

# Developing a telemedical tool for detecting oculomotor dysfunctions

William Starheim Vallestad

Master's thesis in Software Engineering at  
Department of Computing, Mathematics and  
Physics,  
Western Norway University of Applied Sciences  
Department of Informatics,  
University of Bergen  
October 2018



## Abstract

As a lot of children have oculomotor dysfunctions without knowing it, a method of efficient screening will help diagnose those suspected of having problems. As it is expensive and time consuming for visual educators or subjects to travel to a location where the examination can be held, Gunnvor Wilhelmsen wished for a better solution. She therefore suggested to continue development of Ruben Watanabe and Mads G. Eide's CAndLook application, which allows for local examinations[9]. The solution Wilhelmsen wanted was an application which could perform examinations without having the visual educator and subject present at the same location.

This thesis presents an attempt to develop an application which serves this purpose by using CAndLook as a base. The application was developed with a server working as an intermediated and storage device and two clients, one for the visual educator and one for the subject of the examination.

The prototype developed in this projects shows that it is possible for a visual educator and subject to connect to each other and remotely carry out an examination. The results from an examination can also be stored on a remote server for later access and analysis. Although currently being a prototype, the results goes to show that a fully functional telemedical tool for diagnosing oculomotor dysfunction can be developed.

## Acknowledgements

I would like to thank Carsten Helgesen and Atle Geitun for supervising my project and providing me with feedback during this process. I would also like to thank Ruben Watanabe and Mads G. Eide for helping me with technical issues I encountered during my work. Also, thanks to Gunvor Birkeland Wilhelmsen for coming up with the idea for my thesis. I would also like to thank my friend, Ole Kristian Nakken, whom I have discussed ideas with during my work.

## Glossary

Visual educator	Someone studying and aiding people with visual problems
Telemedicine	The use of telecommunications to diagnose and treat subjects
TCP	Transmission Control Protocol, a communications protocol which ensures all data is transferred
CAndLook[9]	The application developed by Watanabe & Eide
TODDT	Telemedical Oculomotor Dysfunction Detection Tool - the name given to the software developed in this thesis
PostgreSQL	An object-relational database management system
Tobii Eye Tracker 4C	An eye tracker developed by Tobii
Oculomotor dysfunction 4C	The absence or defect of controlled eye movements
DigitalOcean	A cloud infrastructure provider
SSH	Secure Shell, a protocol used to securely log onto remote systems
ZeroMQ	Asynchronous messaging library

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Origin of this thesis . . . . .	1
1.2	Motivation . . . . .	1
1.3	Goal . . . . .	2
1.4	Research Questions . . . . .	3
<b>2</b>	<b>Background</b>	<b>4</b>
2.1	Definition and Background of Telemedicine . . . . .	4
2.2	The History of Telemedicine . . . . .	5
2.3	Theoretical background . . . . .	8
2.3.1	Choice of eye tracker . . . . .	9
2.3.2	Choice of database technology . . . . .	10
2.4	Related Work . . . . .	10
<b>3</b>	<b>Problem analysis</b>	<b>12</b>
3.1	Purpose and requirements . . . . .	12
3.2	Basic needs and workings of the application . . . . .	14
<b>4</b>	<b>Design and Implementation</b>	<b>19</b>
4.1	Web application or desktop application . . . . .	19
4.1.1	Web application . . . . .	19
4.1.2	Desktop application . . . . .	21
4.1.3	Summary . . . . .	22
4.2	Components needed . . . . .	24
4.2.1	Eye tracker . . . . .	24
4.2.2	Calibration tool . . . . .	25
4.2.3	Client programs . . . . .	27

4.2.4	Hosting a server . . . . .	29
4.2.5	Communication between clients and server . . . . .	31
4.2.6	Communication between visual educator and subject . . . . .	35
4.2.7	Monitoring of the examination . . . . .	37
4.2.8	Storage of data . . . . .	38
4.2.9	Examining results . . . . .	40
<b>5</b>	<b>Assessment and discussion</b>	<b>41</b>
5.1	Assessment of the project . . . . .	41
5.2	Testing the application on subjects . . . . .	42
<b>6</b>	<b>Conclusion</b>	<b>43</b>
6.1	Research questions . . . . .	43
6.2	Answer to Q1 . . . . .	43
6.3	Answer to Q2 . . . . .	44
6.4	Answer to Q3 . . . . .	44
<b>7</b>	<b>Further Work</b>	<b>46</b>
7.1	Video and audio communication . . . . .	46
7.2	Displaying examinations . . . . .	46
7.3	Being able to choose different examinations . . . . .	46
7.4	Visualizing results after examination . . . . .	47
7.5	Graphical user interface for the visual educator . . . . .	47
7.6	Security . . . . .	47

# 1 Introduction

## 1.1 Origin of this thesis

Gunvor Birkeland Wilhelmsen had the original idea for CAndLook[9], which is a software for detecting oculomotor problems using an inexpensive eye tracker. She later wanted to expand on the idea by continue development of this project into a telemedical solution which could be used to diagnose subjects remotely without having visual educators be on site. As with Watanabe & Eide's project, this thesis is supervised by Carsten Helgesen, Atle Geitung and Gunvor Birkeland Wilhelmsen at the Department of Computing, Mathematics and Physics at Western Norway University of Applied Sciences. Unlike Watanabe & Eide's project, this is a one-person master's thesis.

## 1.2 Motivation

Watanabe & Eide developed a software that was useful for detecting oculomotor problems by using an inexpensive eye tracker. This is a useful tool as, in a study done by Gunvor Wilhelmsen, 35% of children (aged 5-12) examined have some sort of oculomotor problem[30]. However, the current solution of CAndLook has some limitations in how it can be used to examine children on a national or even global scale. As of now, the visual educator has to travel to the subjects location in order to do an examination. This makes it impractical for the visual educator, who has to travel to the subjects, which there could be a large number of in several locations. In conjunction with there, according to Wilhelmsen, being a lack of visual educators, she wanted to continue the work on this project to make the solution more feasible. The idea for the continued development

was then to make a new version of CAndLook that was web based. This would allow for the subject and visual educator to partake in an examination while being in two separate locations.

With a web-based solution, visual educators could be able to run all necessary examinations to diagnose subjects, from their workspace instead of visiting each subject individually. The idea of the web-based solution, is that the only equipment needed at the the subject's location is the inexpensive eye tracker that can either be sent out as required or bought in by institutions, such as schools or municipalities.

There already exists a similar software to CAndLook, which is called Clinical Eye Tracker[26]. This software however, is purely for offline use and does not allow for examinations other than locally on one computer. By making CAndLook into a working web-based application, it should offer something new to the field which does not yet already exists.

### **1.3 Goal**

The goal of this thesis is to develop an Internet based application that allows for remote diagnosing of subjects with oculomotor problems. Oculomotor dysfunction entails a problem with coordinating the muscles in the eye when following an object or reading a text. The goal of this project is to create an application which lets a subject connect to a visual educator from a remote location. Once connected, the visual educator will be able to start an examination of the subject with an eye tracker and different oculomotor tests while communicating with each other during the procedure. Once an examination is completed, the visual educator should have access to the data from the examination, which can be used to diagnose the subject.



Whether this is possible to achieve with this project remains to be seen, and depending on the result, further work on it might be a possibility.

## **1.4 Research Questions**

To reach the goals stated above, a few questions will need to be answered:

1. Can a telemedical application based on CAndLook be developed?
2. How well will such an application be able to remotely start an examination, as well as gather and transmit data?
3. How practical is it to perform examinations when the visual educator and subject are in different places?

## 2 Background

### 2.1 Definition and Background of Telemedicine

Telemedicine is broadly defined as the use of telecommunication for medicinal purposes. The word "telos", comes from Greek and implies distance is part of the context[33]. Telemedicine is defined by the World Health Organization (WHO) to have the following broad definition:

"The delivery of health care services, where distance is a critical factor, by all health care professionals using information and communication technologies for the exchange of valid information for diagnosis, treatment and prevention of disease and injuries, research and evaluation, and for the continuing education of health care providers, all in the interests of advancing the health of individuals and their communities." [29]

This definition was based upon 104 peer-reviewed definition of the term telemedicine. The many definitions used by WHO go to show that telemedicine is still a technology in advancement, which is prone to changes in the still evolving field. One of the biggest advantages of telemedicine is the lower cost of it compared to traditional medicine where subjects would have to be transferred instead of information[33]. It also allows for immediate expert opinion, avoiding the spread of illnesses, and maybe even treatment in places which would otherwise be too desolate for medicinal operations.

Furthermore, the growth of modern technology in the recent decades, has caused all parts of society change. Technological and digital solutions now surround us wherever we go and most of our lives are getting more and

more dependent upon them. As most people living in developed countries now have access to smart phones and smart watches, there could soon be a shift in how we are able to detect, monitor and follow up on health issues. With this change in how the world has gotten more digitalized, it is only natural that the medicinal industry is focusing more and more on developing telemedicine.

## **2.2 The History of Telemedicine**

It is unclear when telemedicine was first used to bridge geographical distances, the concept may have originated centuries ago[33]. There are reports of heliographs, which is a device for reflecting sunlight in a mirror, being used to transmit information about the bubonic plague in Europe during the late Middle Ages [6]. As technology progressed and electrical telegraphs were invented, they were taken in use in The American Civil War, where it was used to transmit information about casualty lists and order medicinal supplies. When Alexander Graham Bell invented the telephone on March 10, 1876, a whole new world of possibilities was opened for the field of telemedicine[19]. The first recorded use of the telephone as a device used in telemedicine was only three years later, in 1879, when a doctor listened to the cough of an infant over the telephone to judge whether it had croup or not. Alexander Graham Bell also applied the newly invented telephone's amplifier to make a functional metal detector[19]. This device was used on Civil War veterans with known lodged bullets, which it was able to detect. In 1881, it was also used on President Garfield which had been shot near the lumbar spine, although doctors were not able to save his life, it shows how technology started playing a part in surgical procedures[19].



Figure 1: Signaling with heliograph near Monument 92. Alaska - Canada border[20].

When World War I started in 1914, radiotechnology had advanced far enough to play a central part in the dynamics of the war and was used extensively for telemedical purposes[6]. In the following World War II and the Korean War, radiotechnology was indispensable to dispatch medical help and helicopters to areas where they were needed.

The first documented use of telemedicine at a hospital was in 1920 at Haukeland University Hospital[6]. Here radiocommunication was used to communicate with ships sailing along the Norwegian coast and provide the sailors with health care services. In 1967 live video communications was for the first time used for telemedical purposes. The link was established between Boston's Logan Airport and Massachusetts General Hospital and was used to exchange x-rays, medical records and laboratory data as well as

remote diagnosis[6]. In the years that followed, the use of videoconferencing technology in telemedicine gained traction and continues to be a popular method of conveying information.

A big part in the more recent development of telemedicine, was done by NASA during the space race between the 1950's to the 1970's. As astronauts in earth orbit would be out of reach of any medicinal personnel on Earth, it was critical to develop the existing telemedicine technology further than it had currently come. When Yuri Gagarin, as the first human ever, was sent to Earth orbit, it was unsure if the human body could even function in space[17]. There was doubt about circulation and respiration functions would be able to operate when not exposed to gravity. To make sure spaceflight was not fatal to humans, both NASA and the Soviet space program sent animals attached to medical systems into Earth orbit. While in space, the biometrical data of the animals were sent back to Earth and monitored by scientists via a telemetric link[17].

In the 90's with robotic technology rapidly advancing, an American robotics company named Computer Motion worked on developing remotely controlled robotic arms, in a project named Automated Endoscopic System for Optimal Positioning (AESOP)[18]. Their work was among others, funded by NASA, which wanted to use this technology to service Space Shuttles in orbit. This was work better suited for machines as it is difficult for humans to perform delicate adjustments in such environments. In 1994 AESOP was approved by Food and Drug Administration in the United States as the first robot for use in surgical procedures. The purpose of AESOP was to maneuver an endoscope inside the subject's body during surgery. A year later, in 1995 Computer Motion presenter their new addition, the ZEUS system which was equipped to use different surgical

instruments. ZEUS was first tested on animals in 1996, and three years later, in 1999, it performed its first surgical procedures on human beings[18].

In 2003 Computer Motion and Intuitive Surgical merged into a single company, whereby work on the ZEUS project was dropped in favor of Intuitive Surgical's da Vinci Surgical System. This system was named after Leonardo da Vinci who did a lot for the early advancements of human anatomy. The merger was of benefit to both companies as they now could combine their efforts of developing a telesurgical solution[22].

As this new technology allowed for remote surgery, the surgeon was no longer required to be at the same location as the subject. Doctors could now be on the other side of the globe and still perform lifesaving surgery. This made it possible for international experts to become available wherever the proper instruments were installed. The potential benefits for this technology is thus measured in the reduced traveling expenses for subjects and specialists, convenience and enhanced performance from the equipment. By providing surgeons with advanced equipment, it has been easier to perform minimally invasive procedures, which offers the benefits of less pain, blood loss and the need for blood transfusion. This again, will in a lot of cases lead to shorter hospital stays and faster recovery for the subject. As of September 30, 2017, 4 271 da Vinci Surgical Systems had been installed worldwide - 2 770 in the United States, 719 in Europe, 561 in Asia and 221 in the rest of the world[27].

### **2.3 Theoretical background**

This part will deal with the solution developed by Watanabe & Eide for their master's thesis. As my project is directly based on theirs, it is important

to go through their work to see the technologies they ended up using and the choices they made. Since I will myself need to decide between different technologies for this project, a good understanding of the ground work done by Watanabe & Eide is vital for my further work.

### **2.3.1 Choice of eye tracker**

In their thesis Watanabe & Eide compared two eye trackers, the TX300 and the EyeX, both which are produced by Tobii Pro[9]. These two eye trackers vary a great deal in pricing, the Tobii TX300 costs \$45 900 according to Watanabe & Eide's thesis while the Tobii EyeX has a much lower price at \$120, although it seems like this product has now been discontinued and replaced by the Tobii Eye Tracker 4C on the Tobii Gaming website[10][11].

The TX300 has the ability to record at 300 Hz compared to the 60Hz of the EyeX, but for the tests done by Watanabe & Eide, the TX300 was used at 60Hz. The testing revealed that the data collected by the TX300 was in general less noisy than with the EyeX. However the difference in the quality of the data collected was not drastic enough to justify the severe difference in pricing of the two eye trackers.

Later in their project Watanabe & Eide changed from the Tobii EyeX to the Tobii Eye Tracker 4C. This new model was not nearly as expensive as the license required to buy the TX300, but is sold at a cost of \$159 according to the website of Tobii Gaming. The Tobii Eye Tracker 4C a frequency of 90 Hz which is a 50% improvement over the EyeX which only has 60 Hz[11]. Another upgrade from the EyeX to the 4C is the inclusion of Tobii EyeChip. According to Tobii, this new technology offers on chip processing and enhanced data streams to host, while also using less power than any other eye tracking device. Tobii are also claiming it is using 1 000 times

less bandwidth when connected to the host, at 0.1 Mbps down from 100 Mbps[28]. Seeing as the Tobii 4C gives an improvement in both efficiency and performance while only being costing \$29 more, it made sense to make the move to the Tobii Eye Tracker 4C.

### **2.3.2 Choice of database technology**

For storing eye recording data, Watanabe & Eide decided to use a database. As they wanted to make their solution as cheap as possible, they only considered open source solutions. They compared two different open source databases, PostgreSQL and InfluxDB[9]. Testing of the two alternatives was divided into two parts, the first part consisted of writing 10 000 samples while the second part consisted of reading the 10 000 samples 10 times. The first test showed that PostgreSQL was faster at writing data using the COPY-command. The second test, which focused on reading data, showed that reading PostgreSQL with PetaPoco was the fastest solution. In the end they ended up storing the data as PostgreSQL-specific statements and read it back as standard SQL data types for increased compatibility.

## **2.4 Related Work**

A similar solution to CAndLook, has been developed by Thomson Software Solutions. Their product *Clinical Eye Tracker*, is described on their website as "a powerful new tool for the clinical assessment of eye movements"[26]. Similarly to CAndLook, Clinical Eye Tracker also uses an eye tracker instrument based on infrared cameras. Thomson Software Solutions claims it can track movements of the eye to less than 1 degree and after a quick calibration, it records the positions of the eye at 60 measurements per second. The total price of the Clinical Eye Tracker is £1 745, which does



not include value-added tax[26].

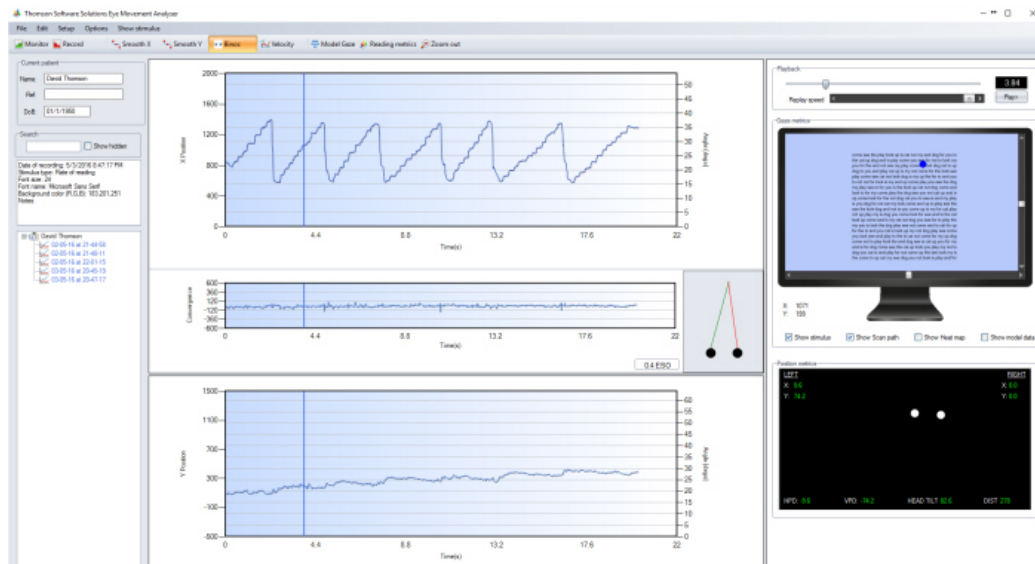


Figure 2: Screenshot of an examination using Clinical Eye Tracker

## 3 Problem analysis

This chapter discusses the requirements of the application and what needs it will fulfill. For this to be done, a detailed analysis of why such an application is needed, has to be established. This is important to further decide how the development will be done, and what components are needed. There will also be the need to decide what technologies will have to be included to develop this project.

### 3.1 Purpose and requirements

The first thing to be decided when work started on the thesis, was what exactly was to be developed. Wilhelmsen had for a time wanted a telemedical solution to CAndLook, and had been in contact with Helgesen and Geitun to try and further develop the idea. According to Wilhelmsen, there is a lack of visual educators compared to the number of people with oculomotor problems, and such an application could help solve this problem. A study done on 86 children by Wilhelmsen in 2012, shows that as many as 35% of children aged 5 to 12 have some form of oculomotor problem, which cause difficulties of different degrees[30]. There have also been similar studies done by others which report numbers between 15% and 25%[13][21]. Oculomotor problems can be hard to detect as children might not themselves know that something is wrong in addition to them having trouble describing what problems they have. Common symptoms of oculomotor problems include headache, fatigue, dizziness, reduced efficiency and productivity, poor attention span and poor academic performance[30]. These are problems which in many cases will affect an individuals self-esteem, as well as their belonging in their respective class environment.

There have been some studies done on treating different kinds of oculomotor problems, which in many cases has shown themselves to be effective to treat[16][4]. Wilhelmsen argues that all children should have to go through at least one screening after starting elementary school to get a broader idea of oculomotor problems in kids. This is also an advantage, as she wishes to start treatment early and prevent children from falling behind academically at an early stage. Presently, the only way children in Norway are thoroughly examined, is if parents, teachers or the healthcare system("Helsetjenesten") suspect that the child might have visual problems. This is in many cases too little, too late.

There are currently both technical and non-technical methods of detecting oculomotor problems. None of these, are however being used in Norway to screen children who are starting their education. The non-technical methods of detecting oculomotor problems are performed by trained experts.

With an application that makes it possible to diagnose and perhaps treat people remotely, the number of people a visual educator can help will increase substantially. This could potentially lead to a big increase in number of children getting the help they need to function better in school from a young age.

The idea to make this happen, is to continue the work Watanabe & Eide has done with their project, which Wilhelmsen, Helgesen and Geitun has deemed an idea worth developing further. The module made by Watanabe & Eide can be further developed for my project in creating a telemedical tool for detecting oculomotor problems. If successful, the application could fill a gap which currently exists in the market.

To my knowledge there are two technical solutions which are moderately affordable. One of these is VisaGraph/Readalyzer, which analyzes eye

movements while the subject is reading a text and gives an overview of the results at the end of the examination[1]. The other solution I am aware of is Thomson Software Solutions' Clinical Eye Tracker. Clinical Eye Tracker works by using a bar with infrared cameras to track the movements of the eyes while the subject is reading a text. It also allows for replaying an examination at any time and see all details. The price of Thomson Software Solutions' Clinical Eye Tracker is £1 745 or £2 094 including VAT[26].

### **3.2 Basic needs and workings of the application**

To develop a telemedical tool, which purpose it is to detect oculomotor problems without a visual educator present with the subject, a few concepts will need to be proved viable. The arguably most vital piece of technology to get working is the transfer of medicinal data in the first place, without this in place the final product will not be a telemedical solution. As the main purpose of the application will be to make examinations over distance possible, it is important to get a functioning prototype that is able to collect and send data between computers by use of the Internet. Data from an examination could both be sent live during the examination, to let the visual educator know the procedure is progressing as it is supposed to, and in its entirety at the end of examination. While it might suffice to only send the data live during the examination, there could be some package loss, causing parts of the data to not arrive at the visual educators computer. Therefor it might be better to also send the data in its complete form at the end, to make sure the visual educator gets it in its entirety.

For the visual educator to be able to start an examination of a subject, both parties will need to connect to each other from their separate locations. This can be done with a remote server working as a link between subject and

visual educator, as well as functioning as a remote storage device for data. Using a server for this purpose, makes it convenient to create an application which will have examinations that will vary in location. A permanent server could also be used as a storage device for previously completed examinations which can be accessed by the visual educator at any time later.

Having two different clients for the application will be of importance, one used by the visual educator and used by the subject. The client used by the visual educator will be more complex and will allow for operations such as setting up, starting and reviewing examinations. The visual educator should be able to select and set up the examinations needed to diagnose the subject. If the visual educator determines the results to be lacking for diagnosing purposes, they should be able to quickly set up a new examination to get better data. Also, if the subject requires a special examination, the visual educator needs to be able to set up a specialized test. The client used by the subject will, on the other hand be kept as simple as possible to make it easy to use and understand. As the subject or counselor does not need to have any understanding of the data collected, there is no need for them to be presented with a complicated client which might confuse and distract them during the examination. Judging whether this way of examining, compared to CAndLook, will work not just in theory, but in practice, will remain to be seen.

Having communication between the visual educator and the subject and counselor will also be an important aspect for the application. If there is no way for the two parts to communicate either before, during or after the examination, misunderstandings can occur which might lead to the examination not being completed as intended. In some cases, the supervisor helping the child during an examination might not be familiar with the

procedure. This is why having continuous communication available for both parts might prove to be necessary for a telemedical solution like this. How the communication will work, is still undecided, but alternatives are text based, video and audio or just audio communications. Text based communications could be implemented as a chat within the application, allowing the subject and visual educator to communicate in a simple way. Audio and video communication could prove harder to implement, but would allow for both parties to both see and hear each other, which might make it easier for the subject to feel comfortable with such an examination. Alternatively, using only audio communication could also be a possibility as it would be simpler to implement and offer many of the same advantages as both video and audio communication. Video and audio conference solutions could be implemented with solutions such as Jitsi, which allows for easy implementation of "Multi-platform open-source video conferencing" according to their website[15]. Another option is using a third party telecommunication platforms such as Skype if there is insufficient time to implement such a solution in the TODDT.

Before an examination starts, TODDT will need to be calibrated to the individual undergoing the examination. This is important as people all have individual characteristics that are important to take into consideration as they could influence the examination data. During the calibration, a model of the users eye is created and includes information about, shape, light refraction and the reflection properties of the different parts of the eye[25]. Tobii has their own software for eye calibration, but as Watanabe & Eide has also developed a software for this, there are two possibilities for what could be used in TODDT.

During the examination, it is an advantage for the visual educator to a be

able to monitor that the examination is progressing according to protocol. If the subject were to make some mistake during the examination, for example look at the wrong part of the screen, the visual educator would quickly know about it and be able prevent and correct the subject. This would also let the visual educator instantly know if the examination would need to be retaken, instead of having to wait until after it has been completed and the subject has disconnected. This could perhaps be implemented such that the visual educator got sent a live video stream of what the subject is seeing at any given time. A drawback of this could be the requirement for better hardware and a stable Internet connection. Another way to monitor the examination, might be to reconstruct the data gathered live at the visual educator's computer, but whether any of these solutions are feasible or not, remains to be seen.

When an examination is concluded, the data collected will need to be sent to the computer of the visual educator via the server, which is acting as an intermediate. The visual educator will then be able to analyze and come up with a diagnosis based of the data from the examination. Analyzing the data could be done in CAndLook, which is well suited for going through examinations.

As this is an application which will have to manage and transfer personal medical data, having security in place will be of importance. This has to be taken into consideration when transferring the data between the computers and when storing the data on a database.

The idea is that the application is going to be used at schools or other public institutions to do screenings of children with suspected oculomotor dysfunctions. Eye trackers will also have to be acquired for the examinations to be carried out. To aid the child during an examination, it is thought that an adult will be present, this could be a school nurse or some else which

will have had training using the application, and know the procedure of an examination.

Based on these needs and requirements Figure 3 was constructed showing the different roles and their responsibilities in the application.

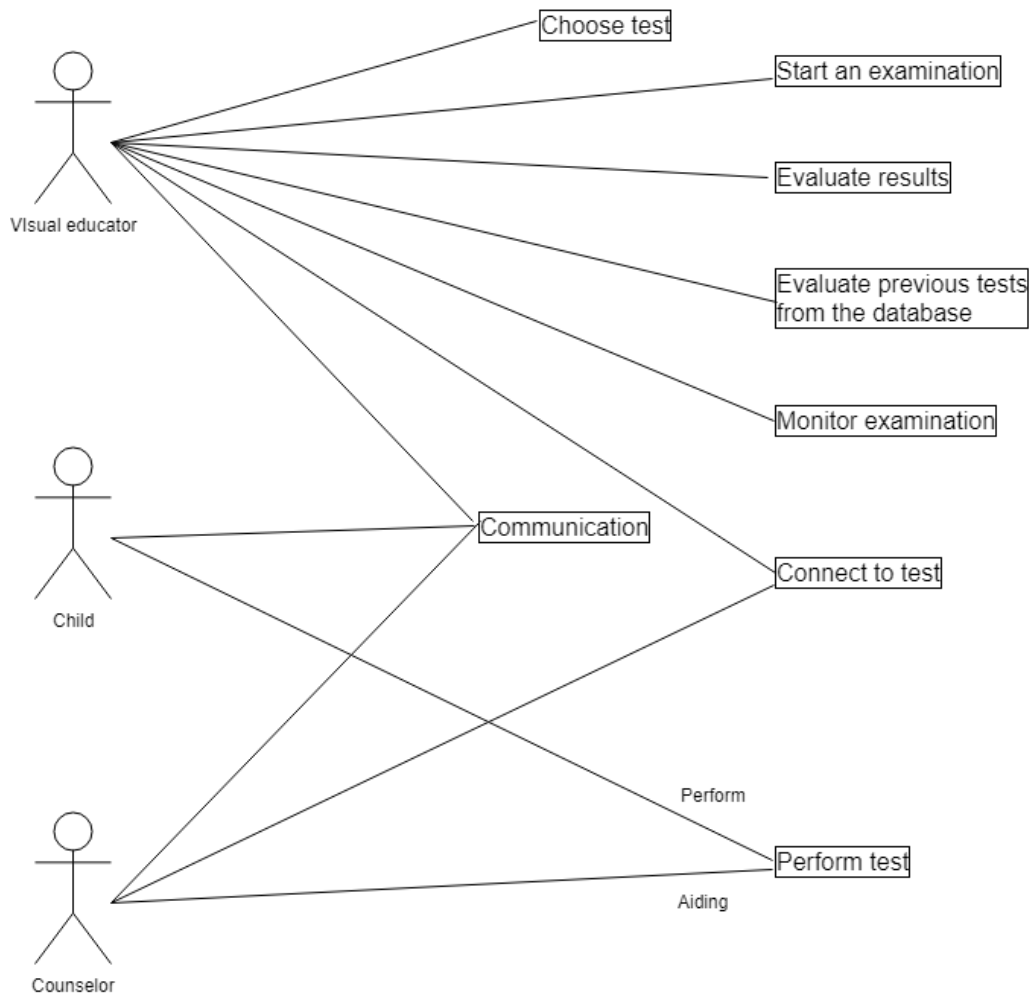


Figure 3: Roles and responsibilities of the application



## **4 Design and Implementation**

While chapter three focused on the needs for a telemedicinal application with the goal of helping diagnose oculomotor problems and what functions it will need, this chapter will deal with the actual design and implementation of the application. The chapter will among other things, discuss what type of application will be best suited for the requirements discussed in the previous chapter. It will also be decided what components and technologies need to be included to make the application work as intended.

### **4.1 Web application or desktop application**

When it comes to the development of the application, a choice has to be made on the architecture. Mainly whether it will be developed as a web application or as a desktop application, both of which have their advantages and disadvantages.

#### **4.1.1 Web application**

A web application would allow for the user to directly access the application from their web browser without having it installed on their computer. Since all the software is based on a server, users does not have to worry about updating their software. Instead all software is updated on the server where it is hosted. Also, the hardware requirements of the computers wanting to use a web application, would not have to be the same as if it were a desktop application. Using thin clients with a web application, could make telemedical examinations of oculomotor problems more accessible to individuals without modern hardware. Even though the web clients themselves are not suited to do complex calculations, the server which they

will be connected to through the web browser, will be.

On the other hand, web applications can be slower and less responsive than desktop applications, especially if the Internet connection is poor. This could in some cases lead to telemedical examinations not be carried out completely. Although, not something I consider a major problem, developing a web application to be compatible with different web browsers is something to keep in mind. Another possible drawback of developing this application as a web application, is the difference in technology from Watanabe & Eide's project and a web application. While .NET does support development of web application with technologies such as ASP.NET, a lot of Watanabe & Eide's existing code will have to be reworked in order to function as a web application. This could prove to be much more time consuming than just continuing work on their existing work and develop it into a telemedicinal solution. As there does not seem to be a way for the web browser to directly access the eye tracker, some additional software might need to be developed to connect them. This could either be a web browser plugin or a .NET application that works as an intermediate between the eye tracker and web application. Either way a web application would be dependent on other software to work as intended.

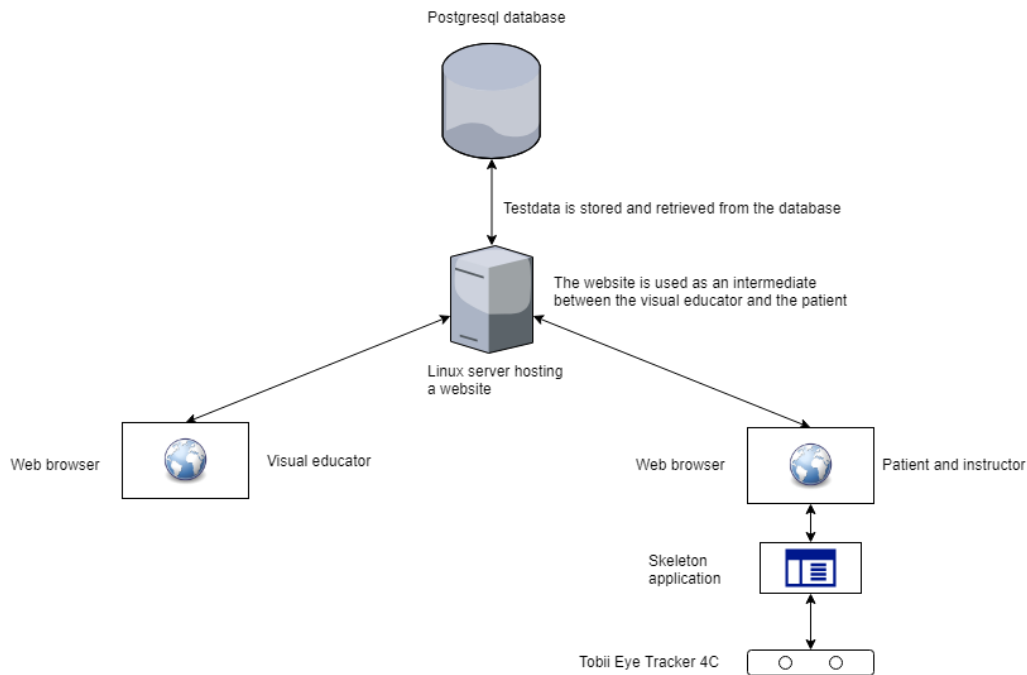


Figure 4: The idea for how a web application would function

#### 4.1.2 Desktop application

One of the most obvious advantages continuing the development of CAndLook as a desktop application, is that it will be more convenient to work on existing code. As getting into a different technology and then also rewriting existing code for it, could end up taking a lot and time and effort. Also, as the software already developed by Watanabe & Eide is fully compatible with the Tobii 4C Eye Tracker, it would not be necessary to develop other modules to communicate with the eye tracker. Regular desktop applications are usually more responsive and less dependent on a fast Internet connection than web applications, even though both are dependent on a stable connection.

As with web applications, desktop applications also have their drawbacks and limitations. First and foremost, desktop applications depend on the

hardware and operating system of the computer they run on. In some cases, this means the computer might not be well suited for tasks the program requires it to do. In contrast, a web application could have a capable server do the heavy operations. In the case of an eye tracking application, this might not be too much of an issue since the calculations done are relatively light. Updates to desktop applications are usually more inconvenient for users as they will themselves have to update the program instead of having it done automatically as with web applications. Users might also, in some cases, be cautious about installing a program they do not know on their computer.

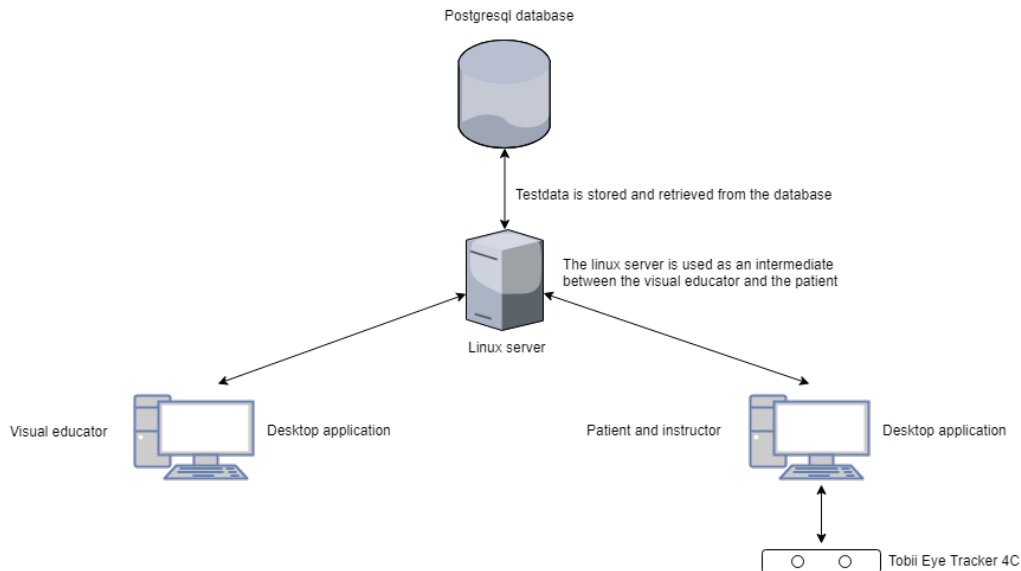


Figure 5: The idea for how a desktop application would function

### 4.1.3 Summary

After having considered both options and looked into the advantages and disadvantages for both, I made the conclusion that a desktop application would be better suited for what was going to be developed. Continuing the work on the existing desktop application is more convenient than starting

over with a web application. It is also already compatible with the Tobii 4C eye tracker which is a big advantage. Another advantage with desktop applications, is the fact that they in many cases can be more responsive than web applications. Although desktop applications might be more limited to hardware and inconvenient to keep updated for users, it is not something I judge to be a significant problem in this prototype.

Seeing as the process of recording data from the eye tracker as well as displaying it is not especially demanding, developing a desktop application is a reasonable choice. It would be a different scenario if the application required demanding calculations. Also, as a web application would require some form of plugin or program to transmit data between the eye tracker and the application. This means, the advantage of web applications not needing any additional modules to work, is lost. Seeing as Watanabe & Eide's code already is fully functional with the Tobii Eye Trackers it will be much more convenient using their existing technologies.

Not being dependent on having software already installed is one of the major perks of using a web application. Some people might not be comfortable with installing program on their computers, but this was judged to be a minor issue.

Having a responsive application is of great importance as the examinations need to be accurate to measure oculomotor problems. If the data were to be sent to a remote server for processing, a poor Internet connection could compromise the results, leaving them inaccurate. This is why having a desktop application, which does all the work locally, is a better choice in this regard.

As Watanabe & Eide developed their project as a .NET desktop application, it was natural to continue on their work in the framework they

had been using. As I am already familiar with C# and .NET both from previous courses and studying Watanabe & Eide's work. Getting into a new technology and rewriting large portions of their code to work as a web application could prove to be time consuming and lead to getting a working prototype not being completed in time.

These reasons combined, is why I decided to develop a desktop application instead of a web application. As this is merely a prototype, making sure I end up with a result which shows the viability of the concept is the main goal. For me this seems to be best achieved by developing a desktop application.

## **4.2 Components needed**

To make the project which have been described, several different technologies and components will be used and implemented in this solution. This part of the chapter will discuss the design and the technical details of how the desktop application will be developed.

### **4.2.1 Eye tracker**

Watanabe & Eide started their project with the Tobii EyeX eye tracker, but later moved on to the newer Tobii Eye Tracker 4C. This model provides increased performance, reduced power consumption and uses less bandwidth than the EyeX. With native Windows 10 support, it is easy to set up and use by simply connecting it with a USB. According to Tobii, it also works on 97% of the population, which makes it the most widely compatible eye tracker on the market[11]. This is an improvement of 2% from Tobii EyeX.

Since they moved on to the Tobii Eye Tracker 4C, it was only natural to continue using this eye tracker instead of the older EyeX. At the start of this project, a Tobii 4C was lent to me by Watanabe & Eide.



Figure 6: The Tobii Eye Tracker 4C product[11].

#### 4.2.2 Calibration tool

The first step of any oculomotor examination usually consists of calibrating the application to the individual performing the examination. Being able to accurately detect where the participant is looking at any given time is crucial to get optimal results. As previously mentioned, a calibration procedure, consists of using the eye tracker to measure the characteristics of the user's eyes in addition to create a 3D model of the eye. This information is used to calculate the gaze points of a person looking at the screen. A gaze point is the location of the screen the eye tracker perceives the participant is looking at

any given time. Before a calibration, the gaze points could be off from where the participant is actually looking. After calibrating, recordings of gaze data are usually much more accurate as shown in Figure 7. The red dots represent where the eye tracer perceives the gaze points are before calibration and the green dots represent the gaze points after calibration.

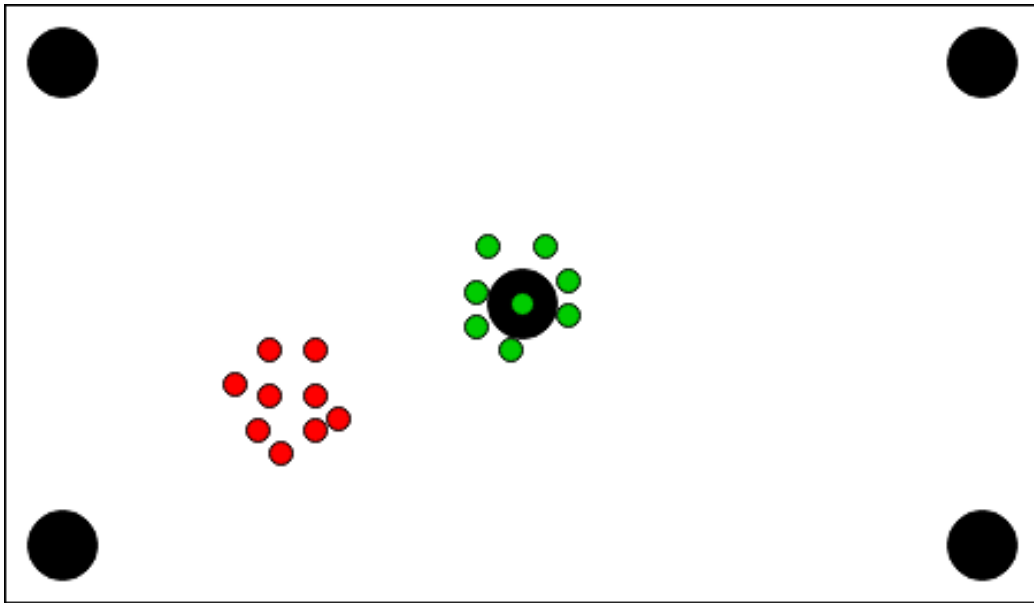


Figure 7: Gaze points before and after calibration.

There are two alternatives when it comes to calibrating participants for the examination. One is using the default solution from Tobii , which can be downloaded from Tobii Gaming’s website[12]. Once downloaded and set up, starting a calibration is straightforward and gives an accurate profile for the user. With the calibration completed, the user can then use the eye tracker for whatever purpose they require, be it an oculomotor examination, gaming or other needs. A downside of using Tobii Gaming’s calibration tool for this project purpose, is the fact that it requires a separate program in addition to the one developed in this thesis, which could prove to be inconvenient for



users.

The other alternative is the calibration tool developed by Watanabe & Eide for their project. This solution is built into CAndLook[9] which makes it possible to reuse for TODDT instead of using a third party program for calibration. The reason Watanabe & Eide implemented their own calibration, was to try out and compare different calibration algorithms. The best option would be to use the calibration tool of CAndLook, so TODDT would not have to rely on other software. I was, however not able to implement this in time, so TODDT will rely on Tobii Gaming's calibration program.

#### **4.2.3 Client programs**

Both the visual educator and the subject will need to have their separate clients in order to initiate and carry out an examination. As previously discussed, the clients will need to be different in order to serve the needs of the visual educator and subject using the program.

The visual educator should be able to choose a specific examination for the subject to perform, evaluate results, connect to and monitor the examination, as well as communicate with the subject and counselor. Ideally, having a graphical user interface would be optimal for all of these purposes, as it would make navigation and execution clear and intuitive for most people. Since CAndLook[9] supports examinations and the visualization of data, I decided to base the development of my clients on it.

I decided to develop the client prototypes as console applications as it would be easier to set up and would give an idea as to how feasible the concept is. This entails that the visual educator will have to type commands into a command window in order to execute operations. This will include commands for starting, choosing and evaluating an examination.

The client for the subject does not need the same level of complexity as the one for the visual educator. This is because they will only need to connect and perform the examination as well as communicate with the visual educator. Even so, having a GUI to show simple information such as connection to the visual educator and status of the eye tracker would be the preferred option for a finished product. With only a simple user interface needed for the prototype, it is enough to display a window when connecting to the visual educator, giving information about the ensuing examination. When the visual educator decides to start, the examination will display in the subjects client and proceed until it ends or is canceled. When the examination is over, the visual educator could decide to redo the examination or some other variant if needed, or in the case no further examination is required, shut down the connection. With the examination over, the examination will close for the subject, displaying a message saying the examination has been completed.

Early in the process of the project, I planned on having the clients communicate with each other directly, but in the end, I decided on using a server as an intermediate. Having the clients connect directly to each other could be inconvenient as examinations would take place in different locations in most cases. This would require the users to find and change the IP they would need to connect to, which could cause a nuisance. Instead, connecting to a server with a fixed IP address, would remove this problem entirely. Having a server would also remove the need for storing data at the visual educator's computer, instead having it accessible at a remote server at any time and convenience.

Communication between the parties will work similarly for both clients, regardless of chat or video communication. With the implementation of a

text based chat, both parties will have a command window dedicated to chat communication open up once a connection between them has been established. The idea is the same for audio and video communications with having separate windows dedicated to them being opened up with connection established. However, as the time remaining for this project was lacking, it was instead determined that a text based chat would be implemented instead.

In the early planning phase, I considered having the server take care of the processing of the data, but this was quickly ruled out. There was no need for the server to handle the examination as it could be done on the client. The examination will be carried out locally on the subjects computer with the data being transmitted to the server at the conclusion of the examination. Seeing as the examination itself is not particularly hardware demanding, as it consists of tracking coordinates, this should not be an issue.

#### **4.2.4 Hosting a server**

In order to set up a server, a computer was needed to run it on. I decided to use a cloud infrastructure service for hosting purposes, as they are inexpensive and allows for easy setup and configuration. Infrastructure as a Service(IaaS) is type of cloud computing which gives the user instant access to virtualized computing resources over the Internet. Services like these typically provides an easily scalable environment with high uptime, good Internet connection, backup of all your files and the benefit of only paying for the services you use. This way you can also avoid investing in infrastructure which you might not end up not needing after a while. There are a multitude of companies who provide this service, among them Amazon Web Services, Microsoft Azure and Digital Ocean. Before choosing

which service to use for my project a few of them was considered to see how they would compare price wise and technically.

Seeing as Amazon Web Services, Microsoft Azure, Digital Ocean and Google Cloud Platform all have some sort of free trial, price was not taken too much into consideration. Having looked through setup tutorials for each service the conclusion was made that what none of the services stood out in terms of advantages for my project. I decided to go with DigitalOcean as it seemed like a simple and straightforward service with easily scalable modules. DigitalOcean also offers an extensive community forum with guides and other types of help for a variety of subjects.

Microsoft Azure	\$200 credits to be used within 30 days and 12 months of use[3].
Google Cloud Platform	\$300 free credit during a 12 month time period[5].
DigitalOcean	\$50 credit through GitHub promo code[8].
Amazon Web Services	12 months free trial with limitations[2].

Table 1: List of IaaS providers.

Setting up a server with DigitalOcean was quickly done once an account had been set up. I decided to use Ubuntu as the operating system and selected the specifications for the computer using the Droplet modules. Droplets lets the users themselves decide the amount of memory, storage and processing power of the virtual machine depending on their needs. I decided to use one of their cheaper options, which includes 2GB of memory, one vCPU(of unspecified capacity) and a 50GB SSD for \$10 per month. With the \$50 included in their free student pack, this would let me host a server for 5 months, enough time to complete this thesis.

Once the virtual machine was set up, a secure connection to it needed to be established. Even though DigitalOcean offers a command prompt to your virtual machine on their website, I decided to set up a SSH connection using PuTTY. This was because the in browser command prompt offered by DigitalOcean was perceived as slow and unresponsive when experimented with. Once the SSH connection was established through PuTTY, it was possible to sign into the virtual machine as root and the computer was ready for use.

#### **4.2.5 Communication between clients and server**

Choosing the correct communication protocol for the project is important to ensure the different components are able to communicate. For this reason, I decided to use Transmission Control Protocol(TCP) as the communication protocol. TCP is a transfer protocol that ensures data has been transferred between clients, in comparison to User Datagram Protocol(UDP) which offers no such guarantee[23]. Certainty of the delivery of data packages is important to this project as both the visual educator and subject will need confirmation that both parties are connected. This is also essential as the visual educator will need confirmation of both the start and completion of an examination.

Developing a TCP messaging library for this project from scratch is both time consuming and unnecessary as there are multiple messaging libraries available. For my purpose, I decided to use the asynchronous messaging library ZeroMQ which serves the needs of this project. ZeroMQ is a socket based distributed messaging system which supports a multitude of languages, among them Python, C++, Java and .NET. ZeroMQ provides a socket structure with several different patterns for different uses[32].

- Request-reply - used for sending requests from a client to one or more

services, and receiving a subsequent reply to each request sent.

- Publish-subscribe - used for one-to-many distribution, from a single publisher to multiple subscribers.
- Pipeline pattern - data is transmitted down a pipeline of nodes, where each node is connected to at least one other node.
- Exclusive pair pattern - each node is connected to exactly one other node.

For the needs of the project, I decided to use the request-reply pattern. This choice was made as the client programs for both the visual educator and subject will act as clients requesting to connect to the server. Once connected, the communications between the client and the server will be in a request and reply format, with the server relaying information between the two clients.

At first, a simple C# console application was set up to test the functionality of ZeroMQ. To get access to the ZeroMQ library, the .NET implementation of ZeroMQ was required. It is called NetMQ and was acquired through the Package Manager Console in Visual Studio. The program consisted of one request and one reply socket. The the request socket sent a message to the reply socket, which then again sent a reply back to the request socket. The prototype was then expanded on by having the client communicate with the DigitalOcean Linux server. The C# program was modified to only have a request socket which was connected to the IP address of the server. On the server, a python server program was running, which had a reply socket that would receive a message and print it on the local computer. It would then send a confirmation message back to

the request socket. When the C# client program received the reply back from the reply socket on the server, it printed out the message.

As I wanted to make sure the server could act as an intermediate between two separate clients, one additional client was then added to the prototype. This confirmed that the test clients of the visual educator and subject were able to communicate and could later be used to implement a telemedical examination. Figure 8 shows how the clients and server will communicate during an examination.

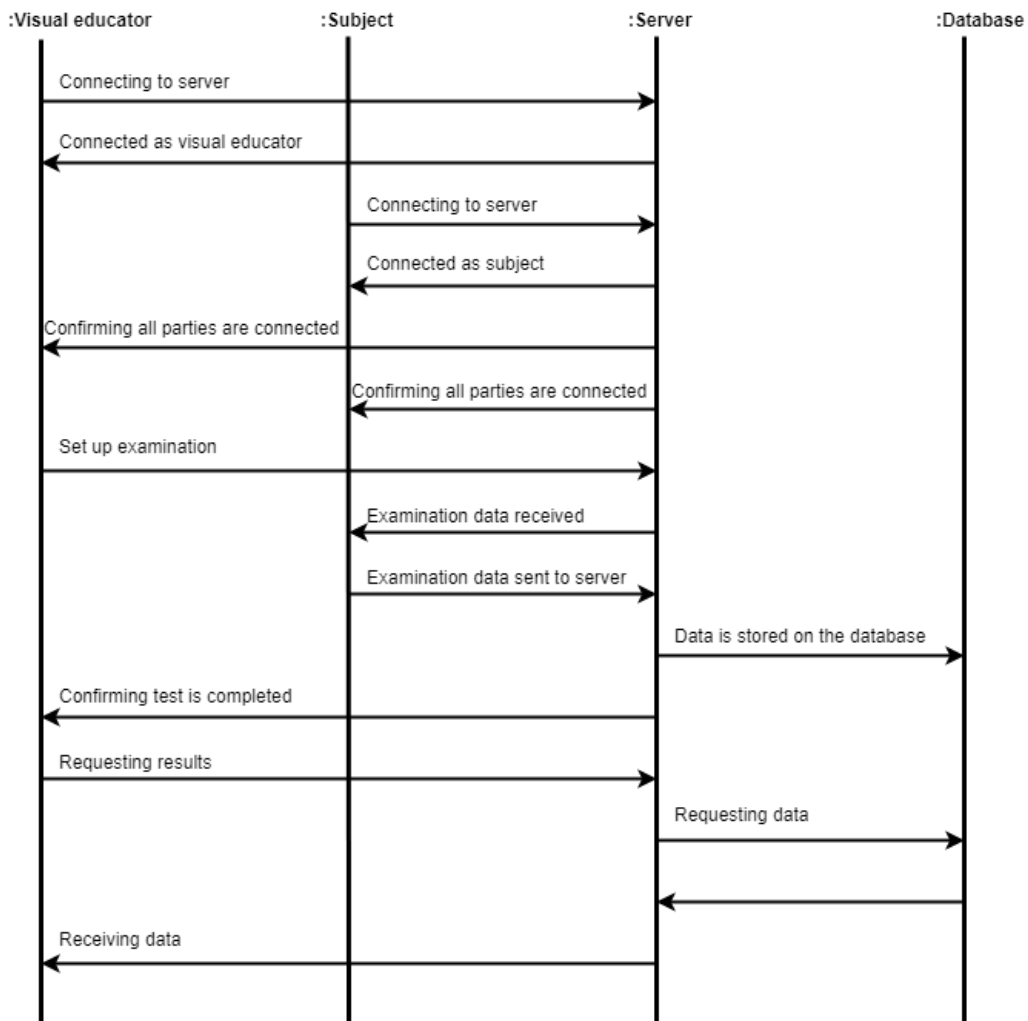


Figure 8: Sequence diagram of the application

Both the visual educator and the subject's client will upon starting send a signal to the server, requesting to connect with it. With both clients connected to the server, it will send a reply to both of them, informing that all parties are connected and ready to proceed. The client program for the visual educator will then have the option to start an examination. When the visual educator executes the command to start the examination, a message is sent to the server, which again relays it to the client of the subject. Once



the message has been received, the client will call on CAndLook to start an examination which will continue until it is completed or canceled. Upon completion, the data from the examination is sent to the server, which again informs the visual educator about the completion of the examination. As for now, the visual educator does not have the option to analyze the data in the client, but this is something which can be implemented in the future. It is, however possible to access the data on the server by use of PgAdmin and import it in CAndLook to be analyzed.

#### **4.2.6 Communication between visual educator and subject**

With communication between the parties involved in the examination being important, different ways of implementing it was considered. Both text based communications, video and audio are options that would work. Video and audio communications in some cases allows for a richer form for communication compared to communication methods like a text based chat[7]. However, there has been done a study on the preference of text based to audio communication[14]. In certain situations people seem to prefer communicating using chat to audio as it can have the following advantages:

- Allows for asynchronous communication.
- The "cost" of interrupting others is lower.
- It can help overcome audio problems in public places.
- Can make it easier to communicate in a second language

While these are valid points to make, oculomotor examinations are not meant to be performed in public. Regardless of what is the most optimal

solution for this application, I had to go for the one that took the least time to develop. That is why I decided to develop a chat as it was judged to be quicker to implement.

For the implementation of the chat, ZeroMQ was once again chosen as the framework, as it has worked well for communication implementation previous in the thesis. The chat application will use the same basic communications architecture as the clients by having the server as intermediate. This is why developing chat communication would be more convenient with the time available. The request - reply socket pattern of ZeroMQ was used to develop the chat. As shown in Figure 9, both client programs are similar and are running two threads. One thread is actively listening for incoming messages forwarded from the server, while the other is running a while loop waiting for text input. On the server side a python program was set running two threads. One thread is listening to messages from the visual educator and forwarding them to the subject, while the other thread is directing messages the opposite way.

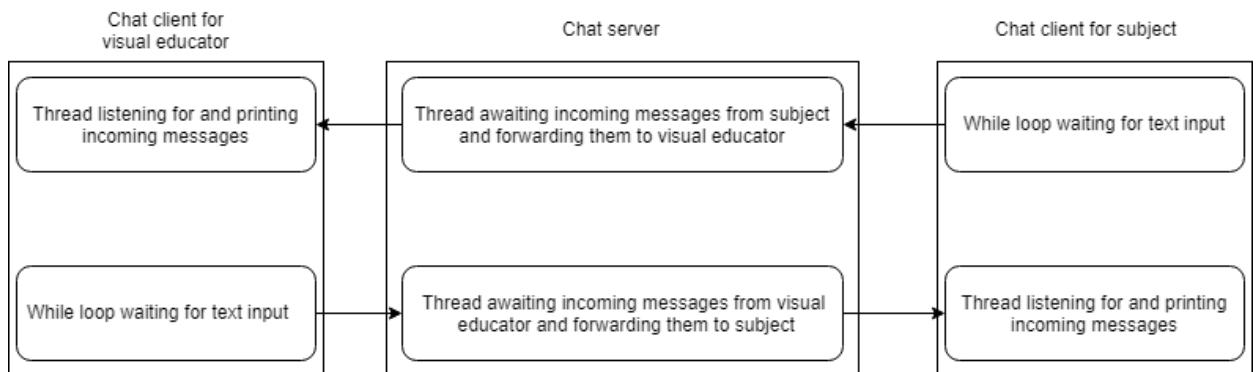


Figure 9: Diagram of the chat implementation.

#### 4.2.7 Monitoring of the examination

Being able to monitor the examination in real time is beneficial to ensure it is carried out like intended. Without any way for the visual educator to know whether or not the subject is undergoing the examination as instructed, the data collected could be flawed and of no use in the evaluation of a diagnosis. The visual educator would then have to wait for the results and then decide to do the examination all over. This would be impractical and favourable to avoid if possible. For example, the subject's counselor could have made a mistake placing the eye tracker, which would lead to it not being able to track the eyes correctly. Other things that could happen, might be that the counselor stopped the examination before its conclusion, which would mean not all the data would be collected.

Because of this reasoning, I deemed critical for a fully functioning product to have this feature integrated. I was however, not able to implement this feature in the prototype in time. This would rather be something worth looking into for the future, should work on this project continue.

There are a few ways this could be implemented. Using screen sharing to monitor how the examination is proceeding in real time by seeing the content of the subjects screen. This however, would probably require more of the hardware and Internet connection of the computers involved. A solution that would probably be better suited for TODDT, would be to simply transmit the recorded data in real time to the visual educator via the server. In this case, the results could be visually recreated on the visual educators computer in real time. With this solution, the visual educator could also customize what data is displayed at any given time depending on the the need.

Another potential helpful way for the visual educator to monitor the examination, could be with the help of a web camera. This could let the

visual educator see and perhaps guide the user in some cases. It is not clear how much of help a tool like this would provide, but it is worth considering for the future.

#### 4.2.8 Storage of data

For simplicity's sake, PostgreSQL was used as Watanabe & Eide used it in their project. Their solution was to store data locally on the computer running the examination. For this project, I instead decided to store the data remotely on the server. The reason for this, is the advantage of having data from an examination available to the visual educator. By having it on a server, it will be available on any computer the visual educator is on, both after the examination and at any later time.

In order to set up the database on the server PostgreSQL needed to be installed. As Ubuntu contains Postgres packages, this could be done using the apt packaging system on Linux. To interact with databases in PostgreSQL, a role with privileges has to be set up. This is in order to manage permissions and ownership to specific databases[24].

As the database needed to have the same structure as the one used in CAndLook, a way of transferring their existing database structure to the server was needed. For this purpose Windows Secure Copy(WinSCP) was used, which is a free and open-source secure copy protocol client for Windows. This tool allows for file transfer between a local and a remote computer, and is based on the implementation of the SSH protocol from PuTTY[31]. In order to transfer the database dump to the server, WinSCP was used to securely copy the file over to the DigitalOcean server. In order to restore a dump file to a database, it has to be restored into an empty database. Once the structure was restored, the database has to be

configured for remote accessibility. For that to be done postgresql.conf and pg\_hba.conf had to be configured. Changing the listen\_address from 'localhost' to '\*' in postgresql.conf as seen in Figure 10, lets PostgreSQL listen to all IP addresses. Adding the line shown in Figure 11 to pg\_hba.conf, allows all IP addresses to establish a connection with the database. With the files configured, restarting the PostgreSQL is needed for the changes to take effect. As the virtual machine also has a firewall in place, the ports used by PostgreSQL had to be opened in order for remote computers to get access.

```
listen_addresses = '*'                                # what IP address(es) to listen on;
```

Figure 10: postgresql.conf

```
host    all             all             0.0.0.0/0      trust
```

Figure 11: pg\_hba.conf

With the database set up, it had to be decided how to transfer and save the data. As the examination data in Watanabe & Eide's application is stored as a custom made JSON object, I considered transferring the object to the server program using ZeroMQ. The JSON objects would have to be serialized before transmitted and then reconstructed on the server as ZeroMQ does not support transfer of custom made objects. This idea, was however discarded in favour of the simpler solution of changing the path of the database in the source code of CAndLook. This consisted of editing the database path from "localhost" to the IP address of the DigitalOcean server.

#### **4.2.9 Examining results**

Being able to analyse the results of an examination is crucial in order to give a proper diagnosis. As the raw data recorded are just coordinates on a screen, it is better to be able to visualize the results graphically. This was implemented in CAndLook, and gave an informative view of the data from an examination. It was initially meant to be implemented in this project too, but I was not able to set up a prototype for it in time. However, as the database structure of this project is identical to the one used in CAndLook, the data from an examination can be imported to CAndLook and visualized there for diagnosis.

## 5 Assessment and discussion

This chapter will focus on the work done in this thesis and what could have been done better.

### 5.1 Assessment of the project

At the start of the thesis, I set out to create a telemedical application based on Watanabe & Eide's CAndLook[9] implementation. The original plan was to create an application which had the functionality described in Figure 3. In the end some of these functionalities were included, while others were not or only partially implemented. The functions not implemented are being able to choose a particular examination, saving examination data, evaluate current or previous results and being able to monitor an ongoing examination. The reason for them not being included, is because I was not able to finish them in time. While necessary for a finished product, these features are not crucial to decide whether this works as a proof of concept. The features implemented are:

- Both parties are able to connect to each other before an examination.
- Chat communication between the two parties.
- The possibility for the visual educator to start an examination.
- The performing of an examination by the subject, although the animations of an examination are not visualized.

While not something adding to the user experience of the product, secure handling of personal medical data is lacking from TODDT. This is something I would have spent more time focusing on in this project. As

this is a prototype, a finished product was not expected, and with further work, the rest of the features can be implemented. Since the original goal was to create a prototype, it was expected that it would not end up as a finished product by the end of the thesis.

## **5.2 Testing the application on subjects**

Even though the Telemedical Oculomotor Dysfunction Detection Tool is just a proof of concept, judging its functionality through real world trials is a good indicator on its usefulness. This would have been done by acquiring volunteers to act as subjects during tryouts of the application. By having the subjects perform an examination while communicating using the built in chat, it would have been possible to gain an idea of the functionality of the prototype.

Collecting data on how potential test subjects experience TODDT, could also be used in further development of the project. In order to develop the best possible product, taking user experiences into consideration is of importance.



## 6 Conclusion

This chapter will attempt to answer the research questions asked at the start of this thesis.

### 6.1 Research questions

This section repeats the questions and attempts to answer them.

1. Can a telemedical application based on CAndLook be developed?
2. How well will such an application be able to remotely start an examination, as well as gather and transmit data?
3. How practical is it to perform examinations when the visual educator and subject are in different places?

### 6.2 Answer to Q1

*Can a telemedical application based on CAndLook be developed?*

Using the core functionality of CAndLook it is possible to create an application which performs an eye screening remotely. A built in calibration tool was not implemented, but with the use of Tobii Gaming's calibration software, accurate calibration of an individual's eyes are still achieved. Even though the visual aspect of the examination is not displayed, recording of eye movements is started from a remote location. The process of implementation is described in chapter 4.

Based on my findings in this thesis, a telemedical solution to CAndLook is something I consider practical to achieve.

### **6.3 Answer to Q2**

*How well will such an application be able to remotely start an examination, as well as gather and transmit data?*

When running the two clients in the application, they instantly establish a connection to the server, which again informs both clients they are connected. The chat window also opens in both clients, which lets the subject and visual educator communicate between each other. As the connection between the subject and visual educator client is established, the visual educator has the option of starting an examination. During the examination, the data is temporarily stored on the subjects computer. At the end of the examination, the data is sent to the server which attempts to store it on the PostgreSQL database.

Establishing a connection between the server and clients, as well as starting a chat, was a success. So was starting an examination remotely, but when the subjects client tried to save the data on the server, an error occurred. I was not able to correct this in time, but I suspect it is due to inconsistent naming of database tables. With further work fixing this error, TODDT will be able to perform telemedical oculomotor dysfunction examinations.

### **6.4 Answer to Q3**

*How practical is it to perform examinations when the visual educator and subject are in different places?*

This question remains to be answered as the prototype developed during the project does not allow for an actual examination, only the tracking of

eye movements. That meant, a remote oculomotor examination on subjects in order to try to answer this question, was not possible. Had this been implemented, it would however have been possible to test the functionality of a telemedical oculomotor dysfunction examination.

## **7 Further Work**

This chapter will discuss ideas which I was not able to implement in the project due to a variety of reasons, but which are interesting for further development.

### **7.1 Video and audio communication**

Even though both the visual educator and subject are able to communicate by text chat during the examination, I believe it could be an advantage to have audio communications too. There exists frameworks for implementing this, such as Jitsi, which have been mentioned previously in this thesis. Using third party programs such as Skype, could also be a temporary alternative.

### **7.2 Displaying examinations**

As of now, the visuals of an examination are not being displayed to the subject. This is a must have feature in a final product, as the subject needs to be able to see the instructions and objects on the screen. While data is being collected during an examination, the subject cannot see the tests displayed, thus no meaningful data is recorded. Having an animation or reading test displayed, TODDT would be able to collect data which could be used by the visual educator.

### **7.3 Being able to choose different examinations**

Another important implementation is for the visual educator to be able to select between different examinations for the subject to undertake. Different examinations can be used to test different problems, and is crucial to have for a complete product.

## **7.4 Visualizing results after examination**

For the visual educator to evaluate an examination, they need to be able to get a visualization of the test data. As for now it is only possible to access the raw data in the database, without it being presented in a meaningful way. It is, however possible to import the collected data during an examination into Watanabe & Eide's CAndLook application for visualization and analysis. Being dependent on another application, is however not optimal, so having this function built into TODDT would be a better long term solution.

## **7.5 Graphical user interface for the visual educator**

The feature mentioned in the aforementioned section would require a graphical user interface(GUI) to function. As of now the visual educator only has a console application to disposal, which limits the information that can be presented. A GUI would also allow for more options to be displayed in a more intuitive and transparent way. With this in mind, for future work on this application, implementing a GUI, would be a big advantage.

## **7.6 Security**

As the TODDT will manage and transfer medical data, it is important to ensure the said data is being handled correctly. This has not been implemented in the prototype, but needs to be featured in a finished product. Data handling will need to be in in compliance with the regulations of the norwegian *Datatilsynet* or similar jurisdictions in other countries the application may be used.

## List of Figures

1	Signaling with heliograph near Monument 92. Alaska - Canada border[20]. . . . .	6
2	Screenshot of an examination using Clinical Eye Tracker . . . . .	11
3	Roles and responsibilities of the application . . . . .	18
4	The idea for how a web application would function . . . . .	21
5	The idea for how a desktop application would function . . . . .	22
6	The Tobii Eye Tracker 4C product[11]. . . . .	25
7	Gaze points before and after calibration. . . . .	26
8	Sequence diagram of the application . . . . .	34
9	Diagram of the chat implementation. . . . .	36
10	postgresql.conf . . . . .	39
11	pg_hba.conf . . . . .	39

## List of Tables

1	List of IaaS providers. . . . .	30
---	---------------------------------	----

## References

- [1] COMPEVO AB. Compevo ab., [Accessed September. 6, 2018]. URL: <http://www.compevo.se/news.php>.
- [2] Amazone. Aws pricing, [Accessed September. 13, 2018]. URL: [https://aws.amazon.com/pricing/?nc2=h\\_ql\\_pr](https://aws.amazon.com/pricing/?nc2=h_ql_pr).
- [3] Microsoft Azure. Create your azure free account today, [Accessed September. 12, 2018]. URL: <https://azure.microsoft.com/en-us/free/>.
- [4] K. J. CIUFFREDA. The scientific basis for and efficacy of optometric vision therapy in nonstrabismic accommodative and vergence disorders. *OPTOMETRY*, 73:735–762, 2002.
- [5] Google Cloud. Do more for less, [Accessed September. 12, 2018]. URL: <https://cloud.google.com/pricing/>.
- [6] P. S. Pandian D. T. Shakuntala Anuradha Srinivas Uma Maheshwari V. M. Mohan Velu K Madhabi Sarkar Safeer K. P. Pragati Gupta Dr. B. S. Sundersheshu Dr. V. C. Padaki Deepa Prabhu, Sumathi S. R. A review of telemedicine in defence scenario. URL: [https://www.researchgate.net/publication/315452825\\_A\\_Review\\_of\\_Telemedicine\\_in\\_Defence\\_Scenario](https://www.researchgate.net/publication/315452825_A_Review_of_Telemedicine_in_Defence_Scenario).
- [7] & Valacich J. Dennis, A. Rethinking media richness: Towards a theory of media synchronicity. 1990.
- [8] DigitalOcean. Simple, predictable pricing, [Accessed September. 13, 2018]. URL: <https://www.digitalocean.com/pricing/>.



- [9] Ruben Watanabe & Mads G. Eide. Detecting oculomotor problems in children using eye tracking. 2017.
- [10] Tobii Gaming. Tobii eyex, [Accessed April. 23, 2018]. URL: <https://tobiigaming.com/product/tobii-eyex/>.
- [11] Tobii Gaming. Tobii eye tracker 4c, [Accessed August. 21, 2018]. URL: <https://tobiigaming.com/product/tobii-eye-tracker-4c/>.
- [12] Tobii Gaming. Eye tracking core software, [Accessed September. 15, 2018]. URL: <https://tobiigaming.com/getstarted/>.
- [13] J. 2004 HEIM. Sluttrapport fra prosjektet.
- [14] Rikard Harr Jeremiah Scholl, John McCarthy. A comparison of chat and audio in media rich environments. *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, pages 323–332, 2006.
- [15] Jitsi. Multi-platform open-source video conferencing, [Accessed July. 31, 2018]. URL: <https://jitsi.org/>.
- [16] MASTER C. L. MESSNER L. V. PANG Y. SMITH C. & STARLING A. J. LEONG, D. F. The effect of saccadic training on early reading fluency. *Clinical Pediatrics*, 53:858–864, 2014.
- [17] NASA. A brief history of nasa’s contributions to telemedicine., [Accessed April. 22, 2018]. URL: <https://www.nasa.gov/content/a-brief-history-of-nasa-s-contributions-to-telemedicine/>.
- [18] NASA. Robotic surgery., [Accessed April. 23, 2018]. URL: <https://spinoff.nasa.gov/spinoff2000/hm1.htm>.

- [19] PhD Nicholas Genes, MD. Alexander graham bell and the birth of telemedicine, [Accessed April. 10, 2018]. URL: <http://www.telemedmag.com/article/alexander-graham-bell-and-the-birth-of-telemedicine/>.
- [20] NOAA. Signaling with heliograph near monument 92. alaska - canada border. international boundary commission. URL: <http://www.photolib.noaa.gov/htmls/theb1633.htm>.
- [21] A. 2007 ORFIELD. Eyes for learning: Preventing and curing vision-related learning problems, [Accessed June. 3, 2018]. URL: <https://eric.ed.gov/?id=ED497320>.
- [22] A. 2007 ORFIELD. Agreement and plan of merger by and among intuitive surgical, inc., iron acquisition corporation and computer motion, inc. dated as of march 7, 2003, [Accessed September. 15, 2018]. URL: <http://edgar.secdatabase.com/2339/89161803001169/filing-main.htm>.
- [23] C. Pakanati, M. Padmavathamma, and N. R. Reddy. Performance comparison of tcp, udp, and tfrc in wired networks. In *2015 IEEE International Conference on Computational Intelligence Communication Technology*, pages 257–263, Feb 2015.
- [24] PostgreSQL. Chapter 21. database roles, [Accessed September. 26, 2018]. URL: <https://www.postgresql.org/docs/10/static/user-manag.html>.
- [25] Tobii Pro. What happens during the eye tracker calibration, [Accessed August. 20, 2018]. URL: <https://www.tobiiipro.com>.

com/learn-and-support/learn/eye-tracking-essentials/  
what-happens-during-the-eye-tracker-calibration/.

- [26] Thomson Software Solutions. Clinical eye tracker, [Accessed April. 23, 2018]. URL: <http://www.thomson-software-solutions.com/clinical-eye-tracker/>.
- [27] Intuitive Surgical. Investor faq. URL: <http://phx.corporate-ir.net/phoenix.zhtml?c=122359&p=irol-faq>.
- [28] Tobii. Components & system design for is4, [Accessed August. 21, 2018]. URL: <https://www.tobii.com/tech/products/components-system-design/>.
- [29] WHO. Telemedicine, opportunities and developments in member states., [Accessed April. 20, 2018]. URL: [http://www.who.int/goe/publications/goe\\_telemedicine\\_2010.pdf](http://www.who.int/goe/publications/goe_telemedicine_2010.pdf).
- [30] G. B WILHELMSEN. Barns funksjonelle syn. 2012.
- [31] WinSCP. Introducing winscp, [Accessed September. 8, 2018]. URL: <https://winscp.net/eng/docs/introduction>.
- [32] ZeroMQ. `zmq_socket(3)`. URL: <http://api.zeromq.org/2-1:zmq-socket>.
- [33] Karen M Zundel. Telemedicine: history, applications, and impact on librarianship. *Bulletin of the Medical Library Association*, 84(1):71, 1996.