Using AR gaze-based interaction to control IoT devices

Alexander Arvebratt, Fredrik Magnusson

DEPARTMENT OF DESIGN SCIENCES | FACULTY OF ENGINEERING LTH
LUND UNIVERSITY 2021

MASTER'S THESIS

jayway.
by devoteam

Using AR gaze-based interaction to control IoT devices

Alexander Arvebratt
al3818ar-s@student.lu.se

Fredrik Magnusson
fr2231ma-s@student.lu.se

June 24, 2021

Master's thesis work carried out at the Department of Design Science,
Faculty of Engineering LTH, Lund University.

Supervisors: Dr. Günter Alce, gunter.alce@design.lth.se
Pär Sikö, par.siko@jayway.com

Examiner: Dr. Mattias Wallergård, mattias.wallergard@design.lth.se

Using AR gaze-based interaction to control IoT devices

Published by

Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O. Box 118, SE-221 00 Lund, Sweden

Subject: Interaction Design (MAMM01)
Division: Ergonomics and Aerosol Technology
Supervisor: Günter Alce
Examiner: Mattias Wallergård

Abstract

Augmented Reality (AR) has lately gained a lot of momentum with popular applications such as Pokémon Go and Snapchat. Also, AR is becoming increasingly popular as a supportive tool for measuring objects and instructing users who are struggling with complex tasks. Moreover, head-worn AR glasses can have the potential of assisting people who needs their hands elsewhere or are unable to use their hands at all. By pairing AR glasses with eye-tracking, the technology can become even more accessible since it is possible to adopt gaze-based interaction.

This master thesis aims to explore whether you can control Internet of Things devices by solely using eye-gaze as input. A prototype using the AR glasses Magic Leap 1 have been created and tested against the native smartphone applications for Sonos speakers and Philips Hue lamps. The study was conducted with 20 test participants and then evaluated and scored based on different usability and workload factors. The results show that gaze-based interaction is viable and usable, but that today's hardware is too cumbersome to use as an everyday equipment.

Keywords: Augmented Reality, Eye-tracking, Gaze-based interaction, Internet of Things, Interaction design

Sammanfattning

Förstärkt verklighet (AR), augmented reality på engelska, har på sistone fått stor uppmärksamhet från kända applikationer som Pokémon Go och Snapchat. AR har dessutom blivit alltmer populärt som dels mätverktyg men också för att instruera och hjälpa användare som jobbar med komplexa arbetsuppgifter. Huvud-burna AR-glasögon kan dessutom ha stor potential att assistera människor vars händer är upptagna eller personer som saknar förmågan att använda dem. Genom att kombinera AR-glasögon med ögonspårning kan teknologin bli ännu mer användbar eftersom det då är möjligt att använda ögonbaserad interaktion.

Det här examensarbetet syftar till att utforska huruvida det är möjligt att kontrollera uppkopplade enheter, Internet of Things på engelska, genom att enbart använda ögonen som interaktionssätt. En prototyp som använder sig av AR-glasögonen Magic Leap 1 har utvecklats och testats mot de traditionella mobilapplikationerna för Sonos-högtalare och Philips Hue-lampor. Studien testades på 20 personer där faktorer som användbarhet och arbetsbelastning betygsattes. Resultatet visade att ögonbaserad interaktion är hållbar och användbar, men att dagens hårdvara ännu är för besvärlig att använda sig av för att fungera som ett vardagligt verktyg.

Nyckelord: Förstärkt verklighet, Ögonspårning, Ögonbaserad interaktion, Sakernas internet, Interaktionsdesign

Acknowledgements

Firstly we would like to thank Jayway by Devoteam for providing us with the equipment used for this master thesis as well as allowing us to use their office. A special thanks to Pär Sikö for supporting us during the project and to all the employees and fellow master thesis students who helped us during the development.

We would also like to thank our supervisor Günter Alce for continuous feedback during the project and for showing a great interest in creating a satisfactory result. He has greatly helped us understand the overall process of working with a master thesis and have been supportive from start to finish.

Lastly, we want to thank everyone who participated in the user study. It provided us with a lot of insightful feedback and data.

Contents

Chapter 1

Introduction

The first chapter of this master thesis presents the subject and motivates why it is needed. The goals, research questions along with the scope and limitations are also given.

1.1 Background

Augmented Reality (AR) is an interactive enhancement to the real world using computer generated information. AR can act through multiple sensory modalities. In other words, AR as a technology supports multimodal interaction including visual, auditory, haptic, somatosensory and olfactory [1]. Visual, auditory and haptic interaction is what is most commonly thought of when discussing AR.

AR as a technology is not a new concept. The first example of it was developed in 1968 by Ivan Sutherland [2]. Since then, the development of AR technology have seen an upswing and is starting to make its way into peoples everyday life. *Head-worn devices* (HWD) such as Magic Leap 1 [1], Microsoft Hololens 2 [2] and various AR-based smartphone applications have made it possible for everyday people to experience the technology hands on. AR is seemingly going to have a big influence in a future society with benefits in fields such as physiology, education, commerce, communications to name a few [3].

Pairing AR glasses with eye-tracking as input will open up the technology to a larger population and have a wider use case. For example, not having to make a call using hands can be appreciated when carrying a baby, or being able to look at instructions seamlessly while assembling an IKEA furniture. Furthermore, by allowing people to interact with technology in more ways will reduce inequalities for people with physical disabilities such as neuromuscular diseases. Having the possibility to play music, turn on lights and write messages to

[1] https://www.magicleap.com/en-us/magic-leap-1
[2] https://www.microsoft.com/en-us/hololens

friends should, according to the authors, not be a privilege, but a right.

AR as a technology is maturing, making it better and cheaper to produce. But in order to get a larger population to use it, user-focused applications needs to be developed. By focusing on *User-Centered Design* (UCD) and *User Experience* (UX), this master thesis will try to see whether it is possible to navigate and perform various actions in a *User Interface* (UI) with only eye-gaze as input. If done correctly, controlling home electronics with gaze-based interaction can have a great potential of making the everyday life easier for a lot of people. Creating such a system with a UCD-focus will make that future one step closer.

1.2 Goals

The goal with this master thesis is to explore and implement gaze-based interaction with *Internet of Things* (IoT) devices using the AR glasses Magic Leap 1. The IoT devices that the implemented system will be tested on are Sonos speakers and Philips Hue lights. To achieve this goal, the implemented system should:

- Design the UI with a focus on UCD
- Efficiently communicate with IoT devices
- Give reliable and consistent feedback to the user

The goals can be summarized into the following goal statement:

> *The system should be made as intuitive as possible and have a strong focus on UCD and universal design. It should communicate seamlessly with IoT devices and give the user continuous and helpful feedback in order to facilitate and improve the UX*

In order to evaluate how well the goals have been implemented, some research questions have been formed and these are the following:

- In which scenarios can a gaze-based AR system for controlling IoT devices be preferred over native smartphone applications?
- How can a system be developed with eye-gaze as the only input method?
- What is the state of today's AR technology in terms of hardware and available resources when using it as a potential everyday tool?

1.3 Scope and Limitations

The proposed solution involves a lot of interaction and design principles. However, some parts have to be left out due to time limitations. This master thesis focuses more on how a potential gaze-based interaction system could be implemented for everyday use. AR glasses are still a product designed to be used in the industry and not in everyday life. Therefore, the implemented solution is more of a proof-of-concept of what can be achieved using gaze-based interaction on IoT devices located at home. Hence, the following parts will be left out:

- The system will not be able to detect interactable devices. Instead, the coordinates of the devices will be hard-coded in order for the UI to be spawned in an accurate position. Therefore, if a device is moved, the UI will not adjust to the new location.
- When controlling the Sonos speakers, it would be desirable to be able to search up a specific song. This will however not be implemented since it is difficult to input text using only eye-gaze. A well-designed input system in an eye-gaze-only environment is a master thesis on its own.
- It will not be possible to completely customize the lighting when controlling Philips Hue. Instead, there are predefined colors or themes accessible to choose between.

1.4 Related Work

Gaze-based interaction has been of interest for a while, mainly in the medical industry. For instance, eye-gaze was used in a study performed by Araujo et. al. [4] to steer a wheelchair. It was meant to be used by people with neuromuscular diseases such as ALS, and in more severe cases, a so called locked-in syndrome (LIS) where they are limited to functional use of their brain and eyes only.

Communication by Gaze Interaction (COGAIN) is a research association who specializes in eye-gaze research aimed towards people with high-level motor disabilities [5]. They have developed standards and recommendations for systems where gaze is the means of communicating. A system based on the COGAIN guidelines is *DOGeye* developed by Bonino et. al [6]. The DOGeye system is designed to enable people with disabilities in controlling their homes using eye-gaze.

Another study made by Møllenbach et. al. examined different types of eye movements that can be utilized when interacting with interfaces [7]. They also studied whether so called *single strokes gaze gestures* (SSGG) is a viable navigation method. More recently, gaze-based interaction has been brought up as an option for controlling smart products at home. In a master thesis written by Lundqvist [8], the author tried to develop a system that could be used for controlling home environments using both AR and eye-tracking. While the same technologies will be used in this master thesis, Lundqvist used a different setup with a separate *head worn device* (HWD) and eye-tracker.

Tovesson investigated in his master thesis the potential of controlling IoT devices from a smartphone using AR technology [9]. Moreover, Tovesson used virtual objects as visual cues, or signifiers, in order to indicate which IoT devices that were interactable. This master thesis will further explore that method and see if it can be adopted in AR glasses as well.

1.5 The Global Goals of Agenda 2030

The Global Goals, more formally known as the *Sustainable Development Goals* (SDG) consist of 17 different objectives that the world should strive for and fulfill by the year 2030 in order to become a better place, no matter who you are or where you live. This master thesis can hopefully satisfy goal number 10: *Reduced Inequalities*, by enabling more people to interact with home devices, despite having any physical disabilities.

Figure 1.1: The 17 global goals for sustainable development.
Source: globalgoals.org

Chapter 2

Theoretical Background

This chapter briefly explains all the used theories and concepts.

2.1 Human-Computer Interaction

Human-Computer Interaction (HCI) is a term coined in the early 1980s. Back then, home computers were on the rise, and when Xerox introduced their brand new model *8010 Star* in 1981 a more user-centered design approach was introduced. HCI is a multidisciplinary field of research and spans across multiple areas such as computer science, HCD and interaction design. Hewett et. al. defines HCI as:

> *Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them* [10]

The *human* part in HCI refers to the actual user of a system and the computer can be for example a personal desktop computer, a smartphone, AR glasses, a microwave or a space shuttle. The *interaction* part is also very broad. Traditionally, the means of interacting with a computer consisted of a simple mouse and keyboard setup. Nowadays a user can utilize gestures, touch, speech and eye-gaze to name a few. In the book *Human Computer Interaction*, Dix et. al. categorizes interaction into two parts: *indirect* and *direct* [11]. Indirect interaction involves the use of a tool in order to perform a task. This includes interactions such as steering and clicking the cursor in a computer using a mouse, drawing images on a tablet with a stylus pen, and pointing and clicking with a laser pointer in an AR system. Meanwhile, direct interaction is when the users body is directly used to perform a task. This can include gestures, voice control and eye-gaze navigation. The two categories of interaction can also be combined. D. Norman criticizes some of the direct interaction methods such as gestures saying that they can decrease the usability since it is difficult for a user to know which actions that are possible and how they are invoked [12]. Moreover, Norman says that some of the *natural* interaction techniques often lack important aspects like visibility and feedback. However, other studies suggest that interaction using for example gestures and eye-gaze can greatly assist everyday tasks if implemented correctly. Graichen et. al. conducted a study where they compared touch-based and gesture-based interaction when controlling the infotainment system in cars [13]. They found that gesture-based interaction was helpful and also non-distractive compared to the touch-based alternative.

2.1.1 Visual perception

Gaze-based interaction is a form of direct interaction where eye-gaze is used for interacting with a system. Gaze-based interaction can be hugely beneficial as mentioned before. That being said does not mean that there are no problems with using eye-gaze as input. "Midas touch" problem is often discussed when using eye-gaze as input. It is essentially missclicks when using eyes as input, in other words it is when an action has been performed when the user did not intend to perform said action. This happens since users are not accustumed to opperating devices simply by moving their eyes[14]. Methods of interacting with interfaces are being developed and tested to avoid "Midas touch" problem while simuntaineosly maintainging a good UX.

2.2 Augmented Reality

Augmented Reality (AR) is a technology where virtual objects or elements are applied to the real world. This can be achieved with, for example, a smartphone camera or holographic lenses in a *head-worn device* (HWD). AR is not to be mistaken with *Virtual Reality* (VR) where the user is entirely encapsulated in a virtual world. One of the most common definitions of AR was written by Azuma in 1997 [15], and he defines AR with the three following characteristics:

(1) Combines real and virtual
(2) Is interactive in real-time
(3) Is registered in three dimensions

While AR can be found in areas such as manufacturing, education and medicine, most people can relate to it through mobile applications like Snapchat and Pokémon Go.

2.3 Eye-Tracking

Tracking a persons eye-gaze or head movement relative to the users eyes is the process of eye-tracking. Eye-tracking have been studied in a number of fields such as psychology, marketing and HCI. Scenarios where eye-tracking is used can be to display where a user looks when reading an article or as an input method in rehabilitative and assistive applications.

Figure 2.1: Usage of AR in the mobile game Pokémon Go
Source: niantic.com

There are essentially three ways of tracking ones eyes: measuring something attached to the eyes, typically a lens [16]; measuring electric potentials using electrodes placed around the eyes [17]; and lastly optically tracking the eye movements without direct contact with the eyes. The last mentioned technique is what is most common in modern eye-trackers and is also used in the projects eye-tracker.

2.4 Internet of Things

Internet of Things (IoT) is a fairly new paradigm that is gaining ground as more and more everyday devices are being connected to the internet. There is however no standardized definition of IoT. Atzori et. al. says:

> *The reason of today apparent fuzziness around this term is a consequence of the name "Internet of Things" itself, which syntactically is composed of two terms. The first one pushes towards a network oriented vision of IoT, while the second one moves the focus on generic "objects" to be integrated into a common framework [18].*

While the term IoT might be cryptic, it can be summarized in the context of this master thesis: as objects or devices connected to the internet and therefore being able to communicate with each other as well as being controlled. The devices are often called *smart* and are becoming increasingly more common in the home. Today it is possible to have for example smart fridges, smart lights and smart coffee makers. The list goes on and on, and as more and more items become *smart*, the term IoT gets increasingly popular and can be considered a de facto standard.

2.5 Idea generation

In this section, different methods of idea generation that has been used during the design process are briefly explained.

2.5.1 Bodystorming

Bodystorming is a tool for idea generation. The exercise situates brainstorming in a physical experience by role-playing and simulating a scenario. It can inspire the participants to explore new ideas and emphasizes spontaneous prototyping. A bodystorming session is conducted by placing a participant in a scenario using *low-fidelity* (Lo-Fi) props if needed. Props such as cardboard boxes, clear glasses or *Post-It* notes are sufficient. The participant is encouraged to explore and try things in the scene while thinking of various design solutions [19].

2.5.2 6-3-5 Brainwriting

6-3-5 brainwriting or the 635 method is a brainstorming technique designed to be performed by a group of six people to generate design ideas. The tools needed for a brainwriting session is a pen and sheet of paper for each participant. The session works like this[1]:

1. Each participant writes down three ideas regarding the domain during five minutes.
2. After the time is up, each participant slides the sheet of paper to the person left of them.
3. Each participant reads what the previous participants wrote on the paper.
4. Each participant writes down three new ideas on the newly received paper.
5. Each participant repeats step 2-4 until handed the original sheet of paper.

It has been documented that 6-3-5 brainwriting can be beneficial for idea generation, mainly since it allows group members to build on shared ideas [20].

2.5.3 Storyboarding

Storyboarding is explained in the book *Universal Methods of Design* as:

[1]https://podojo.com/how-to-6-3-5-brainwriting/

> *A visual narrative that generates empathy and communicates the context in which a technology or form factor will be used* [19].

The goals with storyboarding is to get into the mindset of the end-users and at an early stage come up with different design alternatives. Storyboards consist of multiple panels (interaction steps) similar to comic books and are often hand-drawn.

2.6 Usability Testing

Testing a systems usability is necessary to achieve a UCD. Usability testing should start early on in the development process and be an integral part throughout the project. The work process should be iterative with tests, (re-) evaluation and (re-) implementation. Typically a test consists of a facilitator, tasks, and a test participant. Depending on what wants to be achieved, different types of tests can be performed[2].

Evaluating a systems usability comes down to essentially five points; how intuitive the system is to use and navigate, how long and easy it is for a user to learn the system, how efficient the system is for an experienced user to use, how memorable the system is to visit again, and lastly how prone the system is to errors and the severity of said errors[3]. There are several methods used to evaluate these points and the ones used in the master thesis will be described in the following sections.

2.6.1 A/B Testing

A/B testing is used to compare two versions of the same system in order to find out which of the versions that yields the best result. It is a technique for optimising a system. The test is performed by having half of the population test version *A* of a system and the other half test version *B* until a sufficient sample size is gathered [19]. If the sample population is too small both groups can test both of the versions. If this is the case the versions should be tested in different order to negate the transfer-of-learning effect [21].

2.6.2 SUS

System Usability Scale (SUS) is a usability tool for measuring a systems usability according to a universal scale. The form consists of ten statements with five response options ranging from *strongly disagree* to *strongly agree*. The user should be handed the form directly after performing the test to ensure the best result [22]. The questionnaire is scored on a scale from 0-100 and can be given a letter grade F (being the worst) to A+ (being the best) using mean and standard deviation of other test results[4].

[2] https://www.nngroup.com/articles/usability-testing-101/
[3] https://www.usability.gov/what-and-why/usability-evaluation.html
[4] https://measuringu.com/sus/

2.6.3 NASA-TLX

NASA Task Load Index (NASA-TLX) is a tool developed by the Human Performance Group at NASA which measures the perceived workload a user feels when performing a task. The questionnaire consists of two parts. In the first part the workload of the task is measured based on six different subcategories which are: mental demand, physical demand, temporal demand, performance, effort and frustration. In the second part of the questionnaire, an individual weighting of these subscales are created based on their perceived importance, this to take individual differences into consideration. Weighing the two parts together creates a measurement on the perceived workload [23]. Studies suggest however that the second part is redundant and can be skipped if the only purpose is to get an overview of the workload [24]. A modified version called Raw-TLX can then be used instead which has been done in this master thesis.

Chapter 3
Technical Background

This chapter briefly explains all the used technologies and products.

3.1 Unity and Node.js

Unity is a popular game engine developed by Unity Game Technologies[1]. It works as an editor and contains a lot of libraries and frameworks designed for developing applications in AR such as *AR Foundation*[2]. Unity is compatible with Magic Leap 1 which are the AR glasses used in this master thesis.

Node.js is a back-end runtime environment that can execute JavaScript code outside of a web browser. It is an easy way to build and deploy a lightweight *Hypertext Transfer Protocol* (HTTP) server. Node also comes with a package manager called *Node Package Manager* (NPM). From NPM, a lot of useful packages can be downloaded and used with ease. In this master thesis, three different NPM packages are used: *Express.js*[3], *node-sonos*[4] and *node-hue-api*[5].

3.2 Sonos

Sonos Inc. is a company that develop and sell wireless home audio products and smart speakers[6]. The company is working towards multi-room audio, in other words playing music from

[1] https://unity.com/
[2] https://unity.com/unity/features/arfoundation
[3] https://expressjs.com/
[4] https://github.com/bencevans/node-sonos
[5] https://github.com/peter-murray/node-hue-api
[6] https://sonos.com/

different speakers simultaneously using WiFi. The company have an API designed for developers to implement their own products to interact with the system.

3.3 Philips Hue

Philips Hue is a smart light system produced by Philips[7]. Hue is essentially smart light bulbs which can be connected together to create a whole system of smart light bulbs. The product can also be integrated to other IoT devices such as smart TVs and alarm clocks. Philips Hue lights can be modified in many different ways and the API is available for developers to use.

3.4 Magic Leap 1

Magic Leap 1 (ML1) is a spatial computer developed and sold by the company Magic Leap Inc. Spatial computing is the interaction between computers and humans where the computer has knowledge of and manipulates things in real life such as objects and spaces [25]. The ML1 have a *head-mounted display* (HMD) with cameras and sensors and creates AR using superimposed computer generated 3D images over real life objects. Furthermore, ML1 have built-in speakers mixing real sounds with computer generated sound and can also track the users eye movements using an eye-tracker.

Figure 3.1: Magic Leap 1 displayed with corresponding lightpack, lightwear and controller.

Source: cnet.com

The system can be broken down into three parts: a computer, an HMD and a controller. The three different parts can be seen in Figure 3.1 above. The computer which is called *Lightpack* handles all the computations and is attached to the HMD called the *Lightwear*. The *Lightpack* clips on to the users pocket and runs on a battery, making it easy to carry when moving

[7]https://www.philips-hue.com/

around. The *Lightwear* have a holographic display that displays the superimposed 3D objects and simulates depth and field using the sensors as well as the cameras. Moreover, the display have 50ºof diagonal Field of View (FoV) with a refresh rate of 120Hz. Finally, the system also have a controller which can track in six degrees of freedom (position and orientation)[8].

3.4.1 The Lab and Zero Iteration

The Lab is the software included with ML1 and handles various properties such as libraries, device management and *Zero Iteration* (ZI). ZI is a tool from The Lab which makes it possible to test and run code on the ML1 without having to build and deploy it first [9]. This is an extremely useful functionality since it is also compatible with Unity. All a developer have to do is to save and run the scene in Unity in order to test it in the ML1. Furthermore, ZI works over WiFi making the functionality wireless.

[8]https://www.magicleap.com/en-us/magic-leap-1
[9]https://developer.magicleap.com/en-us/learn/guides/lab-zi

Chapter 4
Technical Development

This chapter describes the overall architecture and the different building blocks in the AR application, such as Unity, the server and the APIs.

4.1 Architectural Overview

The system as a whole consists of three main parts: the spatial computer (ML1), a server (Node.js) and IoT devices (Sonos speaker and Philips Hue lights). The systems architecture is quite complex since the ML1 needs to communicate with the different APIs for the Sonos speakers and Philips Hue lights. A full overview can be seen in Figure 4.1. First there is the ML1 which with the help of Unity renders the virtual elements near the real IoT devices. The eye-gaze of the user is constantly being monitored and if the gaze trail hits one of the virtual elements a call is directly sent from the ML1 to the Node.js server. The server then looks at the request and redirects it to one of the APIs by utilizing different end-points. When the request has been processed, a response is sent back to the server who in turn modifies the response and redirects it to the ML1 which updates the state.

For example, if a user wants to play a song on the Sonos speaker, the data flow would look something like:

1. The user looks at the *play* button.
2. A coroutine is started in the ML1 which sends a HTTP request to the Node.js server stating that the play button has been pressed.
3. The server processes the message and calls the Sonos API with a correct HTTP request.
4. The Sonos API responds with a HTTP status code, and if the request was successful the message: `200 OK` is returned and the music starts playing.

5. The server sends an additional HTTP request for the song information (i.e. song title, artist and album) to the Sonos API.
6. The Sonos API responds with the current information.
7. The server modifies the request and sends it back to the ML1.
8. The ML1 re-renders the new state onto the canvas which is visible through the AR glasses.

The data flow works the same for other buttons and when interacting with Philips Hue.

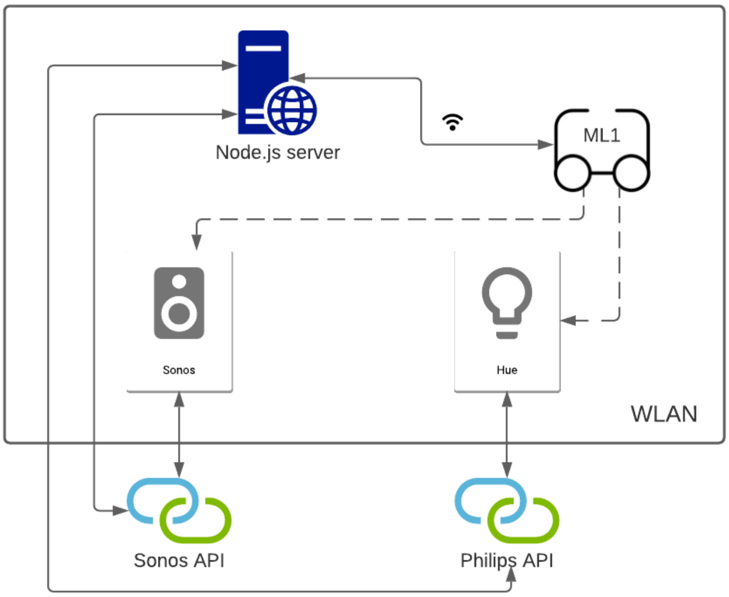

Figure 4.1: Architectural overview

4.2 Implementation

In this section a brief explanation will be given regarding how the virtual objects, the eye-tracking and the server were developed.

4.2.1 Unity

Unity can be regarded as a hub between the server and the ML1. All the virtual objects, for example the markers and canvases, are rendered through Unity. Meanwhile, with the help of scripts, a continuous communication can be maintained with the server.

The scene in Unity consists of only six different virtual objects visible to the user:

- A Sonos marker located over the speaker

- A Sonos media player UI
- Two Hue markers located above the respective light sources
- Two Hue light settings UIs

The two different types of menus spawns when the user interacts with either one of the two types of markers. All of the virtual objects have so called colliders[1]. This means that they have borders and surfaces similar to a real-world object. Having colliders is crucial when interacting since Unity can detect when something, in this case the gaze vector, collides with an object. Therefore each button in the two different menus have separate colliders in order to detect which button that is being pressed. Also, when a button reports a collision (i.e. it is being looked at), it changes color so that the user gets feedback that the desired button is actually being pressed.

Furthermore, the scene in Unity have several scripts working behind the scenes:

- Start scripts for both Sonos and Hue
- Server scripts

The start scripts handles all the eye-tracking functionality and reports which button that is being pressed. Moreover, the start scripts handles the different timers used for calculating the dwell times. Along with the start scripts, the server scripts receives information about what the user is doing. For example, if *play* is pressed in the Sonos menu, the server script gets notified and sends a request to the Node.js server with an appurtenant endpoint.

The eye-tracking functionality was included in an API provided by Magic Leap called MLEyes [2]. It both detects the user's fixation point and calculates the direction of where the user is looking. In order to integrate all the various functionalities, the API had to be imported and integrated into the Unity scene.

4.2.2 Node.js server

The Node.js server is completely separated from Unity but is constantly listening for requests on a specific port number. Other than that, the server works completely independently, meaning that the user never have to do anything to manipulate it.
The Node.js server consists of four different modules:

- A main server module
- A Sonos module
- Two separate Hue modules

[1] https://docs.unity3d.com/Manual/CollidersOverview.html
[2] https://developer.magicleap.com/en-us/learn/guides/eye-tracking-tutorial-unity

When a request is sent from Unity it first reaches the main server module which checks which endpoint is called. Depending on what the endpoint is, the request is forwarded to either the Sonos or one of the Hue modules. Then the associated action is performed since the modules handles the connection with the Sonos and Hue APIs.

The responses from the API calls in the Sonos and Hue modules are then sent back to the main server module which in turn modifies and redirects the response to Unity.

4.2.3 API

Different APIs had to be used in order to interact and manipulate the state of both the Sonos speaker and the Philips Hue lights. Luckily enough, much of the hassle of connecting the APIs was simplified with the help of two NPM packages for each system [3] [4]. The packages both automated tasks such as finding devices as well as facilitated various commands like playing and pausing the music.

[3] https://github.com/bencevans/node-sonos
[4] https://github.com/peter-murray/node-hue-api

Chapter 5

Design Process

This chapter describes the process of generating, testing and implementing different ideas regarding the design of the system.

5.1 Overview

The design process consisted of two different phases, the Lo-Fi phase and the High-Fidelity (Hi-Fi) phase. In both phases, the workflow was iterative, meaning that the design propositions have been tested, evaluated and then re-implemented one to several times per phase. Also, the technical implementations that had nothing to do with the design, for example the Node.js server, was implemented in parallel with the two phases. By doing so, all of the logic was in place when later on implementing the design in Unity. Furthermore, since most of the technologies were new to the authors it was crucial that enough time were to be allocated towards the more time-demanding process of implementing the server, understanding the software included in the ML1 and integrating all the different parts.

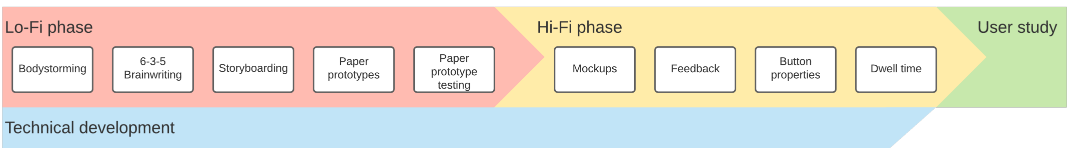

Figure 5.1: Design process timeline showing phases and their content.

5.2 Lo-Fi phase

This section will describe the process of the Lo-Fi development along with the methods and exercises used for encouraging novel ideas. Since the Lo-Fi phase is conducted at an early stage in the design process, much of the focus was put on overall design elements and means of interaction using eyes.

5.2.1 Bodystorming

To start off the idea generation process, it was decided that a bodystorming session would be conducted. Bodystorming as a tool was chosen since it is easy to perform at an early stage. The tool is also preferable since the intended systems use case is in mixed reality and the user is encouraged to walk around and interact with the real world.

The session took about 45 minutes and the two authors were the only participants. It was conducted using a small number of props consisting of a pair of sunglasses and *Post-It* notes with writings on. The sunglasses were used to get immersion of using real AR glasses, and the *Post-It* notes acted as placeholders for the IoT devices. During the session, one of the participants role-played a scenario where he came home from work and wanted to relax with some music and mood setting lighting. When performing actions, the participant used the think-aloud method, explaining what he imagined seeing when using the system. The other participant wrote down what was being said while simultaneously playing music and turning on and off lights accordingly. After one bodystorming session was completed, the authors changed roles so that both got the chance to perform fictive actions. A picture from the session can be seen in Figure 5.2.

Figure 5.2: Photograph from bodystorming session where the participant interacts with a fictive Sonos speaker while wearing sunglasses.

5.2.2 6-3-5 brainwriting

The previous brainstorming session generated some initial design proposals that acted as a good foundation. However, it did only include the authors. It was therefore decided that another idea generating session would be held together with two other master thesis students that were sitting at Jayway's Malmö office at the time. It was believed that the project would benefit from outside opinions and new ideas that might not have been brought up otherwise. 6-3-5 brainwriting was chosen since it is an easy way of generating plenty of new ideas in a not so time-consuming session.

The session started with a five minute introduction to the project and to the exercise. It was decided to keep the introduction short in order to not color the outside participants ideas. The exercise was held in Swedish, as it is the first and preferred language of all the participants. In the original 6-3-5 brainwriting method, six participants are required (the number six in *6-3-5*), but due to the ongoing pandemic resulting in many people working from home, it was decided that four people were sufficient in order to get a considerable amount of ideas. Having four participants yielded 48 unique ideas (six participants would have generated 108 ideas), which was deemed enough. The photograph in Figure 5.3 shows some of the yielded ideas from the session.

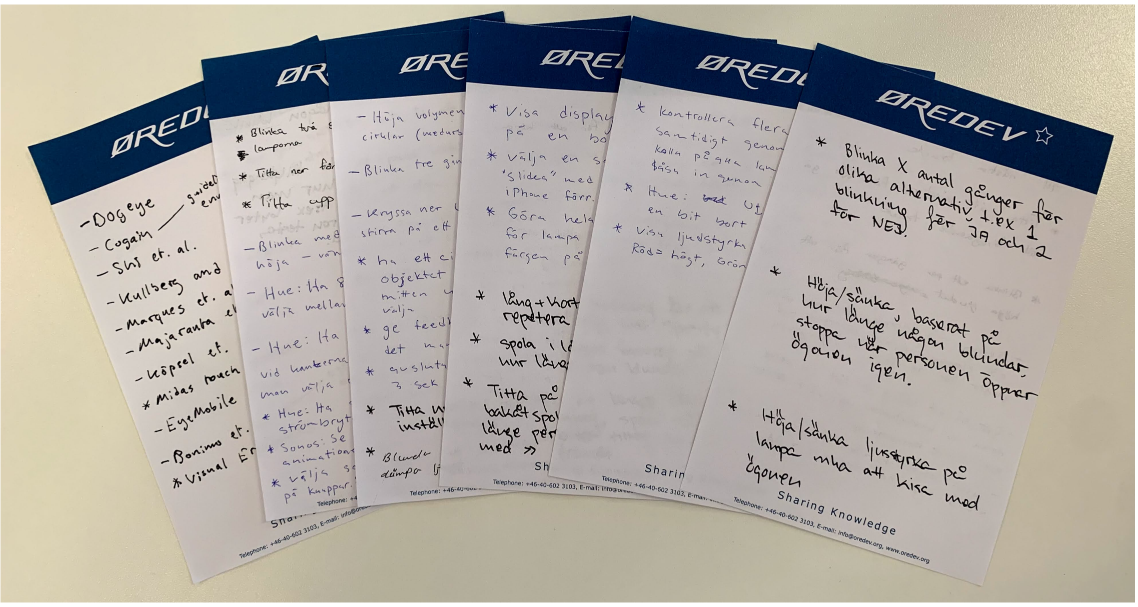

Figure 5.3: Some of the ideas that came up during the 6-3-5 brainwriting session.

5.2.3 Storyboarding

It was decided to sketch up two design proposals for the system, one for interacting with Sonos speakers and one for Philips Hue lights. Storyboarding as a tool emphasizes interaction since it *tells the story* of the interaction of a system, and since the project mainly focuses on interactability, storyboarding was a suitable choice.

The storyboards were first sketched up using pen and paper. This was done in order to agree on an initial idea. After discussing different design alternatives, a final idea was agreed upon and drawn using a Wacom One drawing pad[1] as well as with the program SketchBook from AutoDesk[2]. The computer-based drawing was also refurbished along the way as new ideas emerged. The final result can be seen in Figure 5.4.

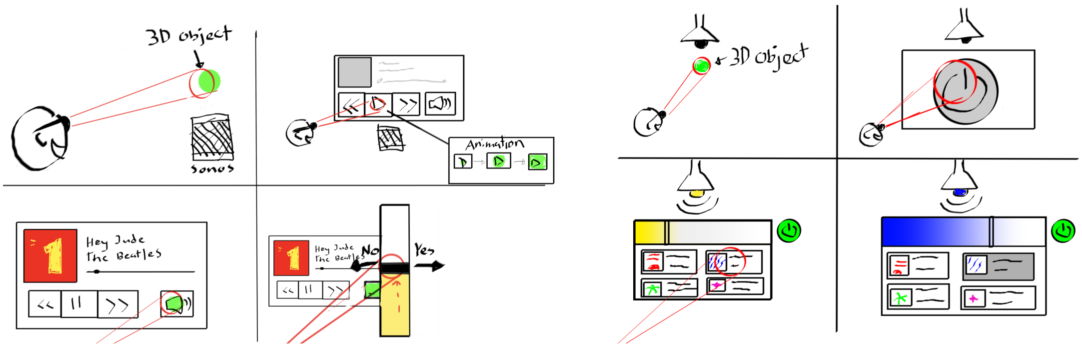

Figure 5.4: Storyboard showing interaction with Sonos speaker (left) and interaction with Philips Hue (right).

5.2.4 Paper prototypes

Since many idea generating exercises had been conducted, some final ideas on design proposals started to emerge. To get a better understanding how the ideas would function when being used by an actual end user, two paper prototypes were created using sheets of paper in different colors, scissors and pens. Both design proposals were made for the Sonos speaker and Philips Hue respectively in order to be able to try them out in two different contexts. This resulted in four paper prototypes - two per system.

The first design proposal which was called *square* is essentially a card-based design which is heavily based on smartphone and computer applications. By mapping the design to common software, the idea is to make the UI recognizable as well as easy to use. Also, by having a common design pattern, users might feel more confident when using the system.

A problem with the square-shaped design is that it takes up a lot of space in the AR glasses. Therefore, the other design proposal which was called *donut* is shaped like a torus (a circle with a hole in it). The buttons are positioned inside the donut similar to the design of a TV remote controller or a more traditional iPod wheel. This design was first thought of during the 6-3-5 brainwriting session and became later on a favorite among the authors. Being round, it takes up the least amount of space while still having large enough buttons. Another thought with this proposal was to have the user rest his/her eyes in the middle hole, making it easy to see all the choices while not accidentally pressing a button.

[1] https://www.wacom.com/en-se/products/one-by-wacom
[2] https://sketchbook.com/

Both design proposals can be seen in Figure 5.5. By having two different paper prototypes for the two intended systems, it was a perfect setting for performing A/B testing.

5.2.5 Paper prototype testing

Before going further and implementing one of the design proposals in Unity, it was decided to have an A/B testing session between the two paper prototypes. The goal of the testing was to find out which design that was preferred from a users point of view. The favorite among the test participants would later on be implemented in Unity.

The session was conducted using the two paper prototypes created earlier in the Lo-Fi phase. During a test there was firstly a facilitator (test leader) responsible for holding an introduction as well as giving instructions. Secondly, another person was responsible for moving the pieces in the prototypes according to what the test participant was doing. The authors rotated these roles after each test. Four people were tested and all of them were Jayway employees. More specifically, they were all females with ages ranging from 23 to 29. They all had an idea of what AR was but had never tested any AR glasses before. When an instruction was given by the facilitator, the test participants had to think-aloud, saying where they would look at while the paper pieces were moved and modified accordingly. Each test participant tried both the Sonos and Philips Hue systems but with different designs. The designs were then shifted for the next participant. For example, one person tried Sonos with the square design and Philips Hue with the donut design while the next tried Sonos with donut etc. After each test session, a short interview was held. The test took about 20 minutes. Figure 5.5 shows both prototypes in use. Only testing the paper prototypes on four people was a result of the ongoing pandemic that forced many of the otherwise available employees to work from home, thus limiting the possibilities of gathering more participants on such a short notice.

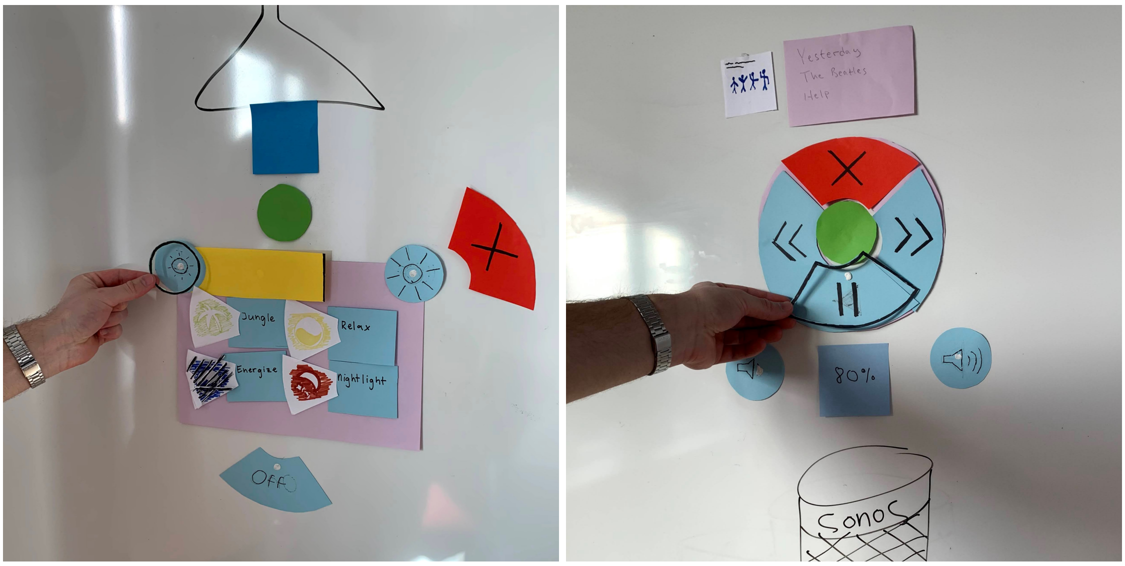

Figure 5.5: Left photograph showing Philips Hue paper prototype with square design and right photograph showing Sonos speaker paper prototype with donut design.

5.2.6 Take-aways

In this subsection, all the findings from the idea generation sessions as well as the testings are presented and summarized.

The Lo-Fi phase generated a lot of different ideas, many of which were later on implemented and tested. First of all, the bodystorming session was very fruitful since the perspective of the user was the main focus. By only wearing glasses while saying what actions that were performed, a lot of insights were discovered such as where to put certain buttons or how to perform a specific action. Since none of the authors had tried a similar system before, it was unclear what challenges regarding the interaction and design laid ahead. The bodystorming session therefore gave a better understanding of the possibilities and limitations that comes with gaze-based interaction.

After the 6-3-5 brainwriting session, 48 unique ideas had been collected. The ideas included aspects such as particular types of eye-movements for different actions, length of dwell times, and various forms of feedback towards the user. Similar to the brainwriting session, making the storyboards was a great opportunity for getting a better understanding as well as coming up with new ideas. The new ideas concerned mostly various ways of interaction. For example, it became apparent that feedback is extremely important in order for the user to understand that an action has had an effect. Things such as progress bars and animations was therefore decided to be an integral part of the UI in order to increase the usability. In the end, when the two storyboards were done, it felt much clearer what needed to be done, and both the authors had a mutual idea of a user-friendly and easy-to-use design.

The Lo-Fi testing gave a good perspective of the challenges and ambiguities that gaze-based navigation entails. Some features or buttons that the authors thought was clear was proven to be confusing for the test participants who saw the prototypes for the first time. For instance, the X-button located at the upper right corner of the UI baffled one of the test participants since it could have multiple purposes. In general, there was often confusion when opening and closing the UI. Two of the test participants did not think that the green ball was interactable. To make it more clear, one of the participants proposed supplementing the buttons with text. Another participant expressed worries of pressing a button by mistake if the dwell time was too short.

To conclude the Lo-Fi testing, all of the test participants preferred interacting with the Sonos speakers, no matter which design that was used. This is probably the case since they were more familiar with a traditional music player rather than controlling lights. When asked which design they preferred, regardless of what system they used, all of the test participants answered the more traditional square-shaped design rather than the donut-shaped one. This again might be the case since the square-shaped design is more commonly seen in smartphones and computer applications. The square-shaped design was therefore chosen for the next phase.

5.3 Hi-Fi phase

After a fruitful Lo-Fi phase it was time to embark into the more meticulous Hi-Fi phase. In this section, the design of the product as well as user interaction will be the main focus. Concepts such as mockups, user feedback, button properties and dwell time are brought up, explained and motivated.

5.3.1 Mockups

When evaluating the paper prototypes it was clear that the square design was preferred over the donut design. By virtue of that result, one mockup for each type of device was created, mimicking the preferred design. Creating mockups as a tool for developing a product is a good idea when it is unclear which direction the design should take. The mockups in this project were created to give the authors a reference point as to what the design should look like.

The mockups were created using Adobe XD[3] which is a UX design tool. Both background images are photographs taken at Jayways office in Malmö which represents the view from the AR glasses.

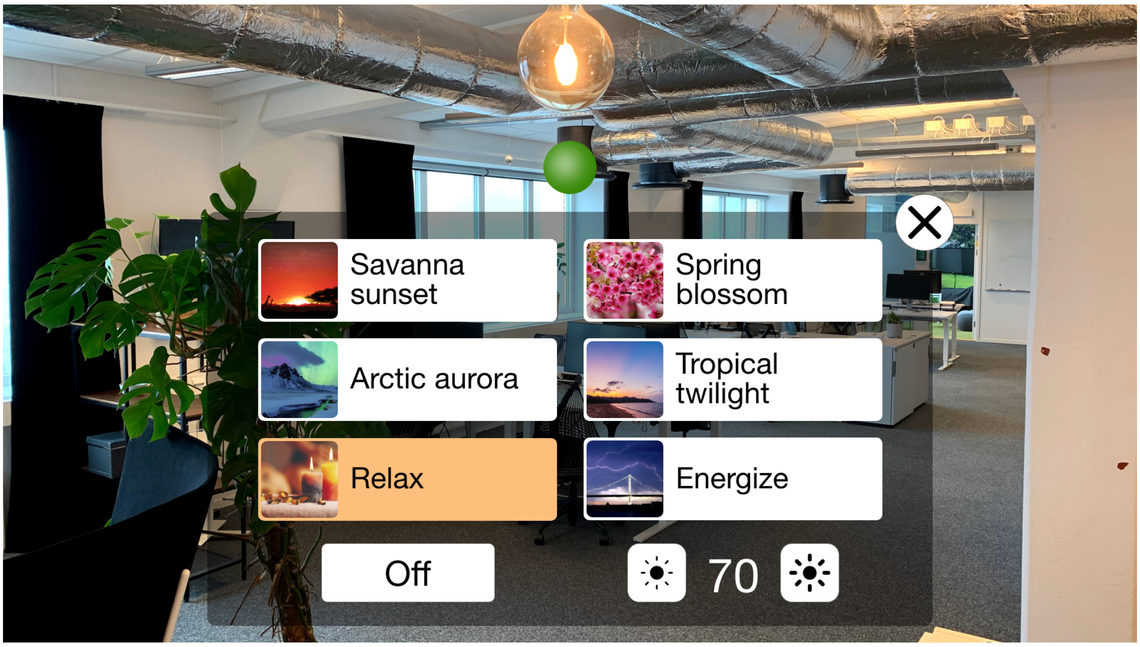

Figure 5.6: Mockup of the Philips Hue UI.

Starting with the mockup of Philips Hue seen in Figure 5.6, the design is partly based on the paper prototype and partly on the iPhone application for the product. In the figure, the theme *Relax* is highlighted which means that it is selected. The highlight color represents the actual color of the selected light, the same way as in the native app. A difference from the

[3]https://www.adobe.com/products/xd.html

paper prototype is the location and design of the brightness level. It was decided that the brightness level should be displayed in digits instead of a bar used in the iPhone application. This decision was based on the fact that it was more difficult to distinguish the brightness level of a bar than simple digits. It mostly depended on the outside environment - if it was sunny outside the visibility of the display became worse, making a light bar inadequate. Furthermore, the brightness indicator and buttons are located as far away from the light source as possible since it can be blinding for a user to look right next to the light bulb when brightening the light.

Figure 5.7: Mockup of the Sonos UI.

The mockup for the Sonos media player can be seen in Figure 5.7. It is solely based on iPhone's media player, which the paper prototype also is. The only difference is the volume level which is, similar to the Philips hue mockup, displayed in digits instead of a bar. The reasons are the same and it also keeps the design consistent between the two mockups.

A problem that was encountered when testing the mockups concerned the visibility of the display during a bright outside environment, for example when the weather was sunny. When having the Sonos or Hue application in front of a window on a bright day, it was difficult to see some of the more brighter colors. In order to counteract the sunlight without having to use the blinds, the background was first colored black while the buttons remained white. This contrast in colors made it much easier to see the buttons and labels, even on a bright spring morning. The background was later changed to a blue color since it looked better against the surrounding environment while still maintaining the necessary contrast. An overview of the tested Hi-Fi prototypes can be seen in Figure 5.8 and Figure 5.9.

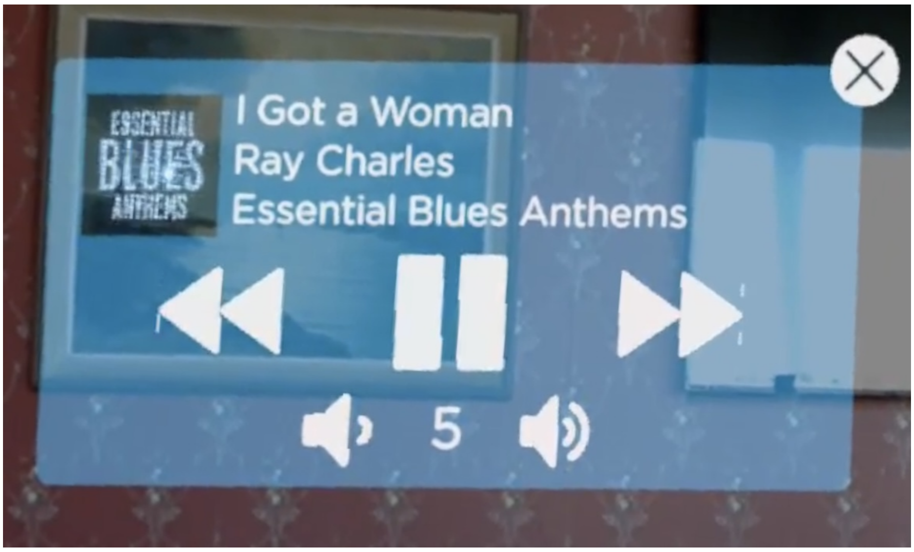

Figure 5.8: Screenshot of the tested media player UI.

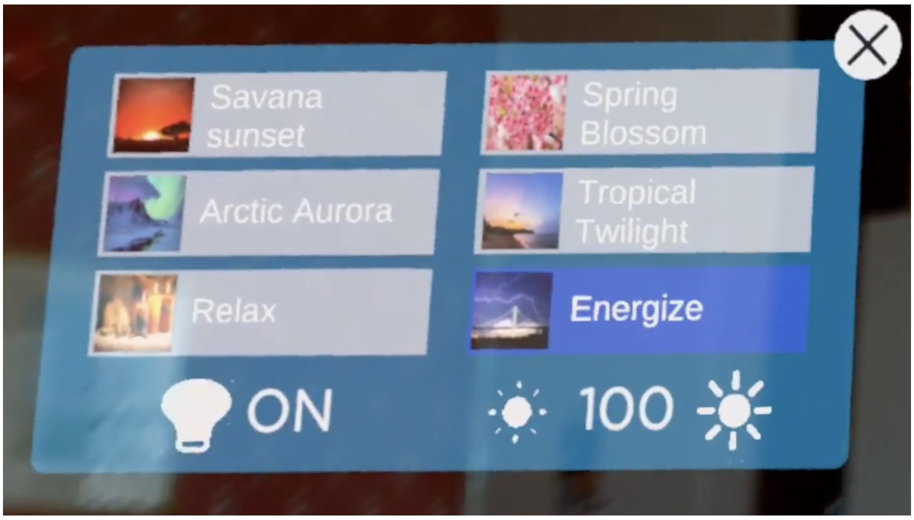

Figure 5.9: Screenshot of the tested light settings UI.

5.3.2 Feedback

During development it became apparent that feedback from the system was needed in order to use it effectively. Newton once said when explaining his third law: *for every action there should be a reaction.* The same principle can be somewhat applied to user feedback, meaning for every time a user performs something, the system should reassure that an action have been taken in the form of feedback. Feedback can be given to the user in a number of ways: auditory, visually, or tactile to name a few. During the project, visual feedback have been the main focus of the system since it is the most compatible variant of feedback when using eye-gaze.

Different actions require different forms of feedback. It became clear early on in the development process that feedback is needed when a users gaze meets a button. Using gaze as interaction can be difficult since there is a risk that the user might feel unsure whether they are performing an action or not. In contrary, smartphones that have touch as input gives the user instant tactile feedback when pressing a button. Unfortunately, the user is not given this sort of instant feedback when using eye-gaze, at least not naturally. The solution proposed in the Hi-Fi prototype was to highlight a button when it is gazed at by recoloring or darken the material. A highlighted button can be seen in Figure 5.10. Highlighting gives the user instant feedback that they have begun the process of performing an action when using the system. Another benefit of using this type of visual feedback is if the user starts to gaze at a button, looks away and then gazes at the button again. The color of the button will then change back to be non-highlighted for a brief moment when the gaze is outside the button. This might happen if the user changes his or her mind or if the ML1 is poorly calibrated. In a case like this the user gets feedback that they have restarted the action and understands that it takes some time to actually perform a button click.

Figure 5.10: Button is highlighted when the users gaze meets the hitbox.

Feedback is also given when a user gazes on a button until an action have been performed. For instance, when a user presses the *play* button on the Sonos part of the system, the icon changes to a pause symbol instead of the play symbol. This feedback is consistent with most other media players seen in smartphones and computers. An example can be seen in Figure 5.11.

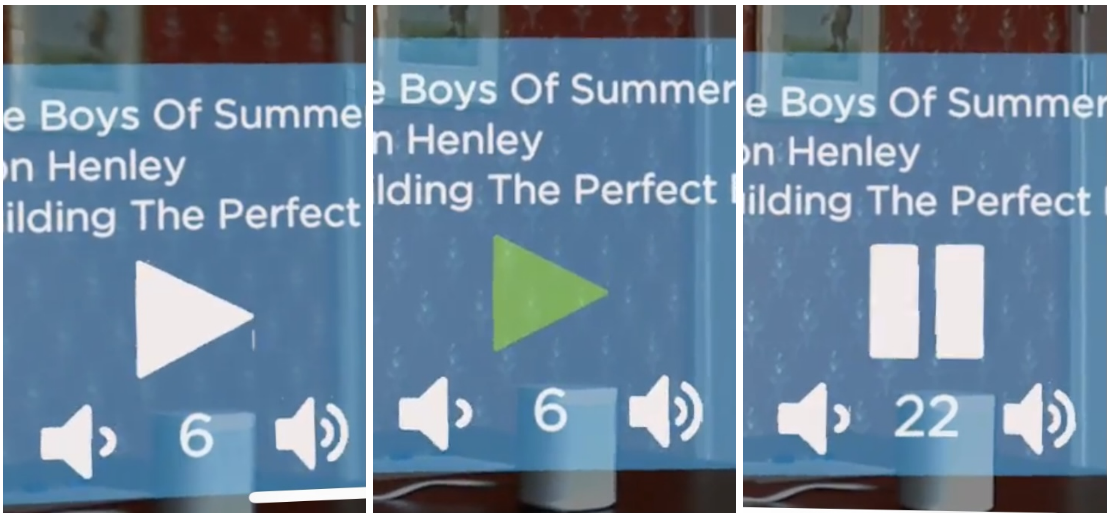

Figure 5.11: Play button changing to pause button when user performs the action of changing state to playing.

Additionally, a user that changes track by looking at the *next* button receives visual feedback when the next song's album cover, song title, artist and album name appears on the cover. Also, the new song does hopefully start playing in the first seconds of the track, giving the user auditory feedback as well. Figure 5.12 displays the process of changing track. A possible difficulty with the visual feedback can happen if the UI is outside the FoV. One solution to this was to make the scale of the UI fairly small, which in turn gives the user a better overview. At the same time, the auditory feedback given to the user when the next track starts playing is also helpful if the song information would appear outside the FoV.

Figure 5.12: Media player showing which track is currently selected with album cover, track name, album name, and artist.

Lastly, feedback is given to the user when changing volume or brightness in both parts of the system by changing the values simultaneously as the user looks at the buttons. This is partly the reason why the buttons are placed directly next to the volume indicator. These buttons work somewhat differently compared to the other buttons since they gradually changes the level step by step. A user changing the brightness can be seen in Figure 5.13.

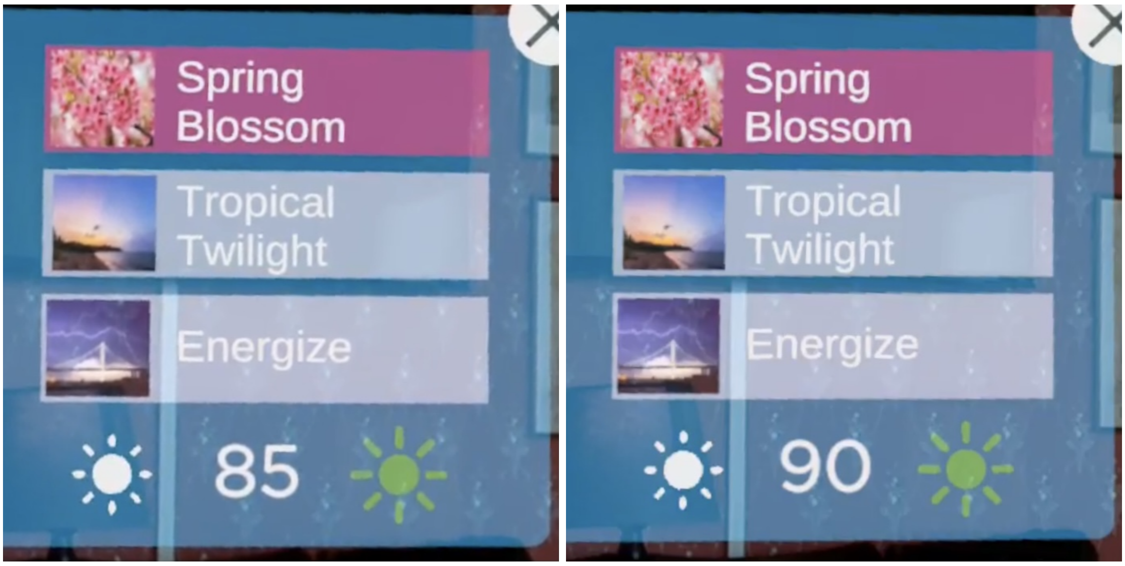

Figure 5.13: Highlighted button when users gaze meets a buttons hitbox.

5.3.3 Button properties

Since the projects goal was to evaluate eye-gaze as interaction, one would think that buttons need to be large with much spacing in order to prevent misclicks. While this is partly true, it was also discovered that the eye-tracker in the ML1 was very precise (if calibrated properly) making it possible to stand several meters away from the menus and still manage to press even the smaller buttons with fairly good accuracy. When taking into account that a normal use-case is in a regular sized room, it became apparent that the buttons do not have to be abnormally large. By having smaller buttons, less space is needed for the menus, and the risk of pressing an unwanted button is reduced. The spacing between the buttons could also be smaller than initially expected due to the well-performing eye-tracker. Apart from sizing and spacing, the position of the buttons tried to mimic the iPhone applications as much as possible. Both UIs had a 2:1 aspect ratio, which suited the FoV of the ML1 well. If too much information would have been stacked vertically, the user might have to move his or her head upwards and downwards in order to see everything.

One difference from the iPhone application is the button for closing the different windows. It was purposely made small since pressing it is relatively uncommon when interacting with one of the two systems. Also, by keeping the button small, the risk of misclicking it and closing the window is reduced. The choice of positioning it at the upper right corner was solely based on the fact that it is the most common place to have a close button.

5.3.4 Dwell time

In the project, the system uses dwell time to perform an action. This means that an action is executed when a user gazes at an object for a predetermined amount of time. Dwell time is an intuitive and easily implemented way of interacting using the eyes. The learning curve is also low, making it simple to grasp for new users [26]. The time it takes to perform an action is different for different buttons. Having different dwell times for various buttons can both have benefits and disadvantages. If the dwell time would be the same, the user can expect and measure the time to complete a button click. The interaction is consistent and predictable. However, some actions such as increasing the volume or brightness are stepwise, meaning that the result changes progressively. Therefore, if the dwell time for the stepwise buttons would be the same as the rest of the buttons, it would take a lot of time changing the volume or brightness level.

A compromise is therefore to group the buttons in two categories: stepwise and finite. The finite buttons includes for example the *On/Off* button in Philips Hue or *Play* in Sonos. These buttons have an immediate effect, and should therefore take longer time to press. For this category, the dwell time was set to 1.0 seconds. Meanwhile, the other category of stepwise buttons had a shorter dwell time of 0.25 seconds. For example, pressing *Volume Up* once raises the volume two steps out of 100. This means that it takes 2.5 seconds to increase the volume by 10 steps - a time not too long for the user to feel restless, but not too short either if it would be a misclick. This method of having variable dwell time has been proven to increase the UX in contrast of having a single dwell time for all buttons [27].

Chapter 6

User Study

This chapter describes the testing of the Hi-Fi prototypes, how it was setup, the participants and the outcomes.

6.1 Setup

All of the tests were performed in the same room located at Jayways office in Malmö. The room normally acts as a conference room, and was slightly modified before the test. Figure 6.1 shows how the room was organized. Seated at one of the armchairs was the test participant. The test leader sat on a sofa to the right while the observer sat on the left side of the table. In the room there were three smart devices, one Sonos speaker and two Philips Hue lamps. One of the lamps as well as the Sonos speaker was placed on the table. The other lamp stood on the floor at the right-hand side of the room. With this setup, the devices were decently spread out so that the virtual displays could not collide with each other. Additionally, the placement of the devices tried to be as realistic as possible. It was important that the prerequisites were the same for all of the test participants. Therefore, no objects were never moved between or during test sessions.

6.2 Participants

The test was performed on a total of 20 participants with ages ranging from 22 to 56. 8 people were female. Occupation varied but most were developers or students. Out of all test participants, 4 wore contact lenses. 18 people knew what AR was and the same amount had used it on a mobile device before. Astoundingly, 8 participants answered that they had previously tried out AR glasses and eye-tracking prior to the test. A total of 17 participants knew the meaning of IoT and 16 had practical experience using it. Moreover, 6 people an-

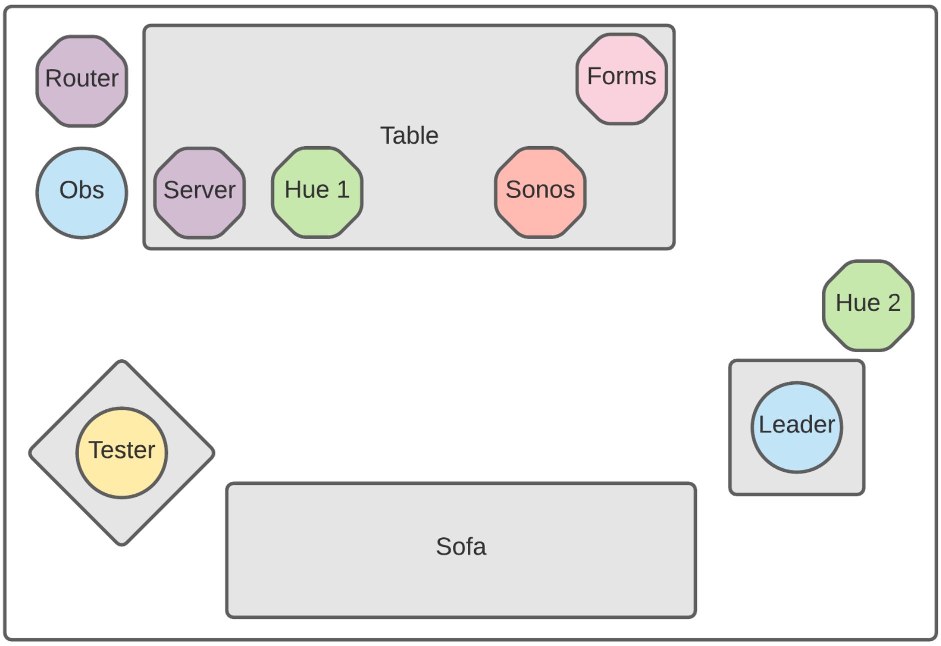

Figure 6.1: Test room showing the different roles and IoT devices. People are indicated by circles and objects as hexagons. *Server* is the computer with the Node.js server running. *Obs* is the observer and is responsible for monitoring the test. *Leader* is the test leader and sits in the armchair in front of the test participant (*Tester*).

swered that they had Sonos speakers at home while 3 said that they owned Philips Hue lights. All of the participants were familiar with the Spotify app while 7 respectively 3 had used the Sonos or Hue app before. Finally, 15 people had phones with iOS while the rest used Android.

Since the tests were carried out during a pandemic there were difficulties finding a considerable amount of participants. A somewhat restrictive goal of testing 20 people was therefore set which luckily became fulfilled. However, more participants would have been preferred in order to get more data and hence a more solid result.

6.3 Procedure

The testing sessions can be condensed into four main phases: preparation, briefing, tasks and debriefing. The phases and their contents can be found in Figure 6.2. Prior to the preparation phase, the participants were welcomed to the facility by the authors - which were the ones carrying out the tests. The participants were asked if they wanted a cup of coffee, tea or water.

The test participants were divided into two groups, A and B. Group A started testing the AR app while group B started with the native apps, i.e. the smartphone Sonos and Hue apps. The reason for starting with varying systems was to negate the transfer-of-learning effect.

Since group B started with the native apps they went back to the briefing phase in order to get more familiar with AR before doing the rest of the tasks.

Aspects that were measured included task completion time and errors. While it was difficult to exactly count the number of misclicks since it often were unclear whether an action was intended or not. However, when a participant explicitly made some sort of mistake, either by saying it or if it was very obvious, it was written down by the observer. The observer also wrote down interesting comments that was being said by the participants as well as the state of the eye-tracking. Number of clicks was not measured since it was deemed unnecessary, especially when comparing the two systems which had very different means of interaction. Also, a *click* in the AR system is quite vague and difficult to register as an actual button press or just a quick glance.

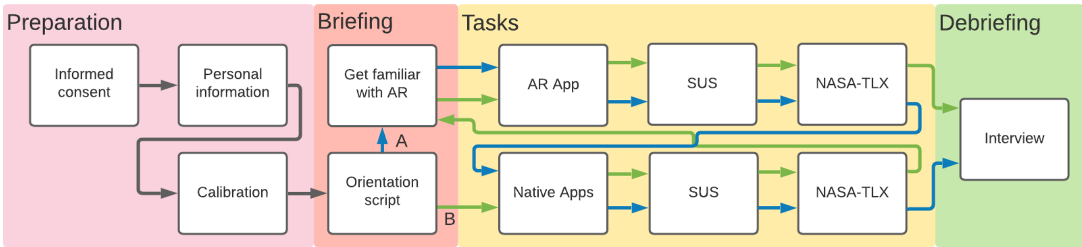

Figure 6.2: Test procedure step by step. Blue arrows represent group A and green arrows represent group B.

6.3.1 Preparation

After they had been welcomed, the participants were shown to the testing area and seated in the form filling section shown in Figure 6.1. They were first asked to fill out an informed consent form which can be seen in Appendix B.1. Secondly, they were instructed to answer another form about their personal information with questions regarding their age, occupation and prior experience with AR and eye-tracking. The personal information form can be seen in Appendix B.2. Lastly they had to calibrate the eye-tracking for the ML1. The calibration included putting on the AR glasses for the first time and running a pre-built program which took around one minute to complete.

6.3.2 Briefing

After the formalities were out of the way, the test session could commence. The participants were seated in the armchair shown as *tester* in Figure 6.1. The test leader read an orientation script giving a short introduction to the test with the purpose of briefing the test participants. The orientation script can be seen in Appendix A.5. While being briefed, the test observer setup the AR glasses for testing. The setup included setting up the test application, launching the AR warm-up game called *Create*, and starting the screen recorder on the ML1.

Create is a pre-built exploring game where the player can for example place virtual objects in a room. It was a suitable tool for letting the test participants explore AR before the actual tasks were given. The participants were asked to explore the environment in the application and notify the test leader when they felt comfortable enough with AR.

6.3.3 Tasks

Leading to this, the test leader started reading the task scenarios which can be found in Appendix A.6. The task scenarios differed very little depending on which system that was tested. During the tasks, the test observer wrote down observations and interesting comments that was being said by the test participant.

After each system was tested, the participant were again seated in the form filling section to fill out a NASA-TLX and SUS questionnaire about the tested system.

6.3.4 Debriefing

Lastly, a semi-structured interview was held regarding the tested systems. The questions can be found in Appendix A.7. Moreover, an additional open discussion with some of the participants was held afterwards. The discussions were held if the participant had any follow-up feedback they wanted to bring up. Before leaving, the participants were given a snack as a thank you for participating.

6.4 Results

6.4.1 SUS

The SUS scores were quite similar for the tested systems, but the native apps got a slightly higher score. Interestingly, the AR system had a bigger range with a minimum score of 45 and a maximum of 100. The minimum and maximum for the apps were instead 55 and 97.5. A total mean and median value can be seen in Table 6.1. The SUS score for each test participant can be seen in Figure 6.3. According to Bangor et. al., a score higher than 68 is considered to be above average, and the SUS scores that both systems received are rated as "Good" in agreement with their adjective scale [28].

Table 6.1: Total SUS score for each system

	Min	Max	Mean	Median	Mode	Range
AR	45	100	74.125	72.5	57.5	55
Native Apps	55	97.5	78.875	77.5	75, 92.5	42.5

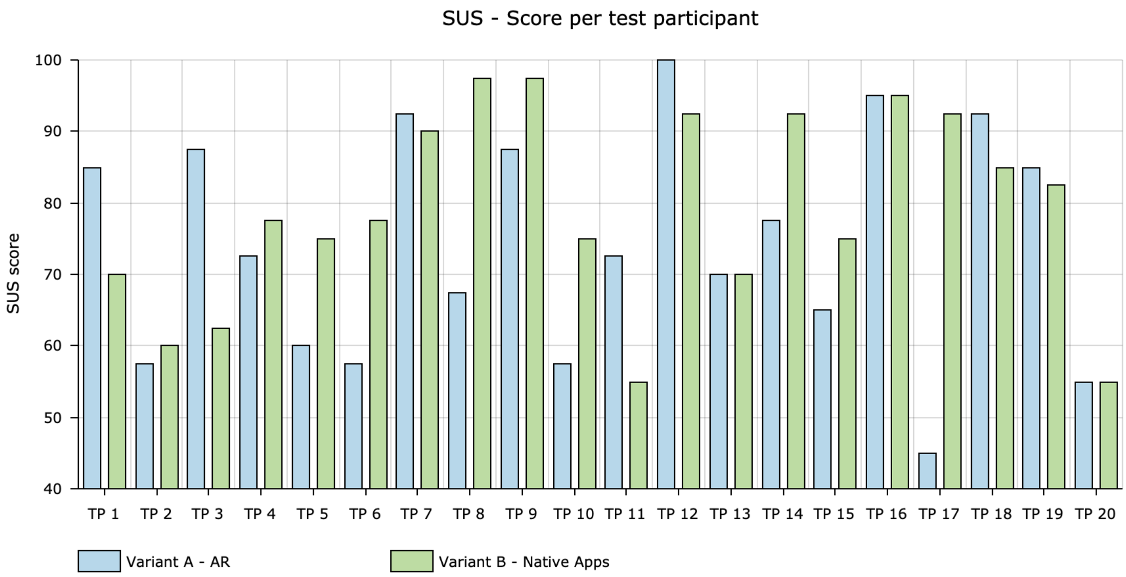

Figure 6.3: Mean SUS score for each test participant

6.4.2 NASA-TLX

A raw NASA-TLX subscale score for the two systems can be seen in Figure 6.4. The scores favored the native apps on all of the six questions. The biggest score difference between the systems was 22.5 on the question regarding effort.

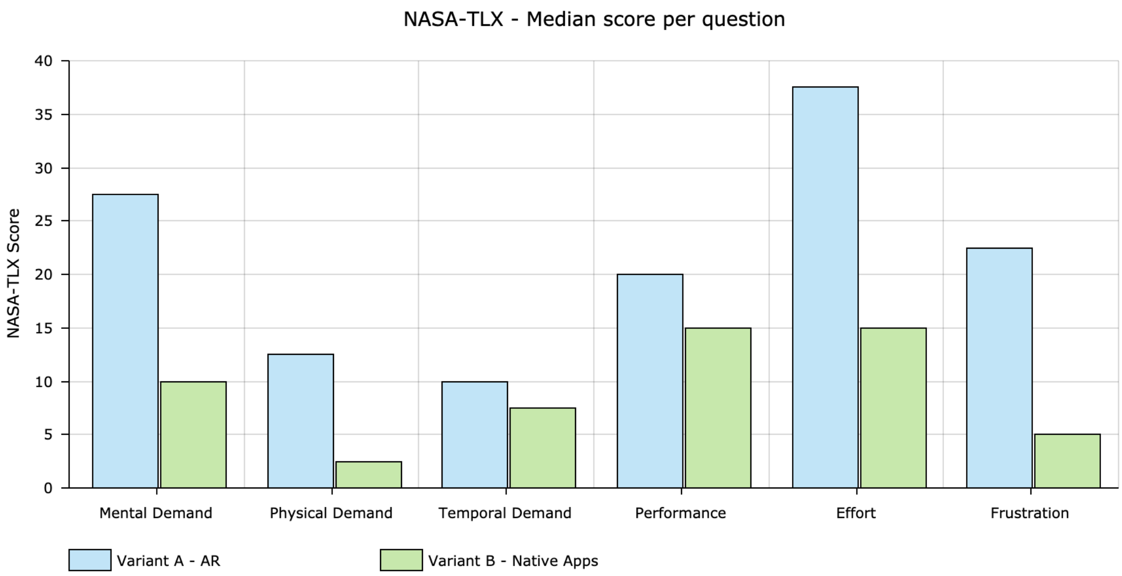

Figure 6.4: Median raw NASA-TLX score for each question

6.4.3 Observations

The observations gave a lot of insights regarding the functionality of the two systems. Several patterns could be seen, and some mistakes were repeated among several of the test participants. For example, some of the participants accidentally looked at the play button for too long when given the task of starting the music. This resulted in a quick toggle that paused the music immediately after it was played. Similar scenarios happened for other buttons as well, such as the *On/Off* button. From the observations it became apparent that the eye-tracking was inaccurate for some of the test participants, making the buttons flicker. This greatly reduced the usability of the AR system and resulted in misclicks and frustration. More on that in the next chapter.

Some of the test participants struggled with the apps as well. Most problems arose when using the Hue app. A number of test participants accidentally turned on both lamps when being asked to turn on the table lamp only. The same thing happened when they were supposed to change the color on only the floor lamp. Some also had difficulties even finding the lamps in the app. Fewer mistakes were made in the Sonos app. Nearly all of the test participants managed to complete every Sonos-related task without any help or struggle. Only one person mistakenly tried to look up a song instead of playing the track that was already selected.

A common misconception was that when being asked to close down the app or window, some test participants paused the music or turned off the lights as well. The test leader then had to intervene since the music and lights had to be on for a future task. While this did not affect the test too much it was interesting to see the same mishap happening regularly.

6.4.4 Interviews

Most test participants preferred the apps when having to choose either of the two systems. However, most of them liked the AR system but thought the technology was not mature enough. If they could for example use their own glasses many would have chosen the AR system over the native apps. Overall, many of the answers were reflected on the fact that the ML1 felt uncomfortable or that the calibration was inaccurate. Many expressed positive comments on the UI of the AR system, but said that the hardware made it cumbersome to use. Out of the two native apps, nearly all preferred the Sonos app, with the motivation that it resembled the Spotify app which everyone had used before.

When being asked about pros and cons with the respective systems, many expressed the simplicity of having one single system (AR) for controlling both types of devices. They also found it useful when having multiple IoT devices in the same room or when the hands are busy elsewhere, for example when cooking. The cons with AR were mainly focused on the ungainliness of the ML1 and having to calibrate it each time it is being used. The benefits of using the native apps were, according to many participants, the flexibility and already being accustomed to the technology. Since apps are common today, many were pleased of keeping it that way and thought it was handy to have a controller in their hands that could be stoved away at any time. Meanwhile, 6 of the test participants thought it was a big disadvantage that there were two separate apps on the smartphone in contrast with the AR system that could control both devices from the same view.

The answers for the question of how it felt controlling the IoT devices using eye-gaze were divided but mostly positive. A common thought concerned the insecureness of not knowing where to look when not performing a task. This could also be observed during the tests when some people stared down on the floor between tasks. They were simply afraid of accidentally clicking a button when looking at it out of curiosity. Meanwhile, many participants expressed a positive feeling of controlling the system using eye-gaze. Some people suggested blinking as a way to actually press a button. Others proposed a longer dwell time in order to reduce the risk of misclicks.

The fourth question regarding potential improvements received a whole bunch of helpful feedback. Much had however already been said in the previous questions. 5 people thought the close button was too small and therefore difficult to hit. Additionally, 3 participants thought the dwell time on the volume and brightness buttons was too short. In general, the common thought was that actions executed too fast, a problem they believed depended on the fact that they were first-time users and not accustomed to the UI. One test participant proposed a solution where the user have the ability to manually set the dwell time.

Next, on the question where the participants had to choose one of the two systems, 4 people picked AR while the rest opted for the native apps. However, many specifically said that their answer completely depended on the hardware, that is the AR glasses, and not the concept in the form of AR and eye-tracking. Moreover, if the AR system could be fitted in regular glasses or even contact lenses, several test participants would have preferred and chosen that system instead. This was the main reason why some participants rated the AR system higher in the SUS questionnaire but still picked the apps if having to choose one.

Finally, during the last question, the participants had a final chance to provide any useful thoughts and feedback. One person suggested auditory feedback as an aid to improve the usability. Two participants would have wanted to see some sort of progress animation when looking at a button. Other comments were more directed towards AR and eye-tracking as technologies. Since no one had tried gaze-based interaction before, many were fascinated by the technology and overall optimistic about the future and how it will evolve over time.

Chapter 7
Discussion

The following chapter discusses both the design process and technical development, but also the results from the user study.

7.1 Design process

7.1.1 Lo-Fi phase

The Lo-Fi phase was an important step in order to come up with fundamental design principles, user feedback as well as ways of interacting. Without creating Lo-Fi prototypes, the authors would have drowned from an overwhelming amount of ideas. Instead, by conducting different idea generating processes and tests at an early stage in the overall process, it was possible to narrow down and cut away a majority of proposals. This in turn hugely facilitated the Hi-Fi and technical development.

While this phase was valuable it sometimes felt rushed. Since the project was limited to 20 weeks, much of the workload had to be allocated towards development and final testing. This meant that the Lo-Fi phase had to be put on low priority. In hindsight, it would have been interesting to further investigate different designs. With the feedback that was given during the final tests in mind, it would have been intriguing to see how for example the donut design would have held up.

The Lo-Fi tests went well. However, only four people participated which is not a lot. This is another consequence of the limited time-frame. If more people would have tested the two Lo-Fi prototypes, the outcome would have perhaps been different. Also, the background of the test participants were similar, with all four being, or are about to be, developers with a keen interest in different technologies. If other occupational groups would have been included, the result would have been more nuanced. The same goes for the 6-3-5 brainwriting session;

the people participating only consisted of engineering students. Moreover, the brainwriting session is originally intended for six people, not four which was the number of participants in this case.

Even though many different activities were used for generating ideas and testing design proposals, there is still not much research of how to test gaze-based systems at a Lo-Fi state. During the Lo-Fi tests, the authors used paper circles that tried to mimic where the participants were looking. This approach felt useful at the time but did not give a lot of helpful feedback in terms of functionality. Therefore, when it comes to Lo-Fi development, it is difficult to design and adapt a gaze-based system just using paper prototypes. An overall design approach can definitely be made, but more functional aspects in terms of eye-gaze have to be dealt with in the Hi-Fi phase.

7.1.2 Hi-Fi phase

With a successful Lo-Fi phase in the basket, the Hi-Fi phase went smooth. There were few to none backlashes when it came to the creation of the UI design. The technical difficulties that emerged can be read in the next section.

Feedback was an important factor when coming up with the Hi-Fi prototype. Using dwell time was an early candidate that have been proven to be efficient in other studies. However, simply highlighting the button that was being pressed was perhaps not enough in terms of feedback. When being a regular user, which the authors were at the time, it felt natural to only highlight the button, especially since the dwell time duration was known. It was during the Hi-Fi tests that many participants remarked on the lacking feedback, most probably since they were first-time users. Some sort of loading animation was discussed at an early stage, but canceled due to technical limitations. Other ways of presenting feedback to the user could have been gathered if a Mid-Fi test would have been conducted. This type of test could for example exclusively focus on feedback, with users only interacting with an UI not connected to any IoT devices. However, once again, this idea was scrapped due to time limitations.

While dwell time was chosen as the way to perform actions, other means of interaction would have been interesting to look at as well. For example, *saccade patterns* where a user draws relevant figures with their gaze trail, could prove to be an efficient way. Sadly the ML1 was quite limited in terms of registering moving gaze trails. There were simply not enough resources in the Lumin SDK that could enable the possibility of alternative gaze-based interaction methods other than dwell time. It seems like Magic Leap have placed all their cards on dwell time as the de facto method for gaze-based interaction. While it might have been possible to develop a script for registering for example saccades, both time and documentation were unfortunately lacking so that idea had to be scrapped. Investigating saccades could be an interesting master thesis on its own.

7.2 Technical development

7.2.1 Server

The process of creating and working with the server was easy since Node.js and Express fixes all underlying configurations for simple servers like the one used in this master thesis. However, some problems occurred when working with the Sonos API which still existed during testing. The package used for working with the Sonos API utilized Axios which is a promise-based HTTP client for Node.js servers. When connecting the server with Unity, some timing issues emerged which sometimes lead to the wrong state being sent to the AR glasses. This resulted in the UI displaying another song's information rather than the one that was actually being played. These issues could probably have been fixed if the authors had put more time on learning proper asynchronous programming between the server and the scripts in Unity.

7.2.2 Unity

Creating 3D-objects and other visual effects was relatively streamlined since it is what Unity was created for. One caveat though was working with materials and placements. Since the AR glasses uses the real world as its environment, creating materials in Unity is done differently. Luckily, Magic Leap's API and Lumin Software Development Kit (SDK) provide ways for creating materials specifically designed for usage in ML1 and the real world. A problem with this however is that there was lacking documentation, so the authors often had to adopt trial and error in order to solve a problem.

Furthermore, placing objects proved to be problematic since the glasses reset the starting position each time the application was restarted. There are methods for saving the positions based on real-world coordinates, but due to a limited time-frame, those methods were not fully investigated. In hindsight, it would have been advantageous to save the coordinates for the virtual objects, both in terms of saving time, but also for making the application more flexible. In addition, since the virtual objects had to be manually placed before each test, the participants had to remove the ML1 after calibration. This resulted in reduced eye-tracking accuracy during the tests which worsened the overall experience.

Lastly, halfway through the development, Magic Leap rolled out a new update of the Lumin SDK to version 0.25.0. The update changed how the API was structured which lead to a complete refactoring of the code which halted the development in many ways. Also, the old documentation became obsolete, so many forums and communication channels had to be used instead. Although the update provided the authors with many problems, it also actually improved the API immensely by making it easier and more effective. This shows that Magic Leap is keen on making gaze-based interaction more accessible and easy to develop.

7.3 User study

7.3.1 Testing AR

Although the testing sessions went well, some things could have been done differently when testing the AR application. A problem the authors noticed was that the eye-tracking got worse for some people when first calibrating the ML1 and then taking them off again. A better solution would been to calibrate, introduce AR and lastly perform the tests without taking off the glasses. This to ensure a better eye-tracking quality. A problem with this approach was that the authors had no way of setting up all these stages remotely from the computer. Instead the test observer had to put on the ML1 and setup the test environment - for example placing the virtual objects in the correct positions - between calibration and testing.

Moreover, it would be beneficial if the authors could see what the participants were seeing in real-time during the testing sessions. ML1 can stream the device's view in real-time using the streaming platform Twitch. This approach was tried but the stream crashed a couple of times during pilot testing and the authors did not want to risk that happening during the actual tests.

Lastly, since all of the 20 participants completed every single task (with different levels of success) one could argue that a gaze-based system can be developed successfully, but the question still stands how it can be made as efficient as possible. As mentioned in Section 7.1.2, *saccade patterns* would have been interesting to research and test against dwell time in order to see which method works best.

7.3.2 Testing Apps

Similarly to AR, testing the native apps went mostly flawless. Some noteworthy comments however was that the phone was an iPhone, which five of the test participants did not use themselves. This could have affected the result in the SUS and NASA-TLX questionnaires. Also, in the Hue app, the names for the table and floor lamps were *Lamp 1* and *Lamp 2*. These names could have easily been changed to something more suitable. This naming flaw might have also affected the SUS and NASA-TLX score negatively.

7.3.3 SUS and NASA-TLX

Even though the two tested systems scored closely, the native apps were preferred when looking at the mean and median of the SUS and NASA-TLX scores. Interestingly though is that a big part of the test participants scored the two systems somewhat similarly (see for example Figure 6.3). It was only a few that gave the AR a much lower score. An explanation might be that the eye-tracking worked poorly for those test participants. These hardware issues can potentially be the cause for people giving poor ratings on the AR system. While it was mostly the UI and way of interacting that was being tested, the ML1 became the focus point for peoples dislikes. One can argue that the scores are somewhat misleading if the sole purpose is to test the application and not the glasses.

On the other hand, some people might have scored the AR application higher based on thinking that the technology is new and interesting. Many of the participants were very technically interested and had used IoT devices and AR before. They probably welcome new technology with open arms and might therefore rate AR higher than people with less technical interest. Also, most of the participants were friends, colleagues or acquaintances with the authors. Although the authors made it clear about the importance of honesty, this is an important factor that might skew the result since they might rate the AR system high just to be kind.

Moreover, when testing the native apps, there are two different applications for the speaker and for the lamps. When answering the questionnaire some participants were unsure how to score the system since they rated the apps differently. For example, some people felt comfortable using the Sonos app while being more confused when testing Hue. The authors told those people to try to evaluate both apps as a whole system. This in turn could lead to some misinterpretations. If the apps would have been scored independently, the result might have looked different.

Lastly, it is difficult to tell if the participants misinterpreted some of the questions. Since both NASA-TLX and SUS are standardized and quite old, some of the questions might have been irrelevant to the test. There is a risk that some participants did not fully understand the language. For example, nearly all participants asked what the word *cumbersome* meant when filling in the SUS questionnaire.

7.3.4 Semi-structured interview

The interview was a good addition to the testing sessions. Participants gave good feedback and some came up with new interesting ideas to try out for future work. However, many answers were colored by the fact that the AR glasses felt clumsy and uncomfortable. This was a result of the questions being asked in a wrong way. The purpose was to see which way of interacting, that is using touch or eye-gaze, that was preferred. Therefore, it would have been better to ask which *interaction method*, instead of *system*, they liked the most. Some participants got the question of which system they favored if the AR system would fit into a pair of regular glasses or even contact lenses. The answers were then quite different, but unfortunately, since not every participant was asked this question, it could not be included in the results. However, the fact that some participants changed their preferred system when being asked that question can somewhat answer the last research question. Since the state of the hardware is too clumsy and uncomfortable, one can draw the conclusion that AR technology is not good enough for everyday use.

Moreover, two questions were quite similar: *Which system did you like the most?* and *If you had to pick one system to use, which one had you chosen?*. Naturally most people answered the same system for both questions since they want to use what they like the most. The second question therefore felt unnecessary, something that was not realized until the tests were ongoing.

One test participant in particular mentioned that the AR system would be really handy if the person was *laying in the bed feeling lazy*. Other participants said that they would prefer the AR system when cooking or in a situation when they need both of their hands for other purposes. These are possible scenarios where a gaze-based AR system for controlling IoT devices can be preferred over native smartphone applications, answering the first research question.

7.4 Future Work

Future work includes testing other types of feedback such as auditory and haptic. Also, further investigating different types of loading animations is an interesting concept that can let the user know that an action has had an effect. Saccade patterns are also an interesting way of making choices. For example can an eye-drawn circle indicate *yes*. The technique could be a good compliment to dwell time. Finally, computer vision could be used to track and detect IoT devices in an environment. As of now, pre-determined anchor points are set in Unity, making the application non-versatile. If a device could be detected with computer vision, the application could be launched from anywhere.

Chapter 8
Conclusion

The purpose with this master thesis was to see whether it is possible to use AR together with eye-tracking in order to control IoT devices using gaze-based interaction. The goals were more specifically to have a user-centered design, efficiently communicate with IoT devices and to give reliable and consistent feedback to the user. When testing the implemented AR system against the more traditional way of controlling IoT devices using a regular smartphone, the results varied. Most test participants rated the native apps higher than the AR system, both in a SUS and NASA-TLX questionnaire. However, difference in score was not that much, so a final conclusion can not be entirely drawn. What can be concluded is that the AR technology is not mature enough in terms of hardware. Furthermore, a system with only eye-gaze as input is definitely possible since all test participants managed to complete every task with mostly positive feedback. Therefore, if the AR system could be fitted into a more compact piece of hardware, such as regular glasses, the concept of interacting using eye-gaze is absolutely feasible in the future.

To summarize, three points can be taken from this master thesis: (1) As of now, head-worn AR systems with eye-tracking are not preferred over native smartphone applications when controlling IoT devices, (2) It is possible to make different kinds of interactions with only eye-gaze as input while still maintaining a good usability, and (3) AR glasses are not comfortable or adaptable enough today in order for them to be used at a daily basis.

References

[1] P. Schueffel, "The concise fintech compendium," *Fribourg, Switzerland*, 2017.

[2] I. E. Sutherland, "A head-mounted three dimensional display," in *Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I*, AFIPS '68 (Fall, part I), (New York, NY, USA), p. 757–764, Association for Computing Machinery, 1968.

[3] P. Cipresso, I. A. C. Giglioli, M. A. Raya, and G. Riva, "The past, present, and future of virtual and augmented reality research: A network and cluster analysis of the literature," *Frontiers in Psychology*, vol. 9, p. 2086, 2018.

[4] J. Araujo, G. Zhang, J. Hansen, and S. Puthusserypady, "Exploring eye-gaze wheelchair control," 06 2020.

[5] R. Bates, M. Donegan, H. O. Istance, J. P. Hansen, and K. J. Räihä, "Introducing cogain — communication by gaze interaction," in *Designing Accessible Technology* (J. Clarkson, P. Langdon, and P. Robinson, eds.), (London), pp. 77–84, Springer London, 2006.

[6] D. Bonino, E. Castellina, F. Corno, and L. De Russis, "Dogeye: Controlling your home with eye interaction," *Interacting with Computers*, vol. 23, pp. 484–498, 09 2011.

[7] E. Mollenbach, J. Hansen, and M. Lillholm, "Eye movements in gaze interaction," *Journal of Eye Movement Research*, vol. 6, 01 2013.

[8] A. Lundqvist Persson, "Eye-tracked gaze gesture implementation in an augmented reality system," 2020. Student Paper.

[9] D. Tovesson, "Control your home with augmented reality," 2020. Student Paper.

[10] T. Hewett, R. Baecker, S. Card, J. Gasen, M. Tremaine, G. Perlman, G. Strong, and W. Verplank, "Acm sigchi curricula for human-computer interaction," 09 1992.

[11] A. Dix, A. J. Dix, J. Finlay, G. D. Abowd, and R. Beale, *Human-computer interaction*. Pearson Education, 2003.

[12] D. A. Norman, "Natural user interfaces are not natural," *Interactions*, vol. 17, p. 6–10, May 2010.

[13] L. Graichen, M. Graichen, and J. F. Krems, "Evaluation of gesture-based in-vehicle interaction: User experience and the potential to reduce driver distraction," *Human Factors*, vol. 61, no. 5, pp. 774–792, 2019. PMID: 30694705.

[14] R. J. Jacob, "Eye tracking in advanced interface design," *Virtual environments and advanced interface design*, vol. 258, p. 288, 1995.

[15] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.

[16] D. A. Robinson, "A method of measuring eye movemnent using a scieral search coil in a magnetic field," *IEEE Transactions on Bio-medical Electronics*, vol. 10, no. 4, pp. 137–145, 1963.

[17] A. Keren, S. Yuval-Greenberg, and L. Deouell, "Keren as, yuval-greenberg s, deouell ly. saccadic spike potentials in gamma-band eeg: characterization, detection and suppression. neuroimage 49: 2248-2263," *NeuroImage*, vol. 49, pp. 2248–63, 10 2009.

[18] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[19] B. Martin, B. Hanington, and B. Hanington, *Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions*. Rockport Publishers, 2012.

[20] P. B. Paulus, R. M. Korde, J. J. Dickson, A. Carmeli, and R. Cohen-Meitar, "Asynchronous brainstorming in an industrial setting: Exploratory studies," *Human Factors*, vol. 57, no. 6, pp. 1076–1094, 2015. PMID: 25850113.

[21] D. Rohrer, K. Taylor, and B. Sholar, "Tests enhance the transfer of learning," *Journal of experimental psychology. Learning, memory, and cognition*, vol. 36, pp. 233–9, 01 2010.

[22] J. Brooke, "Sus: a "quick and dirty'usability," *Usability evaluation in industry*, vol. 189, 1996.

[23] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 9, pp. 904–908, 2006.

[24] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 9, pp. 904–908, 2006.

[25] S. Greenwold, "Spatial computing," *Massachusetts Institute of Technology, Master*, 2003.

[26] P. Majaranta, J. Laitinen, J. Kangas, and P. Isokoski, "Inducing gaze gestures by static illustrations," in *Proceedings of the 11th ACM Symposium on Eye Tracking Research amp; Applications*, ETRA '19, (New York, NY, USA), Association for Computing Machinery, 2019.

[27] Z. Chen and B. E. Shi, “Using variable dwell time to accelerate gaze-based web browsing with two-step selection,” *International Journal of Human–Computer Interaction*, vol. 35, p. 240–255, Mar 2018.

[28] A. Bangor, P. T. Kortum, and J. T. Miller, “An empirical evaluation of the system usability scale,” *International Journal of Human–Computer Interaction*, vol. 24, no. 6, pp. 574–594, 2008.

Appendices

Appendix A
Test plan

A.1 Purpose

The purpose of this test is to see whether it is easier and more intuitive to control IoT devices using eye-tracking instead of with a smartphone application.

A.2 Test participants

The target group for this test will be people who have IoT devices in their household, and are generally interested in new technologies. Another target group is people with neuromuscular diseases who are unable to use their arms or hands. All in all, there are many potential users for this system.

A.3 Test data

Both subjective and objective data will be collected.
Objective:

- Task completion
  Can the test participants complete the tasks that are given?

- Time to completion
  How long does it take to complete a task?

- Number of errors
  How often do errors occur?

- Amount of times that assistance is required
  How often do the test participants need help from the test leader?

Subjective:

- SUS
  Measures the usability for each system
- NASA-TLX
  Rates for example the perceived workload

A.4 Tasks

The tasks that will be performed during the tests by the test participant can be seen in table A.1.

Table A.1: Test participant tasks

Task	Subtask	Finished when	Time cap
1. Open the media player	1.1. Find the Sonos marker 1.2. Press the marker until the media player pops up	The media player is visible	1 minute
2. Start playing music	2.1. Press the “Play” button	The music is playing	1 minute
3. Change to the next track	3.1. Press the “Next” button	The next track is playing	1 minute
4. Change the volume	4.1. Press the “Volume down” or "Volume Up" button	The volume has been changed	1 minute
5. Close the media player	5.1. Press the “Close” button	The media player is closed and the Sonos marker is visible again	1 minute
6. Open the light settings for table lamp and turn it on	6.1. Find the Hue marker above the table lamp 6.2. Press the marker until the light settings pops up 6.3. Press the "On" button	The light settings for the table lamp is visible and the light is on	1 minute
7. Change theme on the table lamp to "Spring Blossom"	7.1. Press the “Spring Blossom” button	The theme on the table lamp is switched to "Spring Blossom"	1 minute
8. Turn on floor lamp and change theme to "Spring Blossom"	8.1. Locate and press the Hue marker above the floor lamp 8.2. Press the "Spring Blossom" button	The floor lamp is turned on and the “Spring Blossom” theme is selected	1 minute
9. Raise the brightness of the table lamp to 85%	9.1. Locate and press the “Brightness up” button on the table lamp 9.2. Stop pressing when the brightness level is at 85%	The brightness level is at 85%	1 minute
10. Pause the music	10.1. Press the “Pause” button in the media player	The music is paused	1 minute
11. Change floor lamp’s theme to "Energize"	11.1. Press the "Energize" button in the light settings of the floor lamp	Floor lamp’s theme is changed to "Energize"	1 minute
12. Close both the light settings and media player windows	12.1. Press the "Close" button on all the windows.	All the windows are closed and the markers are visible	1 minute

A.5 Orientation script

The purpose of having an orientation script was to give the test participants the same prerequisites. It was also meant to give the test participants a brief overview and introduction of the test session and the soon-to-be tested system.

> Hi and welcome to this testing session where you will get the chance to steer smart home devices! I am your test leader and will be responsible for giving you tasks during the test. The person sitting at the computer is responsible for the technicalities and observations.
>
> You will perform two separate tests that will together last for about 30 minutes. During each test you will:
>
> – Test one of the systems
> – Answer a usability questionnaire
> – Answer a workload questionnaire
>
> When both tests are done you will answer some additional questions in a final interview
>
> You will test two different systems for controlling two smart lamps and a smart speaker. The speaker is located at the right side of the table. The lamps can be found to the left of the table and next to the armchair.
>
> The first/second system you will test is:
> The original applications, that is the Sonos and Philips Hue mobile app. These apps come as standard when you buy the products, and you will borrow a phone with iOS during the test. The AR glasses that you just calibrated. These uses eye-tracking. In this system you make a choice by looking at the wanted button for a certain amount of time.
>
> Remember that it is the systems, not you, that are being tested. If you run into any problems we will help you. During the test, we encourage you to think aloud and give us your thoughts. You can stop the test whenever you want, and you do not have to specify why. Good luck!

A.6 Task scenarios

The following task scenarios are read by the test leader and gives the test participant a context as well as a hidden instruction of what to do next. The task instructions differs a bit when testing the AR applications versus the mobile app. However, the tasks are the same.

A.6.1 AR

1. You get home after a tough day at work and feel tired. You're sitting in the armchair wearing your AR glasses.
2. You feel that some music would be nice. Start the media player and turn on the music.
3. You don't like the current track. Change to the next song.
4. You're not satisfied with the volume. Change it to a desired level.
5. You are satisfied with the song and volume. Close the media player.
6. You think it's quite dark in the room. Open the light settings window for the lamp on the table and start the Philips Hue lamp.
7. The current light theme that is selected does not suit you. Change the theme to "Spring blossom".
8. It's still too dark in the room. You like the table lamp's theme and want to use the same on the floor lamp.
9. You think the table lamp's brightness is a bit too low. Change it to 85%.
10. The music is getting annoying and choose to pause it.
11. You feel tired and want to try the "Energize" theme on the floor lamp.
12. Everything is nice and you are done controlling both the lights and sound. Close the light settings and media player windows.
13. You are now allowed to further explore the system if you want. The test is done, good job!

A.6.2 App

1. You get home after a tough day at work and feel tired. You're sitting in the armchair with your phone in your hand.
2. You feel that some music would be nice. Start the Sonos App and turn on the music.
3. You don't like the current track. Change to the next song.
4. You're not satisfied with the volume. Change it to a desired level.
5. You are satisfied with the song and volume. Close the App.
6. You think it's quite dark in the room. Open the Hue App and turn on the lamp on the table.
7. The current color that is selected does not suit you. Change the table lamp color to pink.

8. It's still too dark in the room. You like the table lamp's theme and want to use the same on the floor lamp.
9. You think the table lamp's brightness is a bit too low. Change it to 85%.
10. The music is getting annoying and choose to pause it.
11. You feel tired and want to try a white light with 100% brightness on the floor lamp.
12. Everything is nice and you are done controlling both the lights and sound. Close both apps. The test is now done, good job!

A.7 Interview questions

The following interview questions are read by the test leader or test observer and given to the test participant after both tests have been accomplished.

1. Which system did you like the most?

   If test participant prefers the original apps:

   (a) Was any app (Sonos or Hue) better?

   (b) Was it easy to find the right lamp using the app?
2. Do you see any pros and cons with the respective system?
3. How did it feel to control the light and music with your eyes?
4. What could be done better? Was anything particularly easy or difficult?
5. If you had to pick one system to use, which one had you chosen?
6. Do you have any other questions or thoughts?

A.8 Roles

In a testing session, a test leader and an observer is present. The test leader is responsible for reading the orientation script as well as guiding the test participants through the test. The observer on the other hand is responsible for observing and writing down findings and comments throughout the test. Furthermore, the observer is responsible for setting up the different systems for testing. The post-test interview can be held by either the test leader or observer.

A.9 Equipment and testing environment

The test participants are wearing the ML1 (Lightwear + Lightpack) during the test. Meanwhile, the assistant is controlling the computer which contains the Node server responsible for communicating with the IoT devices. After each testing session, the equipment is thoroughly sanitized.

The room where the tests are performed is located in Jayways office in Malmö. The Philips Hue lamps are located on the table and on the floor. The Sonos speaker is also placed on the table so that it sits between the two Hue lamps.

Before each test starts, some items and equipment must be in place. This means:

- Forms
  - Informed Consent
  - Personal Information
  - SUS x2
  - NASA-TLX x2
- Magic Leap 1
  - LAN enabled
  - Wi-Fi bridge connected
  - Twitch stream up and running
  - Markers placed
  - Create up and running
- iPhone
  - Test screen is selected
  - Screen recorder is on
- Server
  - Spotify list queued
  - Spotify volume at 15 units
- Test area
  - Sanitary napkins ready
  - Chocolate balls ready
  - Coffee machine ready
  - Face masks x2

A.10 Calibration

Calibrating the eye-tracking of ML1 is essential for getting a good user experience. Therefore, a calibration will occur for each test participant before starting the test. The calibration itself takes only about a minute to complete. It is a good starting point for the test participants, since they can test out and become more familiar with the ML1. Also, the calibration gives them a good estimate of the FoV.

A.11 Procedure

Informal consent
Before the testing begins, an informal consent has to be filled by each test participant. The informal consent states for example that the participants are to be recorded and observed, and that it is okay to stop the test whenever. The form can be found in appendix B.1.

Personal information
This is another form that is filled before the actual test begins. This form includes information about the test participant, for example age, prior experience with AR and the ML1. The form can be found in appendix B.2.

Calibration
The ML1 is properly calibrated as a last step before the test participant leaves the form area and is read the orientation script.

Orientation script
The orientation script is read out loud by the test leader. The orientation script can be found in appendix A.5.

Test 1
Half of the participants begin with the mobile application while the other half starts with the ML1 application. This is done in order to minimize the transfer of learning effect. The test leader tells the participant each task, one by one. When the test participants have successfully finished a task, the next one is presented by the test leader. This goes on until all of the tasks are done. The tasks can be found in Table A.1.

SUS and NASA-TLX 1
After the first test, the participants have to fill out a SUS questionnaire found in appendix B.3 and a NASA-TLX questionnaire found in appendix B.4.

Test 2
The other system is tested, either the ML1 or mobile application.

SUS and NASA-TLX 2
A SUS and NASA-TLX questionnaire is once again filled out, now regarding the other system.

Interview
A final semi-structured interview is held where the test participants are asked questions about each system, which one they preferred and why. The questions can be seen in appendix A.7.

Appendix B

Forms

Two forms were filled out by the test participants before starting the test session. The first form to be filled out were a consent form which can be seen in section B.1. Secondly, a form regarding personal information were filled out by the participants which can be seen in section B.2.

After both tests the participants filled out two more forms, one being a SUS form seen in B.3 and one being a NASA-TLX form seen in B.4.

B.1

Informed Consent

Participant ID:

Thank you for participating in this test session. During the test, which will last for around 30 minutes, you will interact with two different Augmented Reality (AR) applications, one using a Smartphone and one using Magic Leap One (AR glasses). If you do not know what AR is, Pokémon Go is a famous example. It is a technique where you put virtual elements in the real world.
Both applications will let you interact with Internet of Things (IoT) devices, which are devices connected to the internet. More specifically, the IoT devices are Philips Hue smart lights, and Sonos smart speakers. The test session will be timed, so perform the tasks as efficiently as possible, but do not feel stressed.

By participating in this test session, you agree to the following:

- The phone you are using during the test will have its screen recorded.
- The AR glasses you are using during the test will have its screen recorded.
- The whole test session will be recorded using a camera with sound.
- Observations during the test session will be noted.
- In the report, you will be mentioned by a unique test ID, not your name.
- You can abort the test at any time without any specific reason.

By signing you agree to the above terms.

Full Name:	Signature:
Date:	Place:

If you have any questions after today, please contact Alexander Arvebratt at al3818ar-s@student.lund.se or Fredrik Magnusson at fr2231ma-s@student.lu.se.

B.2

Personal Information

Participant ID:

1. Gender:

☐ Male ☐ Other
☐ Female ☐ Prefer not to answer

2. Age: ______ 3. Occupation: ______

4. Are you wearing contact lenses?
☐ Yes
☐ No

5. Do you know what AR is?
☐ Yes
☐ No

6. Have you used AR on your phone before?
☐ Yes
☐ No

7. Have you ever used AR glasses before?
☐ Yes
☐ No

8. Have you ever used eye-tracking before?
☐ Yes
☐ No

9. Do you know what IoT is?
☐ Yes
☐ No

10. Have you ever used IoT before?
☐ Yes
☐ No

11. Do you have any of these smart products at home?
☐ Sonos
☐ Philips Hue
☐ Google Home/Amazon Alexa
☐ None of the above

12. Do you use any of the following apps?
☐ Spotify
☐ Sonos
☐ Philips Hue
☐ None of the above

13. Which mobile operating system do you use?
☐ iOS (Apple)
☐ Android
☐ Other

B.3

Participant ID: _____ Site: ___________________________ Date: ___/___/___

System Usability Scale

Instructions: For each of the following statements, mark one box that best describes your reactions to the website *today*.

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

Please provide any comments about this website:

This questionnaire is based on the System Usability Scale (SUS), which was developed by John Brooke while working at Digital Equipment Corporation.

B.4

Figure 8.6

NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High