per process. The sources include books and experts in logistics. The process "inbound of goods" refers to receiving goods that are delivered to the warehouse. "Order picking" is a logistical process that consists of taking articles in a specific quantity, collecting them for an order. The last process, "restocking" consists of replenishing the stock in a warehouse. For every company this process can be different, some activities might not be executed, are missing or activities are executed in a slightly different order. LOGwear used the following sources to identify and document the activities that are specific to this process. [Hausladen, 2011, 2014] [Koch, 2012] [Rushton et al., 2010 (Fourth edition] [Luft and Salz, 2018]. Table L.1 shows the activities and a description per activity for the "inbound of goods" process. Table L.2 contains the activities for the "order picking" process. Lastly, table L.3 has the activities of a typical "restocking" process. The existing hardware and software requirements per activity are also included in these tables and are part of the research executed by LOGwear. The last column defines the use case(s) for smart glasses during the execution of the activity. The use case(s) of smart glasses defined in the tables in chapter L are based on the possibilities of smart glasses currently, mentioned earlier. The most common use case is substituting hand held devices like scanners and Wearable terminals to interact with a WMS and as a result increasing the productivity. [Kohles, 2017]

9.4 Research Conclusion

The objective of this research is to provide information on the advantages and disadvantages of using smart glasses for three commonly implemented logistical processes.

9.4.1 Inbound of Goods

As mentioned before the research results for the Inbound of Goods process are found in table L.1. The biggest advantage of using smart glasses for this process is being able to replace the Personal Computer (PC) or Wearable terminal with smart glasses. Most of the interaction with a WMS can be done via smart glasses. This can give an increase in productivity while decreasing errors and subsequent costs. Using voice recognition or gesture input the operator can request the available pallet IDs when receiving the packing slip from the transporter. When unloading pallets, the smart glasses can help increase productivity even more when it guides the operator to empty spots to place the pallets and boxes. Normally the operator would have to count and compare the unloaded goods with the goods that have been ordered. Smart glasses can increase productivity by keeping track of the goods that are unloaded from the truck and in the end validating the quantity without any extra work required from the operator. Furthermore, if the quantities do not match it can warn the operator and register the error in the system. Documents that need to be printed or send to a customer do not require any access to a PC or Wearable terminal. Instructions can be given via the smart glasses to print a document on a nearby printer. When moving the order smart glasses can guide the operator to the correct location using the shortest path, decreasing errors. Another big advantage is that for scanning barcodes no more hand scanners or ring scanners are required. Smart glasses have cameras that replace the need to carry an extra scanning device. This decreases the costs by removing the need of purchasing scanning devices. Furthermore, it allows the operator to have his hands free, increasing safety and productivity. As mentioned before augmenting the real world is one of the main capabilities of smart glasses. In this case sorting pallets, loading pallets and applying pallet ID labels are all activities that can be supported visually, increasing productivity once more. When entering label information, no PC or Wearable terminal is required if voice commands are used. Requesting storage locations can easily be done using voice commands and the operator can then be visually guided to the pallet storage location. Lastly, once arrived at this location the smart glasses can scan the location barcode, removing once again the need for external scanning devices.

The mentioned use cases require a pair of smart glasses that has the following capabilities and specifications:

- Possibility to run a custom build application that supports real-time connection to a WMS
- High-end camera for object and image recognition (e.g. barcodes)
- Spatial awareness functionality for path finding / navigation through the warehouse
- User input through either gestures or voice commands

9.4.2 Order Picking

The research results for the Order Picking process are found in table L.2. For the order picking process some advantages of using smart glasses are being able to replace the PC or Wearable terminal with smart glasses and navigation to locations that need to be visited. Most of the interaction with a WMS can be done via smart glasses. This can also give an increase in productivity while decreasing errors and subsequent costs. This process starts with the operator requesting the order form, which in this case can be placed directly in the view. When satisfied the operator can start the order processing using a voice command or gesture. The operator can be helped to pick transport equipment by determining and visually indicating the optimal equipment to the operator. This can lead to a higher productivity when working with the optimal equipment. For this process the external scanning device can also be replaced by the smart glasses. Scanning the order ID and location ID can be done by the smart glasses. Guidance to an empty pallet, to locations and regroup locations that must be visited is also possible for the order picking process using smart glasses. More advanced object recognition is required for recognizing and indicating the boxes that need to be collected but this is already possible. The same goes for recognizing the number of articles picked, where an advanced object recognition algorithm would have to keep track of the number of articles picked. This can greatly increase productivity and decrease the amount of errors. After recognizing the picked items packing requirements and customer wishes can be shown. Accessing a PC or Wearable terminal can be replaced by using smart glasses to instruct a print of packing slip and shipping information. Lastly, the operator can use voice commands or gestures to instruct the order to be reported ready for shipping in the system.

The mentioned use cases require a pair of smart glasses that has the following capabilities and specifications:

- Possibility to run a custom build application that supports real-time connection to a WMS
- High-end camera for object and image recognition (e.g. barcodes)
- Spatial awareness functionality for path finding / navigation through the warehouse
- User input through either gestures or voice commands

9.4.3 Restocking

The research results for the Restocking process are found in table L.3. With the order picking process advantages of using smart glasses are being able to replace the PC or Wearable terminal with smart glasses. Here also most of the interaction with a WMS can be done via smart glasses. This can also give an increase in productivity while decreasing errors and subsequent costs. Starting a new order can be done using voice commands or gestures. The number of products to be resupplied can be shown in the view of the operator. When moving to the stock location there can also be the possibility of navigation. Scanning the product ID does not need any more external scanning devices. Advanced object recognition can recognize the number of products and automatically compare the inventory with the inventory registered in the system. This can greatly increase the productivity and costs when no more device is needed to measure the weight and volume of the products to automatically determine the number of products. Recognizing the stock that needs to be corrected for ripening and shelf life can make the life of the operator a lot easier. Requesting system inventory level, adjusting system inventory level, requesting the maximum stock level, adjusting the maximum stock level, entering the order quantity and requesting the delivery data are all commands that can be executed using the smart glasses. The response can then be visualized in the view of the operator. Lastly, placing the order can also be done using a voice command or gesture.

The mentioned use cases requires a pair of smart glasses that has the following capabilities and specifications:

- Possibility to run a custom build application that supports real-time connection to a WMS
- High-end camera for object and image recognition (e.g. barcodes)
- Spatial awareness functionality for path finding / navigation through the warehouse
- User input through either gestures or voice commands

9.4.4 Evaluation

Smart glasses can bring a lot of optimizations in terms of productivity, error reduction and safety. Warehouse operator can work hands free while having information that they require in their view. The applications are not limited to these processes, the use of smart glasses could be widespread and is not only limited to logistical organizations. The most obvious reason is that the implementation and integration of these kinds of applications costs too much money. Considering the costs of purchasing smart glasses for operators alone is not enough. The software system on the smart glasses is also sophisticated and requires a big investment. A good follow-up research would be to find out why exactly the use of smart glasses in logistics is not widespread and if only the costs are preventing companies from using smart glasses in their warehouses.

Chapter 10

Conclusion and Evaluation

Since the start of the project sixteen weeks have passed. The scope of the project was made final at the beginning of the third week, the first two weeks were mostly orientation and start of the project plan. Week three and four were used for an extensive analysis of the situation and design of solutions. During this time the final version of the project plan was delivered as well. Between the midterm report and this report another eleven weeks have passed, a total of six sprints in this time for implementing the requirements of the project.

In the midterm report, the implementation of the products from the first use case looked promising for a very practical solution for KLG. This first version was far from perfect, improvements were needed for recognizing new surfaces faster and then still maintaining acceptable performance according to the requirements. It was already possible though to make very accurate measurements once the environment was mapped sufficiently. The highest priority after implementing height measurements was guiding an operator to the most optimal location and was predicted to be even more challenging. Firstly, the pallet racks in the warehouse have been translated to a scale 3D model that can be used in the application to determine the exact location in the physical world to render holograms. Using prefabs eventually a dynamic warehouse has been created, at least in terms of height. Next, the most optimal location has been determined based on empty locations and the location height compared to the pallet height. The existing KLG demo environment REST API and database have been adapted so that it supports requests for registering pallet's and their height. Pallets and locations can be scanned for registration of pallets and their location in the system. All the functional requirements have been implemented to deliver a complete proof-of-concept. Results of field tests conclude that all respondents think the system can enhance effectiveness on the job for operators. The difficulty of using the application has mixed responses but all find the interaction with the system clear and understandable. If the system would be used in a live environment the results of the field tests indicate improvements are needed to make the application easier to use. A handover document still needs to be created in the next few weeks after delivery of this report. It is meant for KLG and LOGwear to learn how to change, build and deploy the application.

Research has been executed to make recommendations on implementing smart glasses for other logistical processes. The results show a lot of different use cases for smart glasses in warehouses ranging in difficulty to implement. It is worth investing time and effort in this because it can increase productivity and safety in warehouses. Furthermore, some use cases can decrease the amount of errors made by operators.

There was a concrete problem description at the start of the project. The analysis and design sprint ensured an effective approach to the problem and a good design of solutions. Trello has been used to effectively carry out project management. On the trello board made for this project, products and tasks are planned, prioritized and discussed during biweekly sprint meetings with the stakeholders. This ensured effective feedback communication, I am pleased that this went well and because of that a very nice result was achieved. I am very proud of the results of the project. I think the proof-of-concept has multiple use cases for KLG as well as LOGwear. For LOGwear, the project will be used purely as a demo, but it does show a practical example of wearables in logistics. In the future LOGwear would like to use the application for demos without being at the high racking warehouse at KLG, for this the project will have to be continued to ensure a complete dynamic pallet rack model in terms of scale. I hope that the project can be picked up by KLG in the future so that it can eventually be used in a live environment. Furthermore, I advise KLG to look at the use cases of smart glasses in other processes within their warehouses. A more in-depth reflection is in the separate reflection report.

References

Emily Friedman. Top features of smart glasses: See-what-i-see communication, 2016. URL https://brainxchange.io/top-features-smart-glasses-see-what-i-see-communication/. Accessed 07-06-2018.

Johan Haleby. Rest-assured testing and validating rest services in java, 2018. URL http://rest-assured.io/. Accessed 07-06-2018.

Iris Hausladen. *IT-gestützte Logistik*. 2011, 2014. ISBN 978-3-8349-4664-5.

KLG. Logistic solutions klg europe, 2018. URL https://www.klgeurope.com/en/logistic-solutions. Accessed 07-06-2018.

Susanne Koch. *Logistik Eine Einführung in Ökonomie und Nachhaltigkeit*. 2012. ISBN 978-3-642-15288-7.

Camila Kohles. Smart glasses: use cases, challenges and future potential, 2017. URL https://www.wikitude.com/blog-smart-glasses-challenges-future/. Accessed 07-06-2018.

X Development LLC. Google glass, 2018. URL https://x.company/glass/. Accessed 07-06-2018.

LOGwear. Logwear, 2018. URL https://logwear.eu/. Accessed 07-06-2018.

Alex Luft and Christian Salz. Geschäftsprozessmodell in der logistik, 2018. Retrieved 29-5-2018.

Microsoft. Mixed reality toolkit unity, 2018a. URL https://github.com/Microsoft/MixedRealityToolkit-Unity. Accessed 07-06-2018.

Microsoft. Microsoft hololens, 2018b. URL https://www.microsoft.com/nl-nl/hololens. Accessed 07-06-2018.

Microsoft. Using the windows device portal, 2018c. URL https://docs.microsoft.com/en-us/windows/mixed-reality/using-the-windows-device-portal. Accessed 07-06-2018.

Philipp A. Rauschnabel, Alexander Brem, and Young K. Ro. Augmented reality smart glasses: Definition, conceptual insights, and managerial importance, 2015. URL https://www.researchgate.net/profile/Alexander_Brem/publication/279942768_Augmented_Reality_Smart_Glasses_Definition_Conceptual_Insights_and_Managerial_Importance/links/5721ec2e08aee857c3b5dd6c/Augmented-Reality-Smart-Glasses-Definition-Conceptual-Insights-and-Managerial-Importance. Retrieved 07-06-2018.

Courtney Regan. Creating a klg europe simulation environment to test different wearables, 2018. Retrieved 9-2-2018.

Alan Rushton, Phil Croucher, and Peter Baker. *The Handbook of Logistics and Distribution Management*. 2010 (Fourth edition). ISBN 978-0-7494-5714-3.

Bastian Solutions. Warehouse management system processes, 2017. URL https://www.wikitude.com/blog-smart-glasses-challenges-future/. Accessed 07-06-2018.

Trello. Klg pallet racking trello, 2018. URL https://trello.com/b/m4my520H/klg-pallet-racking. Accessed 07-06-2018.

Vuforia. Vuforia, attach digital content to specific objects, 2018. URL https://vuforia.com/features.html. Accessed 07-06-2018.

Zebra. Rs6000 bluetooth ring scanner, 2018. URL https://www.zebra.com/us/en/products/mobile-computers/wearable-computers/rs6000.html. Accessed 07-06-2018.

Appendix A

Sprint Planning

Table A.1: Trello Board Export 31/05/2018 Part 1

List	Title	Labels
Product backlog	Sprint 9 (11 juni t/m -)	
Product backlog	Deployment proof-of-concept	Task
Product backlog	User manual - How to setup and deploy the application, how to use it	Task,High Priority
Current Sprint	Sprint 8 (28 mei t/m 8 juni)	
To Do	Update swagger API documentation	Task,High Priority
To Do	Optimal location model not activated on rare occasion	Bug,Low Priority
To Do	Sometimes the spatial mapping is not updating when the sphere is picked up.	Low Priority,Bug
To Do	Ground plane creation is for area of 20x20 meters, not complete surface of the warehouse	Issue,Low Priority
To Do	Scanning a pallet barcode while a location barcode is expected causes optimal location to be stuck highlighted on the same lo	Low Priority,Bug
To Do	Hologram clipping not enabled.	Issue,Low Priority
In Progress	Final report	Task
In Progress	Research if smart glasses could be useful for different logistical processes	Task
Waiting for customer approval	QR code scannen in UI text	Issue,High Priority
Waiting for customer approval	Text sphere en UI text	Issue,High Priority
Waiting for customer approval	Logo only on startup, add fontys and logwear	New feature,High Priority
Waiting for customer approval	Waiting for location scan	New feature,High Priority
Waiting for customer approval	Remove test model	Task,High Priority
Waiting for customer approval	Cursor stuck in sphere	High Priority,Bug
Waiting for customer approval	LOGwear logo hardly visible	Issue,High Priority
Waiting for customer approval	Setup field-tests with goals and recommendations	Task,High Priority
Waiting for customer approval	Demo video	Task,High Priority
Waiting for customer approval	Testing of the proof-of-concept in a production environment with operators	Task,High Priority
Waiting for customer approval	Sprint 7 (14 mei t/m 25 mei)	
Waiting for customer approval	Testing REST API using rest assured	Task,Low Priority
Done	Sprint 1 (5 t/m 23 feb)	
Done	Analysis of the situation	Task
Done	Analysis of pallet racking system at KLG	Task,High Priority
Done	Sprint 2 (26 feb t/m 9 mrt)	
Done	Add prototyping folder to bit bucket	Task,High Priority
Done	Requirements gathering	Task
Done	Design of 3D scene for the pallet racking system	Task,High Priority
Done	Prioritize use cases per step	Task,Medium Priority
Done	Start with research, what and how	Task,High Priority
Done	Project plan	Task
Done	Sprint 3 (12 mrt t/m 30 mrt)	
Done	Prepare workshop demo TEC7 (Automatic identification techniques)	Task,High Priority
Done	Mid-term self-reflection report	Task,High Priority
Done	Mid-term report	Task,High Priority
Done	Mid-term: add use case diagram	Task,Medium Priority
Done	Mid-term: add software design section (including activity/sequence diagram)	Task,High Priority
Done	Mid-term: add overview current architecture KLG simulation and what is going to be adapted for this project	Task,High Priority
Done	Mid-term: add demo video to results	New feature,High Priority
Done	Mid-term: change png / jpg images to pdf	Issue,High Priority
Done	Rework research design to include AR in general for the logistical processes	Task,Medium Priority
Done	As an operator, I want to measure the height of a pallet so that I know where it will fit	User story,High Priority
Done	Display selection indicator on top of objects instead of half inside	Issue,High Priority
Done	Distance to ground based on largest distance raycast hit gives problems with spatial meshes below ground plane	Issue,High Priority

Table A.2: Trello Board Export 31/05/2018 Part 2

Done	Update spatial mapping more frequently and detailed	Issue,High Priority
Done	Calculate and display distance between highest point and ground plane	High Priority,New feature
Done	Display selection indicator for highest point on "air-tap" gesture	High Priority,New feature
Done	Create ground plane based on spatial mapping floor	High Priority,New feature
Done	Direction indicator to show current position of the sphere	New feature,High Priority
Done	Start up screen during scanning process and ground plane creation	New feature,High Priority
Done	Mid-term presentation	Task,High Priority
Done	Sprint 4 (2 apr t/m 13 apr)	
Done	Import warehouse pallet rack layout measurements from a CSV file	New feature,High Priority
Done	Dynamically create 3D model of the warehouse based on data from CSV file	High Priority,New feature
Done	Display best fit locations using holograms	New feature,High Priority
Done	Algorithm for determining best fit location(s) based on pallet height	High Priority,New feature
Done	Add indication that ball is placed (e.g. change color)	New feature,Low Priority
Done	Can't read file when build to HoloLens	Issue,High Priority
Done	Increase size of sphere for easier handling	Issue,Medium Priority
Done	Known spatial mapping appears about 2 meters lower than actual floor	Bug,High Priority
Done	Sprint 5 (16 apr t/m 27 apr)	
Done	As an operator, I want to see which locations the pallet is best fit so that I can make a better choice	User story,High Priority
Done	Add select sound to sphere	New feature,High Priority
Done	Determine optimal location for row closest to HoloLens	New feature,High Priority
Done	Show 3D model of warehouse exactly overlaying the physical environment	New feature,High Priority
Done	Rows with extra space differ in extra space	Issue,High Priority
Done	LOGwear simulation database entry for pallet ID, height and location	Medium Priority,New feature
Done	API Request to DB for registering pallet height and ID	Medium Priority,New feature
Done	Improve accuracy of height measurements by testing what causes the inaccuracy	Issue,Medium Priority
Done	Rotate model after QR code scan for exact calibration	New feature,High Priority
Done	Sprint 6 (30 apr t/m 11 mei)	
Done	As an operator, I want to scan a location barcode so that I can enter the pallet's location into the system	User story,High Priority
Done	As the product owner, I want to see the demo work in different environments	User story,High Priority
Done	Scan pallet barcode and display info	Medium Priority,New feature
Done	Scan location barcode to register pallet location	Low Priority,New feature
Done	Confirmation to register pallet location	Low Priority,New feature
Done	API Request to DB for registering pallet location	Low Priority,New feature
Done	Disable warehouse model after initial calibration	New feature,High Priority
Done	Optimal location model not displayed	High Priority,Issue
Done	Disable location model after location registration	Issue,High Priority
Done	Directional indicator to best fit location(s)	Medium Priority,New feature
Done	All directional indicators to locations are active when model is being placed, causes massive frame rate drop	Issue,High Priority
Done	Option to move sphere back in front of user using a voice command or gesture	New feature,Low Priority
Done	Directional indicator to a location points not completely correct	Issue,High Priority
Done	Livestream not available while using vuforia framework	Issue,High Priority
Done	Direction indicator object remains in scene after disabling it	High Priority,Bug
Meetings	Intro	
Meetings	Prioritizing requirements, prototype scene	
Meetings	Sprint 3 meeting	
Meetings	Sprint 4 meeting	
Meetings	Sprint 5 meeting	
Meetings	Sprint 6 meeting	
Meetings	Sprint 7 meeting	

Appendix B

Project Organization

The stakeholders of this project are listed below with their role in the project. The description also contains a statement about their power and interest in the project. This gives a diagram as seen in figure B.1 where the power and interest gives an indication on how to manage these stakeholders.

- The student

  Expected to work independently in preparation, execution and completion; a proactive attitude is required. A suitable assignment is found, executed autonomously, and organizes (additional) feedback within the framework of the graduation.

- LOGwear project lead

  The scope of the project is within the goals for the LOGwear project. Because of this the project lead of LOGwear has a lot of interest in the project. They also have a lot of power, they could stop the project at any given moment if not satisfied. They also control the changing requirements and can add new ones when necessary.

- KLG Business Development Manager

  Peter Jacobs is the Business Development Manager at KLG. Peter will be highly involved in the project and is very interested in the outcome. During the internship period Peter will be present in customer meetings and can give feedback about the process and products.

- Warehouse Manager

  Ron van Doorn is the Warehouse Manager at KLG. Ron will also be highly involved in the project and will be using the proof-of-concept in practice when it turns out successful. He is responsible for managing the operators that will be using the application. Ron is also present during the customer meetings at KLG and will be able to give feedback about the process and products as well.

- Warehouse Engineer

  René Geurts is a Warehouse Engineer at KLG. René is also highly involved in the project. He is among other things responsible for creating an optimal layout for the pallet rack. René is present during the sprint meetings and can give feedback on the process and products that are created.

- KLG warehouse storage process employees

  The employees at KLG that operate during the storage process are part of the testing phase of the project. They are the end-users and have valuable feedback and decide eventually if the project could be used in a practical production environment.

- Internship-graduation office

  The office is responsible for organising all issues relating to the graduation, as well as archiving documents and information. The office:

  - receives accepted assignments, provides the study programme with the assignments (portal), and informs students about the approval; collects student' products;
  - contacts and informs organisations about the internship and the responsibilities of the organisations;
  - ensures that information and reports are passed on to the supervising lecturer, the examiner and the external commissioner and is responsible for archiving;
  - plans and administrates interim- and final presentations;
  - ensures that the graduation process is completed by (digitally) sending out and receiving evaluation forms.

  This gives the office a lot of interest in the project, but they do not have a lot of influence in how the project develops.

- Supervising lecturer

  The task of the supervising lecturer is to supervise in carrying out the assignment. Supervision may relate to the (technical) content of the project and will certainly focus on the process itself. The lecturer has high interest in the project because he must supervise it and has high power because he can advise changes in the process.

- Company supervisor

  One or more persons from the company are responsible for day-to-day supervision within the company. From the perspective of the student, the company supervisor often has the role of client. In determining the assessment, his advice on the performance of the student will be heard and considered by the examiner. The company supervisor is an expert in the field of the assignment. Of course, the company supervisor also has a lot of interest in the project, he has to supervise it. The power is also very high because as the customer of the object he determines the requirements that must be implemented.

- External commissioner

  Every graduation project is monitored by an external subject expert, when available, the external commissioner. This is someone from the private sector who has the task of monitoring the level of the graduation project. He is an expert in the field and has a clear view of the level of performance of the individual student, as well as the level of performance of 'his' graduates. To assess your performance, he reads the interim and final report and is present during the interim and final presentation. In determining the assessment, the commissioner serves as an important consultant to the examiner. Furthermore, he comes from a different company than where you are completing your graduation. Involvement in the study programme (e.g. confer about study programme content, the end level, or improvements in the curriculum) is a key aspect. The external commissioner has high interest because he plays a role in grading the assignment. His power in the project is also medium to high because he reads the mid-term report and advices on how to continue the project and what to do for a successful ending.

- Examiner

  The examiner bears final responsibility for assessing your performance. For the assessment, he will be supported by the supervising lecturer, the company supervisor and the external commissioner. Like the external commissioner he has medium to high power and high interest in the project.

- Internship-graduation coordinator

  The internship-graduation coordinator plays an important role in the preparation phase of the assignment. In addition, the coordinator has a shared responsibility in (yearly) preparations of and advice about the policies concerning internship and graduation. He also initiates process and organisational improvements within the framework of internship and graduation. The graduation coordinator has low to medium interest in the project after it has been initiated. The coordinator also has low power because she has no influence on the requirements of the project.

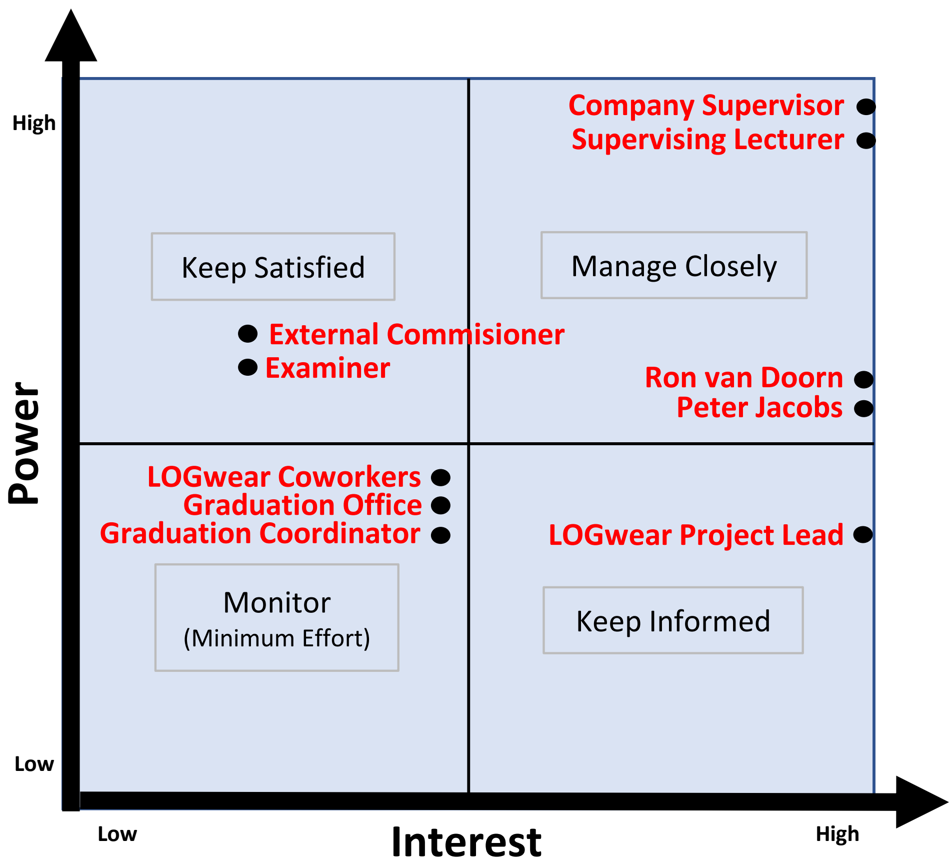
- LOGwear customers

  The partner companies that LOGwear works with are highly interested in the findings of the project research, but they only have medium power during this project. They will not be present during all customer meetings but may have input in the requirements throughout the project.

- LOGwear co-workers

  The co-workers have low to medium interest and power in the project. They will not be present during customer meetings but can give feedback during the project. Also, they are not directly part of the process so are less interested.

Figure B.1: Stakeholder Analysis



Appendix C

Risk Analysis

A risk analysis has been conducted to identify and classify the potential problems that might occur during the project. Table C.1 categorizes risks based on probability of them occurring and their impact on the project. This determines a risk level that is used in table C.2 which gives the complete overview of all the risks identified for this project. Table C.2 also gives a description of the risks and the actions that will be undertaken to minimise the impact.

Table C.1: Probability and Impact Matrix

		Impact				
		Trivial	Minor	Moderate	Major	Extreme
Probability	Rare	Low	Low	Low	Medium	Medium
	Unlikely	Low	Low	Medium	Medium	Medium
	Moderate	Low	Medium	Medium	Medium	High
	Likely	Medium	Medium	Medium	High	High
	Very likely	Medium	Medium	High	High	High

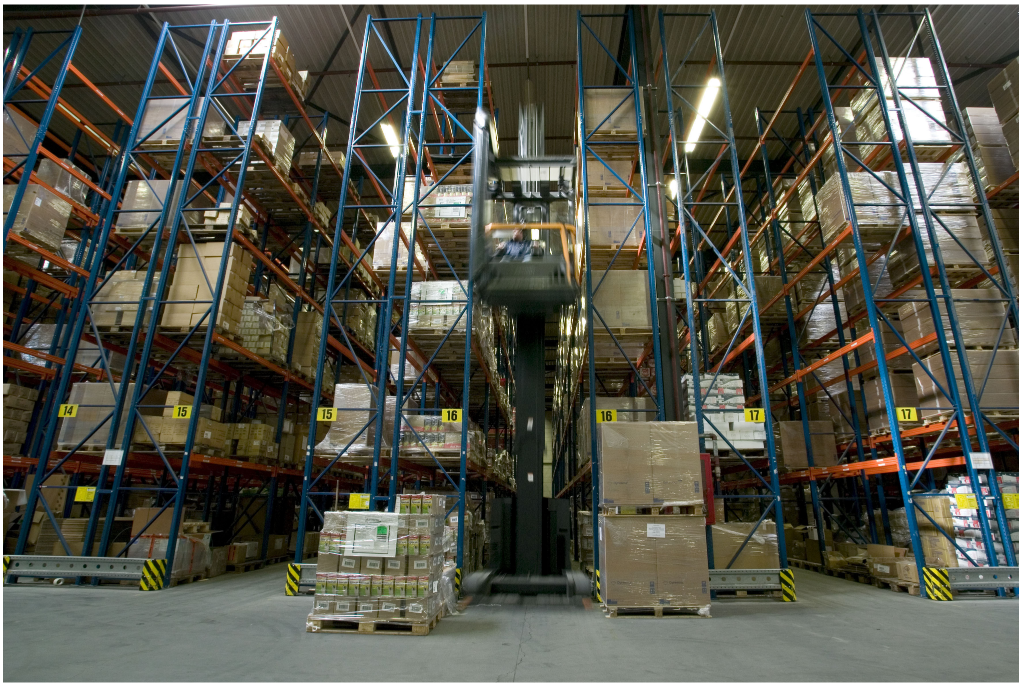
Table C.2: Risk Analysis

#	Risk	Description	Probability	Impact	Risk Level	Actions
1	Time issues	Every software project is unique and makes making a correct planning difficult. Work load has to be estimated and if done wrong time issues may arise.	Moderate	Moderate	Medium	Good planning and dividing the tasks up into smaller tasks gives a smaller impact.
2	Improper use of new technologies	During the project new technologies, including hardware and software are used. This could lead to project failure because of not enough training and knowledge. Another problem might be time issues because training is required during the project and thus less time is over for actual development.	Moderate	Major	Medium	Make sure documentation is readily available.
3	Big changes in requirements	The requirements that are received at the start of the project might have big changes during the project. The customer might want requirements to be implemented differently or add completely new ones. This may cause time issues because the proof-of-concept might not be done on time.	Very likely	Moderate	Medium	Scrum will be used for many feedback moments and thus fast response to changes in requirements.
4	Wrong choice of platform	During the design phase there might be a wrong choice of platform for a specific part the application. Depending on how late during the project this becomes a problem it might have disastrous consequences for the project.	Unlikely	Extreme	Medium	Discussions with other software developers within the company to make the right decisions.
5	Performance of the application is poor	Because of bad optimization of the application or hardware limitations the performance might be not on the expected level of the customer. Since using spatial mapping for measurements is one of the requirements a lot of heavy computations must be done on the HoloLens, which is an unavoidable performance hit. If performance in terms of FPS is below certain thresholds the application might make you nauseous and would make the project a failure.	Likely	Major	High	Turn down graphical quality and optimize application for best possible performance.

Appendix D

Man-up Forklift Truck

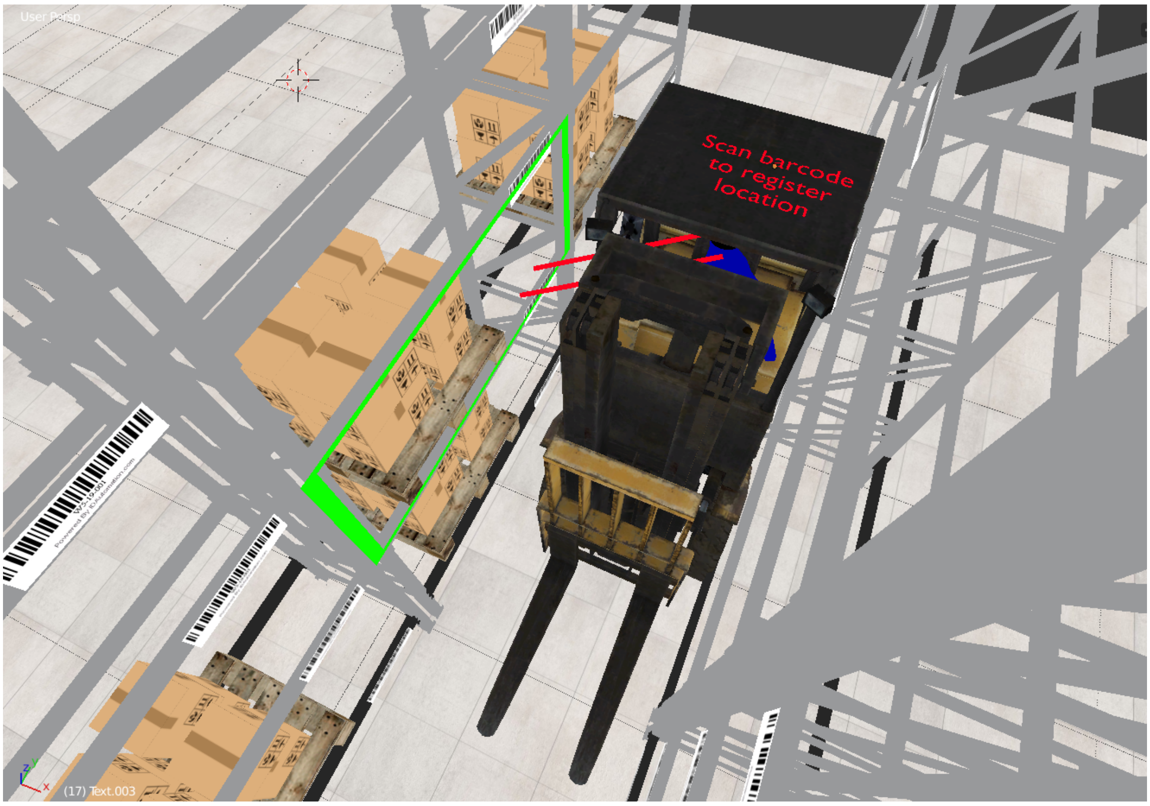
Figure D.1: Man-up Very Narrow Aisle Forklift at KLG Venlo [KLG, 2018]



Appendix E

Prototype

Figure E.1: Prototype Register Pallet Location



Appendix F

Measuring Height Demo

This video can also be found on youtube: `https://www.youtube.com/watch?v=hm5UIBH5IYI`

Figure F.1: Video: Measuring Height using Spatial Mapping

Appendix G

Requirements Specification

Requirements Specification
For the HoloLens Pallet Racking Application

Niels Killaars (2521806)

Fontys Venlo Techniek en Logistiek
Informatics
Graduation

Venlo, June 10, 2018

Contents

Chapter 1

Introduction

LOGWear is a research project and focusses on documenting how wearables can be used in logistics processes, e.g. Order Picking, to improve the work process. LOGWear also wants to do research on the Microsoft HoloLens specifically in logistical processes. Project partner KLG Europe came with a very specifc problem in their pallet racking storage system. The pallets with products that have to be stored are varying widely in their total height. This means that the operator that is responsible for storing the pallet in the rack has to choose a location which is best fit in terms of height. The goods are stored in racks from which the different locations are also varying in size going higher up. When the operator chooses to put the pallet in a location that is much higher than the actual height of the pallet a lot of space is wasted. When the operator makes this mistake multiple times, eventually there might be pallets that can not be stored because all the empty locations are too small in terms of height. This could have been prevented if the smaller pallets were placed in the most optimal locations. This problem exists for only a few customers of KLG that have a lot of variation in the total height of pallets that need to be stored.

To solve the problems at KLG's pallet racking system an application will be developed for the Microsoft HoloLens. The idea is that this application will help operators make better decisions. This document focusses on the requirements for this application.

Chapter 2

Functional Requirements

2.1 User Stories

User Story	**Operators can select a pallet to measure the height**		
Code	US-1		
Package			
File	**undefined**.tex		
Story	Operators of a forklift truck have to store pallets in a location on the rack based on the total height of a pallet. The operator wears the HoloLens. The operator can select a pallet using the airtap gesture, the HoloLens then makes a 3D mapping of the pallet. This 3D model is then used to determine the height of the pallet.		
Refined by	UC-1		
Version	1.0	Author	Niels Killaars

User Story	**Operators can see which location is best fit for the current pallet**		
Code	US-2		
Package			
File	**undefined**.tex		
Story	After the operator measured the height of a pallet he wants to store it at the most optimal location. The system checks the free locations available and calculates which location will have the least wasted space. The HoloLens displays to the operator using holograms which location is best fit.		
Refined by	UC-2		
Version	1.0	Author	Niels Killaars

User Story	**Operators can enter the location of a pallet into the system**		
Code	US-3		
Package			
File	**undefined**.tex		
Story	Operators have to register the location of a pallet into the system. All locations have unique barcodes, which are saved in the system. The operator scans a location barcode to register a pallet's location in the system.		
Refined by	UC-3		
Version	1.0	Author	Niels Killaars

2.2 Use Cases

Use Case	**Operator registers a pallet for storage**		
Code	UC-1		
Package	KLG		
File	**undefined**.tex		
Actors	Operator with forklift working at pallet racking system		
Description	Register a pallet for storage in the rack, scan the ID and measure the height		
Precondition(s)	A pallet with products is at starting location, operator is wearing the HoloLens with the application running		
Scenario	1. Operator scans the pallet's barcode to register the pallet ID 2. System displays the scanned pallet's ID and asks the user to select the pallet. 3. Operator selects the pallet using the air-tap gesture. 4. System makes a 3D mapping of the environment, including the pallet and shows this to the user. 5. Operator selects the highest point of the products that are on the pallet using the air-tap gesture. 6. System places a hologram on the highest point according to the 3D mapping, displays the height and asks user to verify this being the highest point. 7. Operator selects" 'register height' 8. System registers the height of this pallet in the database.		
Extensions	**6.1** System reports that a ground-plane / floor-plane has not been created yet 1. System displays: Please scan the ground. 2. Operator scans the ground. 3. Use case proceeds at step 5.		
Result	The pallet's ID and height is registered in the system.		
Version	2.0	Author	Niels Killaars

Use Case	**Operator moves a registered pallet to the optimal location in the pallet rack**		
Code	UC-2		
Package	KLG		
File	**undefined**.tex		
Actors	Operator with forklift working at pallet racking system		
Description	Move a pallet to storage location according to the most optimal location		
Precondition(s)	A pallet which has been registered is at starting location, operator is wearing the HoloLens with the application running		
Scenario	1. System calculates the optimal storage location in the pallet rack based on the height of the pallet. 2. System displays arrow to indicate the direction to which the operator should move and marks the location. 3. Operator follows instructions to indicated location		
Extensions	**4.1** Operator ignores direction of arrow. 1. Operator moves pallet to different location as indicated. 2. Use case ends here.		
Exceptions	**2.1** No fitting locations are left. Use case ends here.		
Result	The pallet is moved to a location in the pallet rack for storage		
Version	2.0	Author	Niels Killaars

Use Case	**Operator registers location of pallet**		
Code	UC-3		
Package	KLG		
File	**undefined**.tex		
Actors	Operator with forklift working at pallet racking system		
Description	Register a pallet's location in the system		
Precondition(s)	A registered pallet has been moved to a location in the pallet rack, operator is wearing the HoloLens with the application running		
Scenario	1. System is waiting for operator to register the storage location of the pallet. 2. Operator scans a location barcode. 3. System displays that the location is registered to this pallet.		
Exceptions	**4.1** The scanned location is already registered to another pallet in the system. Use case ends here.		
Result	The pallet is registered in the system to a location in the pallet rack		
Version	2.0	Author	Niels Killaars

Chapter 3

External Interface Requirements

3.1 Hardware Interfaces

The HoloLens application shall get scan data input from an external barcode scanning device called "ProGlove". The communication between the HoloLens and the ProGlove will be difficult. The ProGlove communicates with their own accesspoint connected to a computer. The computer then communicates with the HoloLens via bluetooth.

3.2 Software Interfaces

The HoloLens application shall send and receive data via HTTP requests. These requests go via the already existing API from LOGwear that was created for the KLG simulation. Requests will need to be added specific for the pallet racking process at KLG.

Chapter 4

Non-functional Requirements

This project is about creating a proof of concept and all involved products are newly developed or still in development which makes it very hard to estimate how realistic the below defined requirements and numbers are.

Performance Key performance metrics for the pallet racking HoloLens application* :

Metric	Acceptable Value	Ideal Value	Actual Value
Maximum Memory usage	< 900 MB	< 500 MB	650 MB
Average CPU usage	$\leq$ 50%	$\leq$ 25%	24.6%
Average FPS	$\geq$ 55 FPS	$\geq$ 60 FPS	$\{x \mid 55 \leq x \leq 59\}$
Battery drain (From 100%)	2 hours	$\geq$ 3 hours	130-150 minutes

Table 4.1: Performance Metrics

*: The ideal and acceptable values for performance metrics are based on the performance recommendations for HoloLens by Microsoft available at:
`https://docs.microsoft.com/en-us/windows/mixed-reality/performance-recommendations-for-hololens-apps`
If the acceptable performance metrics are not met there will be drastic consequences to the immersive experience. Further more, thermal performance is impacted and battery drain will increase because power consumption will be too high. To ensure the performance metrics are met the development process guidance is used also available at the previously mentioned URL.

Safety

1. It **shall** be ensured that all actions or events that trigger a database transaction verify that the transaction is executed completely and correctly.

Reliability

1. The HoloLens **shall** be able to make a measurement of a pallet's height that is on average in a range of 10 centimeters of the actual height.
2. The HoloLens **should** be able to make a measurement of a pallet's height that is on average in a range of 3 centimeters of the actual height.

Usability It shall be ensured that...

1. the way to work with the HoloLens is easy to learn and navigate,
2. buttons, headings, and help/error messages are simple to understand.
   This must be verified with end-users testing the application.

Chapter 5

Product Backlog

This chapter gives an overview of the product backlog.

Use case 1

- Scan pallet barcode and display info
- LOGwear simulation database entry for pallet ID, height and location
- Create ground plane based on spatial mapping. HIGH PRIO
- Display selection indicator for highest point on "air-tap" gesture. HIGH PRIO
- Calculate and display distance between highest point and ground plane. HIGH PRIO
- API Request to DB for registering pallet height and ID

Use case 2

- Algorithm for determining best fit location(s) based on pallet height. HIGH PRIO
- Translation from theoretical locations to physical locations to display best fit locations using holograms. HIGH PRIO
- Directional indicator to best fit locations(s)

Use case 3

- Scan location barcode to register pallet location
- Confirmation to register pallet location
- API Request to DB for registering pallet location

Appendix H

High Racking Layout

Figure H.1: WH12 High Racking in Venlo, Location Width

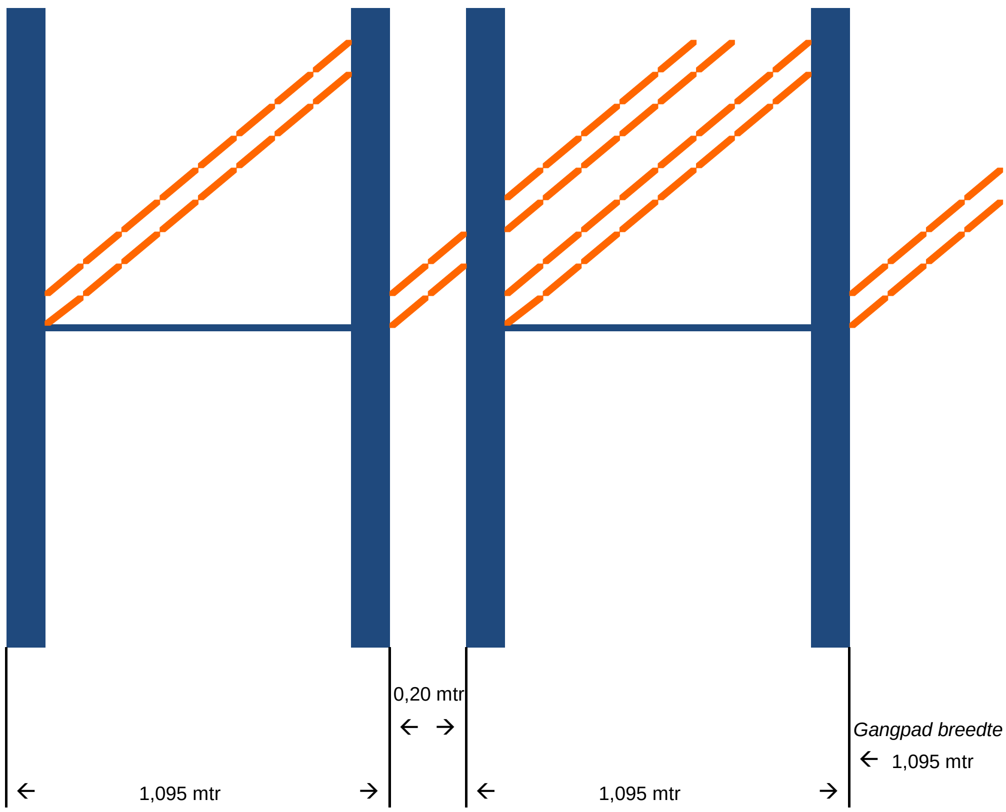


Table H.1: WH12 High Racking in Venlo on 2-3-2018

Nivo (Hoogte in cm)

Rij	Rij_L	1	2	3	4	5	6	7	8	9	#Lvl	Tot H
1	1L	194	194	194	194	234					5	1,058
	1R	194	194	194	194	234					5	1,058
2	2L	194	194	194	194	234					5	1,058
	2R	164	164	164	164	164	179				6	1,059
3	3L	164	164	164	164	164	179				6	1,059
	3R	164	164	164	164	164	179				6	1,059
4	4L	194	194	194	194	234					5	1,058
	4R	194	194	194	194	234					5	1,058
5	5L	194	194	194	194	234					5	1,058
	5R	209	239	179	174	209					5	1,058
6	6L	209	239	179	174	209					5	1,058
	6R	169	134	134	134	134	164	119			7	1,060
7	7L	209	239	179	204	189					5	1,068
	7R	209	239	179	174	219					5	1,068
8	8L	124	124	109	109	99	99	99	99	99	9	1,057
	8R	209	239	179	204	174					5	1,053
9	9L	209	184	184	204	224					5	1,053
	9R	189	194	194	214	219					5	1,058
10	10L	209	239	179	204	179					5	1,058
	10R	209	239	179	204	179					5	1,058
11	11L	169	134	134	134	134	134	149			7	1,060
	11R	179	184	179	174	134	144				6	1,054
12	12L	124	129	124	124	124	124	124	99		8	1,056
	12R	124	129	124	124	124	124	124	99		8	1,056
13	13L	124	129	124	124	124	124	124	99		8	1,056
	13R	149	139	139	139	139	139	139			7	1,055
14	14L	149	139	139	139	139	139	139			7	1,055
	14R	149	139	139	139	139	139	139			7	1,055
15	15L	119	124	124	124	124	139	124	99		8	1,061
	15R	134	124	124	124	124	139	124	99		8	1,076
16	16L	134	124	124	124	124	124	124	110		8	1,072
	16R	134	124	124	124	124	124	124	110		8	1,072
17	17L	224	224	224	149	184					5	1,053
	17R	149	139	139	139	139	144	139			7	1,060
18	18L	149	139	139	139	139	144	139			7	1,060
	18R	149	139	139	139	139	144	139			7	1,060
19	19L	149	139	139	139	139	144	139			7	1,060
	19R	149	139	139	139	139	144	139			7	1,060
20	20L	209	239	279	294						4	1,057
	20R	189	124	124	124	124	124	176			7	1,057
		40	40	40	40	39	23	19	8	1	250	
		2280	2280	2280	2280	2223	1311	1083	456	57	14,250	

Table H.2: Configuration File Excel Template

1L	183	194	194	194	194	234				
1R		194	194	194	194	234				
2L	185.5	194	194	194	194	234				
2R		164	164	164	164	164	179			
3L	184	164	164	164	164	164	179			
3R		164	164	164	164	164	179			
4L	183	194	194	194	194	234				
4R		194	194	194	194	234				
5L	183	194	194	194	194	234				
5R		209	239	179	174	209				
6L	185.5	209	239	179	174	209				
6R		169	134	134	134	134	164	119		
7L	182	209	239	179	204	189				
7R		209	239	179	174	219				
8L	186.5	124	124	109	109	99	99	99	99	99
8R		209	239	179	204	174				
9L	184	209	184	184	204	224				
9R		189	194	194	214	219				
10L	184	209	239	179	204	179				
10R		209	239	179	204	179				
11L	185	169	134	134	134	134	134	149		
11R		179	184	179	174	134	144			
12L	184	124	129	124	124	124	124	124	99	
12R		124	129	124	124	124	124	124	99	
13L	186	124	129	124	124	124	124	124	99	
13R		149	139	139	139	139	139	139		
14L	184	149	139	139	139	139	139	139		
14R		149	139	139	139	139	139	139		
15L	185	119	124	124	124	124	139	124	99	
15R		134	124	124	124	124	139	124	99	
16L	183.5	134	124	124	124	124	124	124	110	
16R		134	124	124	124	124	124	124	110	
17L	183.5	224	224	224	149	184				
17R		149	139	139	139	139	144	139		
18L	184.5	149	139	139	139	139	144	139		
18R		149	139	139	139	139	144	139		
19L	184	149	139	139	139	139	144	139		
19R		149	139	139	139	139	144	139		
20L	185.5	209	239	279	294					
20R		189	124	124	124	124	124	176		

Table H.3: CSV Configuration File

```
1L;183.0;194;194;194;194;234;;;;
1R;;194;194;194;194;234;;;;
2L;185.5;194;194;194;194;234;;;;
2R;;164;164;164;164;164;179;;;
3L;184.0;164;164;164;164;164;179;;;
3R;;164;164;164;164;164;179;;;
4L;183.0;194;194;194;194;234;;;;
4R;;194;194;194;194;234;;;;
5L;183.0;194;194;194;194;234;;;;
5R;;209;239;179;174;209;;;;
6L;185.5;209;239;179;174;209;;;;
6R;;169;134;134;134;134;164;119;;
7L;182.0;209;239;179;204;189;;;;
7R;;209;239;179;174;219;;;;
8L;186.5;124;124;109;109;99;99;99;99;99
8R;;209;239;179;204;174;;;;
9L;184.0;209;184;184;204;224;;;;
9R;;189;194;194;214;219;;;;
10L;184.0;209;239;179;204;179;;;;
10R;;209;239;179;204;179;;;;
11L;185.0;169;134;134;134;134;134;149;;
11R;;179;184;179;174;134;144;;;
12L;184.0;124;129;124;124;124;124;124;99;
12R;;124;129;124;124;124;124;124;99;
13L;186.0;124;129;124;124;124;124;124;99;
13R;;149;139;139;139;139;139;139;;
14L;184.0;149;139;139;139;139;139;139;;
14R;;149;139;139;139;139;139;139;;
15L;185.0;119;124;124;124;124;139;124;99;
15R;;134;124;124;124;124;139;124;99;
16L;183.5;134;124;124;124;124;124;124;110;
16R;;134;124;124;124;124;124;124;110;
17L;183.5;224;224;224;149;184;;;;
17R;;149;139;139;139;139;144;139;;
18L;184.5;149;139;139;139;139;144;139;;
18R;;149;139;139;139;139;144;139;;
19L;184.0;149;139;139;139;139;144;139;;
19R;;149;139;139;139;139;144;139;;
20L;185.5;209;239;279;294;;;;;
20R;;189;124;124;124;124;124;176;;
```

Appendix I

Design

Figure I.1: ER Model KLG Wearable Simulation System Database Regan [2018]

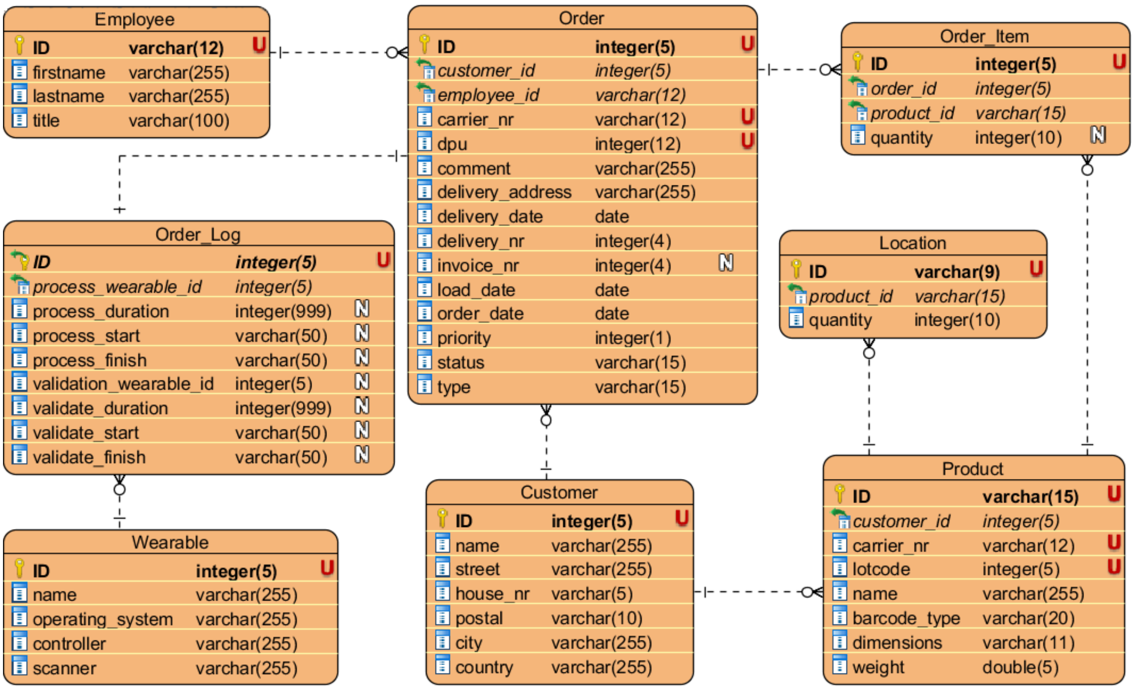


Figure I.2: Sequence Diagram "Operator Registers a Pallet for Storage" (UC-1)

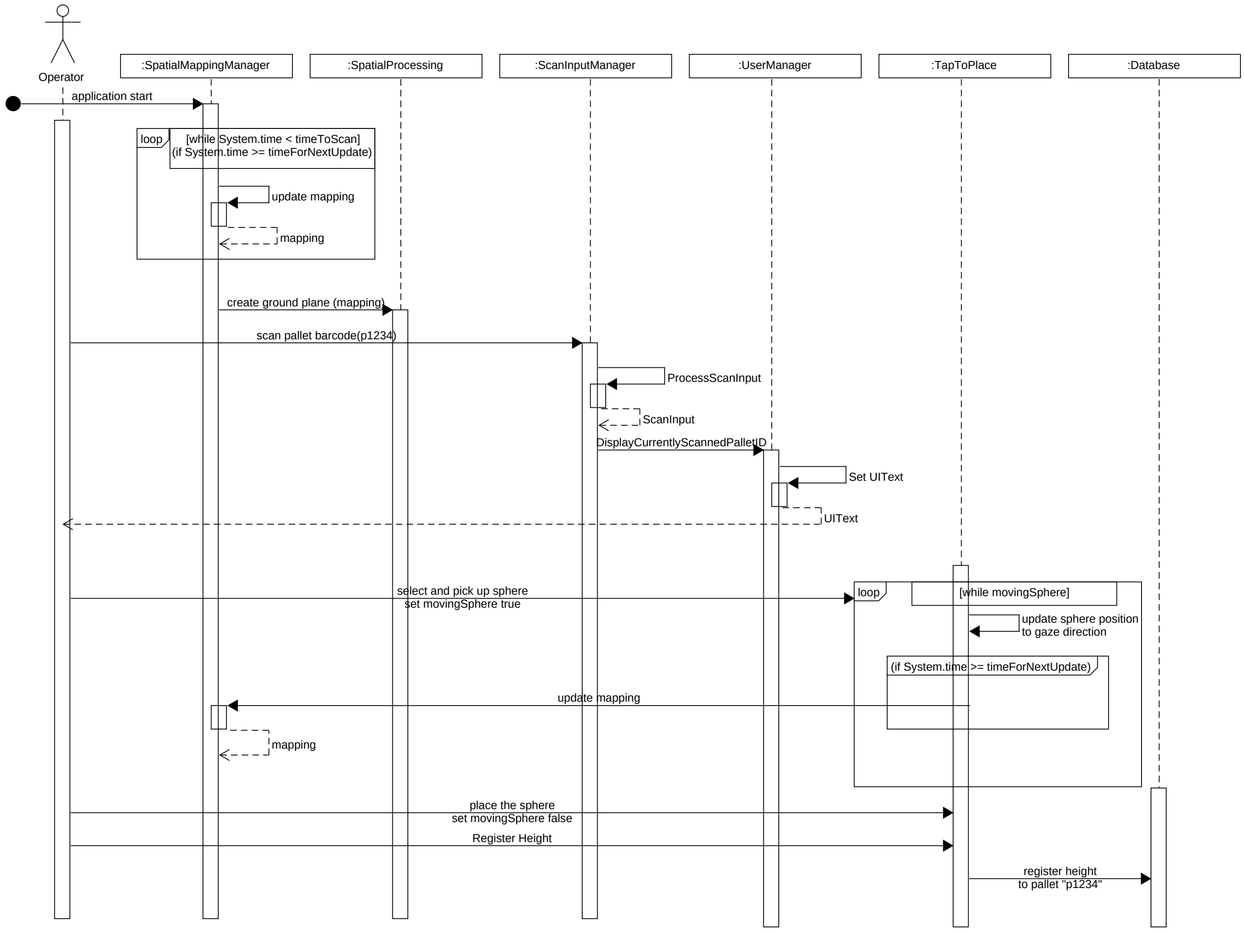


Figure I.3: Sequence Diagram "Operator Registers a Location to a Pallet" (UC-3)

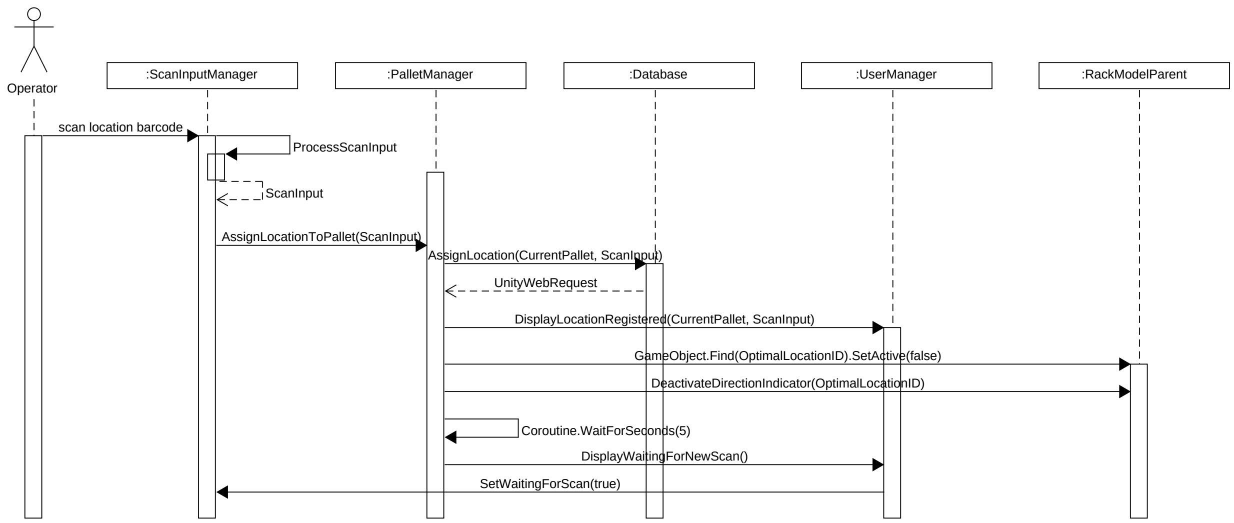
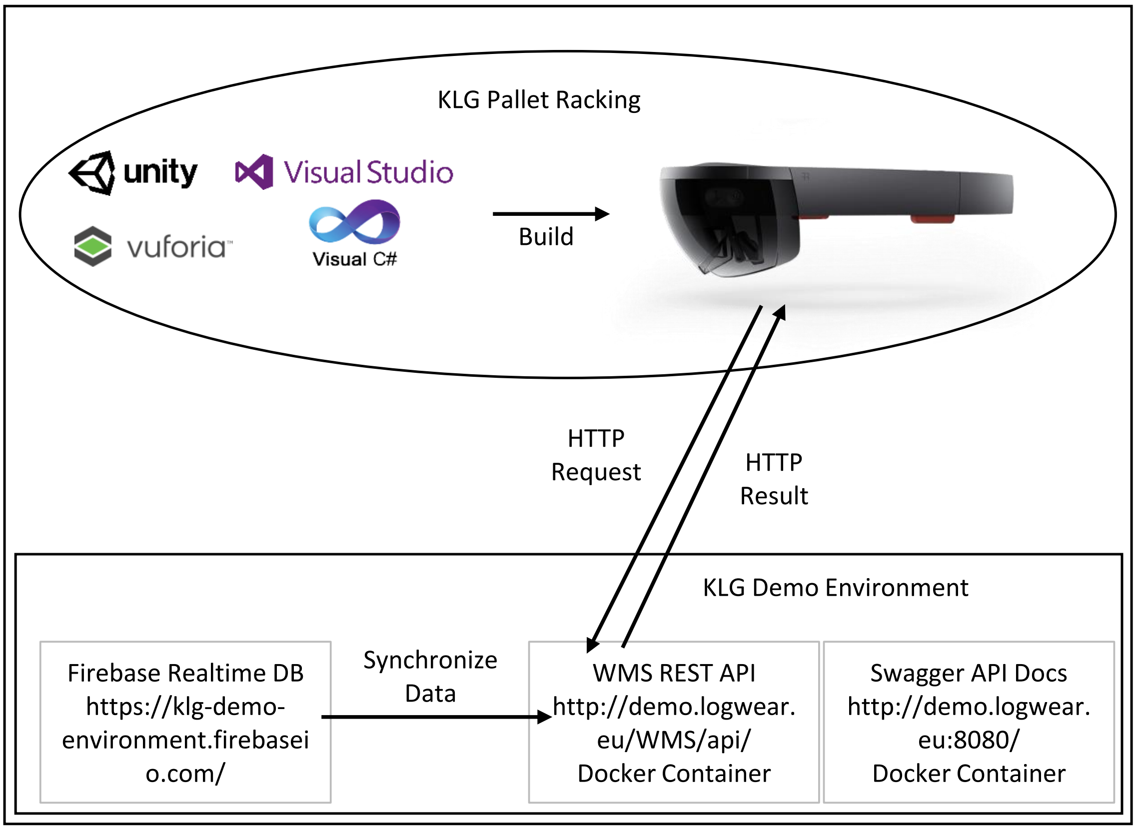


Figure I.4: Architecture Diagram



Appendix J

Power Consumption

Figure J.1: System Power Utilization Idle

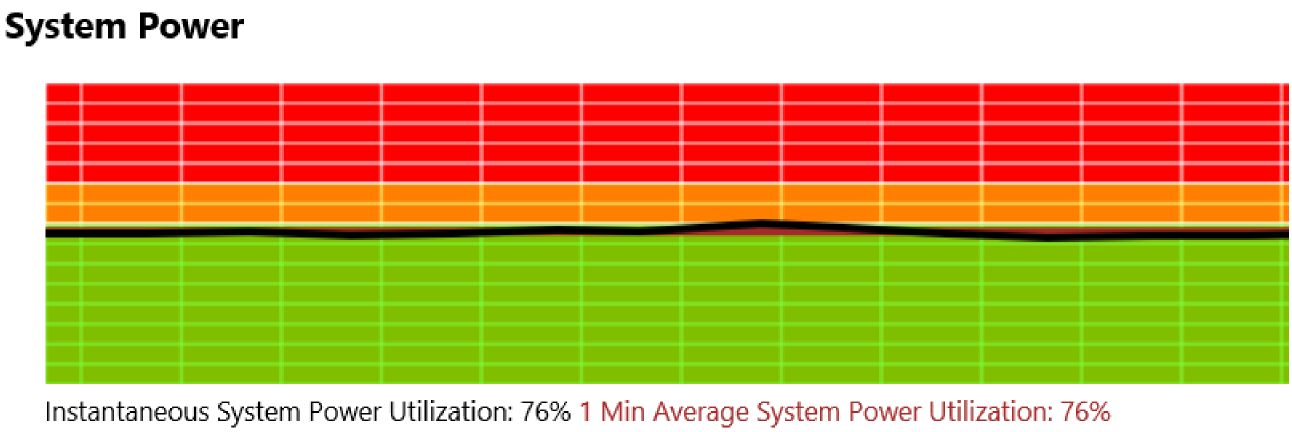


Figure J.2: System Power Utilization Spatial Mapping

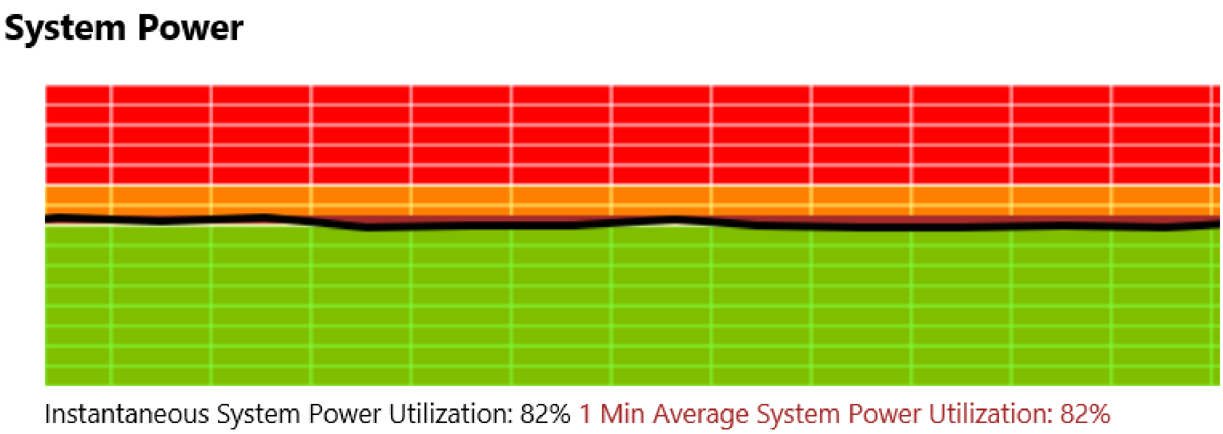


Figure J.3: SoC Power Utilization Idle

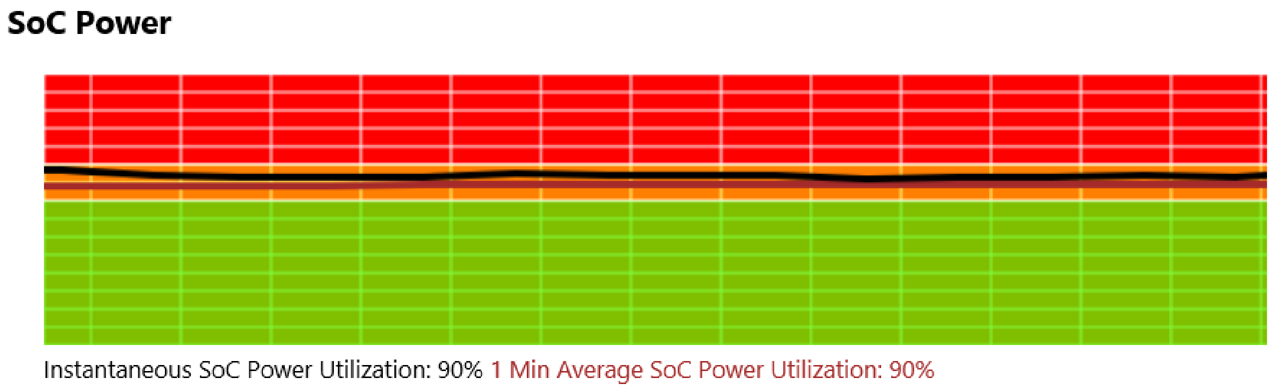
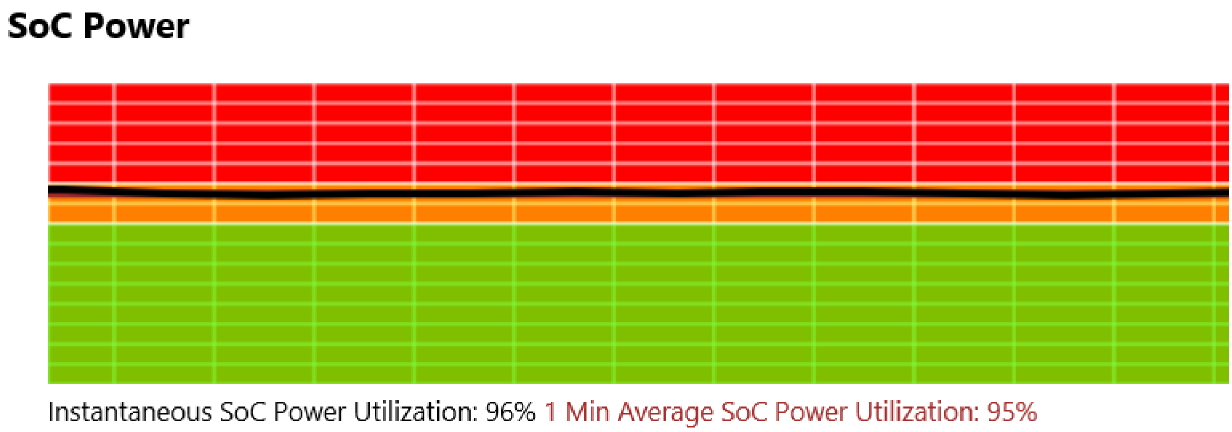


Figure J.4: SoC Power Utilization Spatial Mapping



Appendix K

Test Document

Test Document

Niels Killaars
Fontys Venlo Techniek en Logistiek
Bachelor Thesis

Venlo, May 31, 2018

1. Introduction

This document defines which features will be tested or will not be tested, which techniques/methods will be used for testing the features. The testing approach includes user acceptance testing, functional testing and performance testing. This document also contains the test results.

2. Test Execution Process

The test approach process describes how the application functional tests will be set up, executed and documented. The test approach is modeled with following figure.

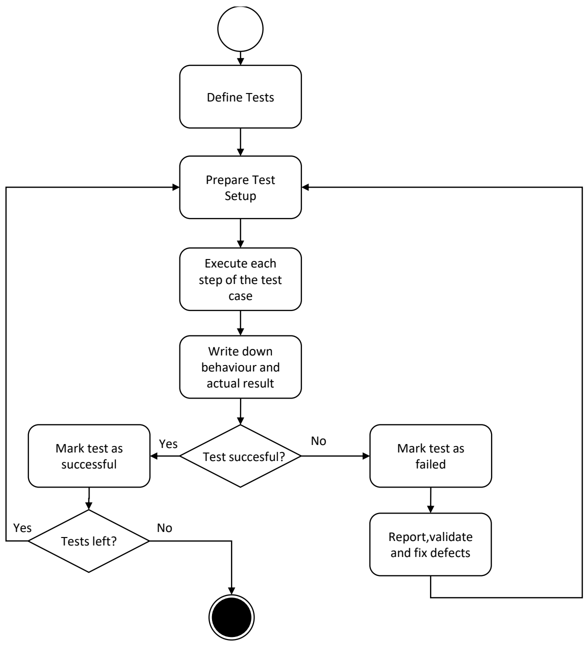


Figure 2.1: Test approach process model

First, the tests are defined. Every test definition contains following parameters:

- **Test Case ID** A unique number given to test case in order to be identified.
- **Description** The description of test case to be tested.
- **Test Setup** Anything that needs to be set up outside of the application for example printers, network and so on.
- **Expected Result** The description of what the function is expected to do.
- **Actual Result** pass / failed If pass - What actually happens when the test is run. If failed - put in the description what has been observed.

After the tests are defined, the test setup according the test definition is prepared and the test steps are executed.

3. Functional Tests

This chapter describes each feature, the corresponding function in the application, what the test prerequisites are and the expected results.

Test case ID	#1
Description	QR code Image recognition for identifying the exact location to place the virtual pallet rack
Test Setup / Prerequisites	QR code is placed on the pillar between row 8R and 9L exactly in the middle
Expected Result	QR is recognized, virtual pallet rack is placed exactly overlaying the physical model.
Result	Success

Table 3.1: Test QR Code initial model calibration

Test case ID	#2
Description	Barcode scanning to register a pallet
Test Setup / Prerequisites	Pallet that has received a barcode and is ready to be placed in the pallet rack & Test #1 successful
Expected Result	Pallet ID is recognized and entry is made in demo DB, application asks to measure the pallet's height
Result	Success

Table 3.2: Test Scan Pallet Barcode

Test case ID	#3
Description	The height of the pallet is measured and registered
Test Setup / Prerequisites	Test #2 successful, application asks to measure the height
Expected Result	Height can be measured using reference object and then registered in the demo DB to the pallet ID from Test #2
Result	Success

Table 3.3: Test Measure and Register Height of a Pallet

Test case ID	#4
Description	The optimal location for a pallet is requested by registering the pallet height
Test Setup / Prerequisites	Test #3 successful, application displays the measured height
Expected Result	Once the height is registerd, the optimal location is shown using holograms that mark the physical location, a UI text explaining the row and level of the location and a directional indicator shows the rough direction
Result	Success

Table 3.4: Test Determine and Display Optimal Location

Test case ID	#5
Description	The operator stores the pallet in a location and scans the location barcode
Test Setup / Prerequisites	Test #4 successful, operator has moved the pallet to a location in the rack.
Expected Result	When the location barcode is scanned, the application shows in the UI text that this location has been registered to the pallet ID scanned in test #2. The location is registered to the pallet ID in the demo DB.
Result	Success

Table 3.5: Test Register Location

4. Non-Functional Tests

Non-functional tests describe all tests that are not related to any specific function and have a great influence on the user's satisfaction. For this project performance/hardware tests, field tests and user experience tests have been executed.

4.1 Performance Tests

The performance tests are based the performance requirements found in the requirements specification document. The results are discussed in the performance testing section of the quality management chapter.

4.2 Field Tests

With the field tests the goal is to evaluate the adoption of product features. These tests gather data and feedback on natural, unguided use of the product over an extended period of time. In this case operators will have time to try out the application for half a day. The idea is to have a few operators try to do their normal routine when storing pallets, but in this case have access to extra guidance from the HoloLens. After the field test the operators will have to fill in a user experience and usability survey about the application. The results of the survey have been collected and are available below. The results are discussed in the quality management chapter of the final report.

High Racking using HoloLens

User Experience and Usability Survey

How easy is the application to use? *

	1	2	3	4	5	6	
Very easy to use	○	○	○	○	◉	○	Very difficult to use

If difficult to use, why?

I would find it easy to get the system to do what I want it to do *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	◉	○	Likely

My interaction with the system would be clear and understandable *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	◉	○	Likely

Using the system in my job would enable me to accomplish tasks more quickly *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

Using the system would enhance my effectiveness on the job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

Using the system would make it easier to do my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

I would find the system useful in my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

List the most negative aspect(s) of the application

List the most postive aspect(s) of the application

You can see the height very quickly and which location is fitting

Do you have any other remarks?

I was a forklift operator in the high racking warehouse

High Racking using HoloLens

User Experience and Usability Survey

How easy is the application to use? *

	1	2	3	4	5	6	
Very easy to use	○	○	◉	○	○	○	Very difficult to use

If difficult to use, why?

I would find it easy to get the system to do what I want it to do *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	◉	○	Likely

My interaction with the system would be clear and understandable *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	◉	○	Likely

Using the system in my job would enable me to accomplish tasks more quickly *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

Using the system would enhance my effectiveness on the job *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	◉	○	Likely

Using the system would make it easier to do my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

I would find the system useful in my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

List the most negative aspect(s) of the application

- Network connection is not always easy to make
- Option to scan 1D barcodes not available

List the most postive aspect(s) of the application

- You do not have to measure height physically
- The accuracy of: displaying while on the move, knowing it's own position in space

Do you have any other remarks?

I am a warehouse engineer at KLG

High Racking using HoloLens

User Experience and Usability Survey

How easy is the application to use? *

	1	2	3	4	5	6	
Very easy to use	○	○	◉	○	○	○	Very difficult to use

If difficult to use, why?

Can't find the pallet

I would find it easy to get the system to do what I want it to do *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	○	◉	Likely

My interaction with the system would be clear and understandable *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	○	◉	Likely

Using the system in my job would enable me to accomplish tasks more quickly *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	○	◉	Likely

Using the system would enhance my effectiveness on the job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

Using the system would make it easier to do my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	○	○	◉	Likely

I would find the system useful in my job *

	1	2	3	4	5	6	
Unlikely	○	○	○	◉	○	○	Likely

List the most negative aspect(s) of the application

Highlighting pallet did not work

List the most postive aspect(s) of the application

Operator can save time, a long time test regarding motion sickness is required

Do you have any other remarks?

LOGwear employee

4.3 User Experience

The following items have been tested to specifically test the user experience in different conditions. The criteria are received from the "Testing 3D in 3D" section of the official Microsoft development documentation, available using the following URL: `https://docs.microsoft.com/en-us/windows/mixed-reality/testing-your-app-on-hololens#testing-3d-in-3d`

- Is the application working throughout the complete warehouse? Yes, all rows are supported, the applications works very the complete pallet rack.
- Hologram placement
  - Are the virtually indicated optimal locations appearing at the right locations in the physical world? Yes, they are placed very accurately after the calibration. If the spatial awareness is lost, the locations may appear at the wrong position. This only happens when the view is obstructed by people or when standing too close to physical objects.
  - Are holograms placed at the expected location even at different angles? Yes, there is no difference at different angles.
- Light
  - Are the lighting conditions in the room influencing the user experience? Yes, dark locations in the warehouse cause the spatial mapping to be not complete. Dark object in dark areas can be not recognized at all.
  - Are the holograms bright enough / too bright? The holograms are bright enough and are not too bright.
- Sound
  - Are ambient noises influencing the user experience? No the radio that is playing in the background and sound of forklift trucks driving around does not disturb the user experience or usability of the application.
  - Are normal conversations possible while using the application? Yes, this does not affect the user experience or usability of the application other than being distracted.
  - Are the (feedback) sounds clear and loud enough or unclear and too loud? The feedback sounds are clear and loud enough.
- User Movement
  - How does the application respond to different head and body movements? Normal body and head movement does not affect the user experience or usability of the application. Rapid movements of the head may cause a loss of or difference in the spatial awareness and then causes holograms like optimal location indications to be rendered at a position that is offset from the physical location.
  - Is there a deviation between virtual movement and actual movement causing holograms to be displaced?
- Clipped Holograms

  - Are holograms properly fading getting very close to a hologram? The holograms do not fade when getting closer to them.
  - Is the distance that holograms start fading close enough or too far away? The holograms do not fade when getting closer to them.

- Is the quality of holograms high enough or too high causing performance issues? The quality of te holograms is already low, the performance instability is only felt during the calibration of the model. This is because the whole virtual pallet rack is rendered during this time.
- Is the cursor accurate enough to easily hover over the desired virtual objects? The cursor is very accurate and follows the head rotation very well.
- Are gestures accurate enough to easily select virtual objects using the cursor? Selecting objects using gestures is hard to get used to, but with a bit of experience gets very easy. Of course this is also different per person.

Appendix L

Research Results

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
1	Receiving packing slip	The transporter hands over the packing slip to the correct department / person for unloading the goods.	Personal computer or Wearable terminal with access to WMS required to determine empty pallet ID labels	Smart glasses can take over the job of a PC ,Wearable terminal or printed documents and provides direct access to the WMS to see available pallet IDs in the view.
2	Unloading pallets / boxes	The goods (pallets, boxes, etc) are unloaded from the truck.	None	Smart glasses can indicate to the person unloading the truck where to put the pallets / boxes based on empty spots.
3	Compare delivered quantity with packing slip	The user compares the goods that are unloaded with the packing slip to determine whether the delivered goods have been ordered and whether everything ordered has also been delivered.	PC or Wearable terminal to access the WMS and register if delivered goods are not equal to ordered goods.	Smart glasses can automatically detect and count the number of articles received using the barcodes and then compares it with the packing slip.
4	Signing goods for receiving	The user confirms the goods transfer in writing. The transporter receives a document as confirmation.	PC or Wearable terminal to access the WMS and print or send the document to the transporter.	Smart glasses can replace the need to access a computer to send the document, the instructions can be given via the smart glasses to print or send the document.
5	Move order	The user moves the goods to a warehouse for further processing.	PC or Wearable terminal with access to WMS to see where to move the goods in the warehouse	Smart glasses can directly display the information where to move the goods in the view of the operator.
6	Scan packing slip	The user scans the packing slip.	A hand scanner or ring scanner connected to the WMS	Smart glasses can take over the role of the scanner using the camera to detect and scan barcodes.
7	Unload pallet	The pallet is unloaded.	None	None

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
8	Inspect pallet or boxes for damage	The user inspects pallets or boxes for damage.	PC or Wearable terminal with access to WMS, required to register the damage.	A damage assesment can be done by smart glasses and automatically registered in the WMS. The user can also tell the smart glasses what damage is assessed, and the smart glasses then register this information in the WMS.
9	Sort pallets by product	The user sorts the pallets for products that must be stored separately.	None	Smart glasses can help sorting the products by visualizing how to sort the products.
10	Load pallet	The pallet is loaded.	None	Smart glasses can help the operator loading the pallet by visualizing the optimal positions to place the products on the pallet.
11	Apply pallet ID label	The user sticks the label on the pallet.	PC with printer to print the label	Smart glasses can visually indicate to the user the optimal location to stick the label on the pallet.
12	Enter label information	The user enters the relevant data for the label: customer ID, the number of articles and the batch code.	PC or Wearable terminal with access to WMS to enter the relevant data.	Smart glasses register voice input to enter the relevant data in the WMS.
13	Request storage location goods	The system calculates the storage location based on the data from the pallet.	PC or Wearable terminal to access the WMS and request storage location.	Smart glasses can retrieve the storage location from the WMS and visualize the storage location.
14	Transport pallet to storage location	The pallet is transported to the storage location by the user.	None	Smart glasses can guide the user to a location to store the pallets.
15	Scan location ID	The user scans the location ID. This means that the goods are added to the stock in the system.	A hand scanner or ring scanner connected to the WMS	Smart glasses can scan the location ID and register the location to the pallet in the WMS.

Table L.1: Inbound of Goods

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
1	Request order form	The order form is requested and displayed on a Wearable.	PC or Wearable terminal connected to the WMS to display the order form	Smart glasses can visualize the order form in the users view.
2	Start order processing	Accept the order and thereby adjust the status in the system.	PC or Wearable terminal to accept the order.	Smart glasses can provide the functionality to accept an order and adjust the status in the WMS.
3	Take transport equipment	Take transport equipment to pick the orders. (e.g. a pallet truck)	None	Smart glasses can indicate the optimal transport equipment and indicate the location of this equipment.
4	Scan order ID	The order ID is scanned	A hand scanner or ring scanner connected to the WMS	Smart glasses can scan the order ID.
5	Take empty pallet	Take an empty pallet using a pallet truck	None	Smart glasses can indicate the location of empty pallets and guide the operator.
6	Take container and scanning	Optionally, the container on the pallet is considered as a collection container for products.	PC or Wearable terminal to access the WMS and check if the container on the pallet is considered as a collection container.	Smart glasses can visualize if the pallet is considered a collection container.
7	Go to location	Move the pallet to the storage location.	None	Smart glasses can display the position of the location and guide the operator to this position.
8	Scan location ID	The employee scans the ID of the storage location.	A hand scanner or ring scanner connected to the WMS	Smart glasses can scan the location ID.
9	Pick up entire pallet	The entire pallet where the product(s) are on is picked up. (think of larger products)	None	None
10	Collect products (boxes)	Pick up a few boxes from the pallet and transport them with the pallet truck.	None	Smart glasses can indicate which boxes to pick up from the pallet.
11	Confirm product quantity (manually)	Manually count and enter the picked quantity.	PC or Wearable terminal to register the picked quantity	Smart glasses can recognize the number of articles picked and register the amount picked in the system.
12	Confirm product quantity (automatically)	The system automatically checks the picked items using the volume and weight.	Device that automatically checks the picked items using the volume and weight.	None
13	Transport to regroup location	Transport the picked items to the regroup location.	None	Smart glasses can visualize and guide the operator to the regroup location.

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
14	Regroup products	Sorting the picked items by product categories and packing requirements.	None	Smart glasses can help sorting the picked items correctly by recognizing the picked items. Smart glasses can display the packing requirements to keep in mind.
15	Add customer wishes	Pack the items as the customer wishes.	None	Smart glasses can display the customer wishes in the view of the user.
16	Check quantities	The picked quantities are checked / counted again.	None	Smart glasses can validate the picked quantities by recognizing the products.
17	Add packing slip / invoice	Create a packing slip / invoice and add it to the order.	PC with printer to print the packing slip / invoice.	Smart glasses can take over the job of the PC by communicating with the printer to print the packing slip / invoice.
18	Add shipping information	Create a label and confirm it with the order.	PC with printer to create and print the label.	Smart glasses can create the label based on user input and communicate with a printer to instruct the printing of the label.
19	Collecting pallets	Merge items that belong to the same order.	PC or Wearable terminal to see the items that need to be merged.	Smart glasses can display the items that belong to the same order and need to be merged.
20	Report an order ready	The order is reported 'ready for shipping' in the system.	PC or Wearable terminal with access to the WMS to report an order ready for shipping	Smart glasses can report an order ready for shipping in the WMS.

Table L.2: Order Picking B2B

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
1	Start order	Start a new order in the system.	PC or Wearable terminal with access to the WMS to start a new order.	Smart glasses can start a new order in the WMS on request of the operator.
2	Determine products to be resupplied (automatically)	The system automatically calculates the number of products to be resupplied.	PC or Wearable terminal to display the number of products to be resupplied.	Smart glasses can display the number of products to be resupplied in the view of the user.
3	Go to stock location	The employee moves to the storage location of the product that needs to be resupplied.	None	Smart glasses can visually guide the user to the storage location.
4	Scan product ID	The product ID of the goods is scanned.	Hand scanner or ring scanner connected to the WMS	Smart glasses can scan the product ID.
5	Compare inventory with physical inventory (Manual)	The system compares the available quantity of the product that is present at that moment with the quantity in the system. (e.g. by counting the number of products)	PC or Wearable terminal with access to the WMS to enter the number of products that have been counted.	Smart glasses can recognize the number of products and automatically compare the inventory with the inventory in the system.
6	Compare inventory with physical inventory (Automatically)	The available quantity according to the system is adjusted to the current available quantity. (automatically: for example, by calculating the volume or weight)	Device to calculate the volume or weight and determine the amount of products in inventory.	None
7	Correct stock for ripening	The refining function of the warehouse serves to refine certain goods or to increase the quality so that the product can be sold. Typical examples of refinement are the storage of wines and spirits such as whiskey, which only this way develop the desired flavour. Bananas usually also have to mature in storage, because they are still green and unripe during transport.	None	Smart glasses can indicate which stock needs to be corrected for ripening.
8	Correct stock for shelf life	Certain products have a limited shelf life, this shelf life is compared with the use-by date.	None	Smart glasses can recognize the product and use-by date to display to the user.

#	Activity	Description	HW/SW Requirements	Smart Glasses Use Cases
9	Calculate order size automatically	The system calculates automatically, on the basis of statistical data, the quantity of goods to be ordered.	WMS to calculate the quantity of goods to be ordered.	None
10	Calculate the maximum order size	Then the maximum order size per product is calculated.	WMS to calculate the maximum order size per product.	None
11	Request system inventory level	Inventory level according to the system.	PC or Wearable terminal to access the WMS and request the system inventory level	Smart glasses can access the WMS and display the system inventory level to the user.
12	Adjust system inventory level	If necessary, adjust the stock level.	PC or Wearable terminal with access to the WMS to adjust the system inventory level	Smart glasses can access the WMS and adjust the system inventory level.
13	Request maximum stock	Request the maximum stock level.	PC or Wearable terminal to access the WMS and request the maximum stock level.	Smart glasses can access the WMS and display the maximum stock level to the user.
14	Adjust maximum stock level	Adjust the maximum inventory level.	PC or Wearable terminal with access to the WMS to adjust the maximum stock level.	Smart glasses can access the WMS to adjust the maximum stock level based on user input.
15	Enter order quantity	An employee specifies the final order quantity. This can be a confirmation of the order quantity according to the system, but also a manual entry of the employee.	PC or Wearable terminal with access to the WMS to enter the order quantity	Smart glasses can enter the order quantity in the WMS based on user input.
16	Determine delivery date	Determine the delivery date of the ordered products.	The WMS determines and displays the delivery date on a PC or Wearable terminal	Smart glasses can display the delivery date when it has access to the WMS
17	Place the order	The order(s) are split per supplier and are placed via the system of the supplier.	PC or Wearable terminal with access to the WMS to place the order	Smart glasses can place the order on user input when it has access to the WMS

Table L.3: Restocking