Western Norway University of Applied Sciences

BACHELOR'S THESIS

HEQED House

Lars Emil Bjørnsen Norevik

Oliver Abraham O´Loughlin

Trond Hauge

Information Technology/ Computing

Faculty of Engineering and Science

Department of Computer science, Electrical Engineering and Mathematical sciences

Atle Birger Geitung

3rd of June 2022

I confirm that the work is self-prepared and that references/source references to all sources used in the work are provided, cf. Regulation relating to academic studies and examinations at the Western Norway University of Applied Sciences (HVL), § 10.

TITLE PAGE FOR MAIN PROJECT

Report's title:	*Date:*
HEQED House	2022.06.03
Authors: Oliver Abraham O'Loughlin Lars Emil Bjørnsen Norevik Trond Hauge	Page amount w/o attachement: 48 *Attachement page amount: 0* *Number of floppy disks/ CDs: 0*
Contact person at field of study: Atle Birger Geitung	Grading: None
Notes: None	

Client:	Client's reference:
HEQED	None
Client contact person: Maria Nordheim Alme	Telephone: +47 93445929
Client contact person: Clark Nowack	Telephone: +31 6 55757214

Abstract:

The HEQED project aims to increase health equity in the world through education, where this project, the HEQED House, is a subproject. Through prototyping and evaluation of different technologies within game engines, this project aims to create a framework for learning and information exchange that fits within vision. The framework will consist of a website that acts as a portal for a library of documents, information about the HEQED project, as well as carrying an interactive, virtual application. This will form the HEQED House. With a limited budget, and a need for further iterations, this project needs to fulfil the vision in a way that makes the HEQED House relatively cheap to operate. At the same time, the complete service must be safe, universal, and easy to use, taking the target group, the public, into account.

Keywords:

Next.JS	React.JS	Unity

Western Norway University of Applied Sciences, Faculty of Engineering and Science
Postal Address: Postboks 7030, 5020 BERGEN Address: Inndalsveien 28, Bergen
Phone: 55 58 75 00 Fax 55 58 77 90 E-mail: post@hvl.no
Homepage: http://www.hvl.no

PREFACE

Health Equity in Education, HEQED, is a project that aims to improve health equity in the world through education. HEQED House is a subproject within HEQED that works towards building an educational platform for this purpose.

The HEQED team gathered in Bergen, Norway February 24th. The goal was to talk about- and define the project. It started as a very open project and is continuously being narrowed down. Our part of the project was concretised to focus on refugees.

A big thank you to Maria Nordheim Alme, Atle Birger Geitung and the rest of the HEQED team. We also want to extend a special thank you to Clark Nowack, Isabel Anton Solanas and Davide Ziveri for working closely with us on the HEQED House project.

DICTIONARY

Word/ Abbreviation	Definition
Health Equity	Individuals' feeling of being equally treated in- and receiving equal quality of healthcare.
HEQED	Health Equity in Education or Health Equity through Social Education for a Sustainable Society.
NGO	Non-Governmental Organisation.
GDPR	General data protection regulation, an EU law that regulates handling of personal data.
Game engine	Core software for games and most interactive, virtual applications.
UE5	Unreal Engine 5
Unity	Unity game engine, made by Unity Technologies.
GUI	Graphical User Interface.
SEO	Search Engine Optimisation.
SSR	Server-Side Rendering.
SSG	Static Site Generation.
CSR	Client-Side Rendering.
API	Application Programming Interface
DDoS	Distributed Denial of Service
UI	User Interface – a term used both for interfacing between user and application. This can be both the information itself, and the way it is represented to the player, or the way player communicates back.
CPU	Central processing unit, the "brain" of the computer, used to handle most data.
GPU	Graphics processing unit, a processing unit that normally is better suited at rendering graphical outputs than the CPU.
WebRTC	An internet protocol suitable for real-time streaming. This means that some packages can be lost in order to avoid delays in the stream.
WebGL	A JavaScript API that allows rendering of complex 2D and 3D in a web browser.
Web Assembly/ WASM	A low level, faster alternative to JavaScript that is more suitable for performance demanding applications, like game-like applications that needs to be played in a web browser.
Developer team	The student group working on this project

TABLE OF CONTENTS

Contents

TABLE OF FIGURES

1 INTRODUCTION

1.1 Context

The goal of the HEQED project is to promote health equity through education. While this is a very wide scope, the intention is to narrow it down and measure concrete factors in health equity. The HEQED project started in autumn of 2021 and will run for the following three years.

Computer technology is one of the elements being considered. It is easily scalable and can – if done right – be widely accessible. This is, however, in a relative sense. There are often limiting factors in the areas where health problems are most severe, such as lack of mobile coverage or electricity. This is one of the problems highlighted by the NGO Humanity and Inclusion.

University of Zaragoza has experience with escape rooms and putting students in different situations to socialise and engage learning. Therefore, they want something that can be used in their current education. This shows different interests, despite the common goal being increasing health equity, and the HEQED project's need for different expertise.

1.2 Motivation

There is a significant health inequity in the world. It is an economic burden to the countries involved in refugee crises, and an even bigger burden on the refugees themselves, often resulting in both mental and physical problems. It is important for the public and health workers who meet these refugees to understand this. To be able to emphasise with and understand each other better, may lead to individuals making better informed choices. Especially in the context of healthcare. This is at the core of what health equity means – that people feel equally treated in healthcare and receive the same quality of healthcare. When this is achieved, it can make integration of refugees easier, but also provide better general help for those who need it.

February 24th, 2022 is a day that brought war back to Europe. While it is not the goal of this project to go into politics, it still shows the importance of educating societies in making better choices. War and famine are not new problems and continue to be the source of major health crises, and education may play an important role in preventing this.

The situations can be extreme, first and foremost for the refugees (Jong, 1995), but also for the people working with the refugees. This makes it important to be able to train for what to expect. In *Protecting the psychological wellbeing of staff exposed to disaster or emergency at work: A qualitative study* (Brooks, et al., 2019), it is stated that many participants in the research tended to receive more reactive psychological support, rather than proactive. The participants also suggested that psychosocial training may be beneficial. More on this in Problem Description and Goals in this chapter.

The use of, and adaptation of, technology may lessen some of the challenges faced in the mentioned situations, such as the need of psychological preparation for relief workers. I.e., for what the worker is going to encounter. In the Norwegian oil industry, workers are being prepared using virtual reality simulations, before being sent offshore. This is one of many potential solutions that can be adapted to help relieve workers, or just increase the knowledge of health inequity.

1.3 Project Owner

The project owner, the HEQED project group, consists of employees from Western Norway University of Applied Sciences, HAN University of Applied Sciences, University of Zaragoza, Arcada University, Humanity and Inclusion, and Center for Victims of Torture. These are the owners of the HEQED House project.

1.4 Problem Description and Goals

As mentioned in the previous subchapters, there are many challenges and stakeholders in the HEQED project. The needs are many, and each stakeholder's main interest varies, which itself becomes a challenge on a meta level: The HEQED project's changing requirements and goals during their ongoing discoveries, putting more responsibility on the developer team. Therefore, it is important to understand the context of this project, being under the HEQED project. Agility is therefore important, and preparation for sudden changes.

This leads to the importance of focusing on the absolute most basic and/or flexible features, with the least likelihood of being scrapped. Hence a goal is to create a product and explore ways it can solve problems, and therefore be of value to the HEQED project.

A request from the start is a framework for some sort of a virtual application that lets multiple users interact in a virtual space. Referencing chapter 1.1, stakeholders from University of Zaragoza are requesting a service that can work as a tool in education. Combining this with what is stated in chapter 1.2, the Zaragoza stakeholders have a potentially strong case for problem solving, in combination with their previous experience with escape rooms and putting students into certain situations. The task is therefore to create a framework that can facilitate this, and which can be further expanded upon.

Furthermore, the HEQED project needs to be able to easily reach out to the public. This is where the need for a website comes in. The aim here is to create an easy-to-use platform, that connects as much of the project as possible. If different disciplines of the stakeholders in the project are to be combined, it is important to combine them in a meaningful way. Hence, the website should work as a portal for both immediate information about the HEQED project and mission, as well as the virtual application. How this is to be done is the main challenge of this project.

1.5 Report Structure

Chapter 1 Introduction introduces the HEQED House project, the motivation behind the project, who the HEQED-project group is, what the problem description is and what the goals of the project are.

Chapter 2 Project Description gives an in-depth description of the project, initial requirements and initial idea for solution, delimitations of the project, the resources used and literature on the issue.

Chapter 3 Project Design goes through the different proposed solutions, choice of tools and the reasoning behind it, and the project methodology.

Chapter 4 Product Design clarifies and explains the solution in greater detail, outlining the applied approach to achieve project goals. It is important to note that the final product is not intended to be in a finished or fully featured state, but to be a ground layer and function as a framework for further development.

Chapter 5 Results describes how the evaluation of the result is done. It is described which types of evaluation methods are used to ensure the quality of the project and the evaluation results.

Chapter 6 Discussion discusses the results and explains the reasons that lead to them.

Chapter 7 Conclusion and Further Work concludes the project and goes into to detail about potential further work on the project.

Chapter 8 References list up all the references that are used in the report.

2 PROJECT DESCRIPTION

2.1 Practical Background

2.1.1 Pre-Existing Work

The HEQED House project is a standalone, scratch-built project in the HEQED project.

The HEQED project consists of members with different competencies within healthcare. These are people who have worked in relation to relief work, hence the HEQED project builds upon the People with Refugee Experience Project (PREP) – another project building competence in working with refugees. One of the products from PREP is an online course aimed at physiotherapists. The goal of both HEQED and PREP is to better prepare healthcare professionals in the face of treating refugees.

The background of the involved organisations of the HEQED project is strongly focused on helping people with disabilities, for which Humanity & Inclusion is a co-recipient of the 1997 Nobel Peace-Prize. This is a group of people that grows during conflict (Edward Winter, n.d.), and whom the HEQED project aims to increase feeling of being treated fairly in order to increase health equity.

2.1.2 Initial Requirements

The requirement from the task description for the project, is a framework for a virtual application where students, teachers and clinicians from all over the world can meet. Material should be available in the application, and it should be possible to share documents in the virtual space.

From the initial meetings in January 2022, it is determined that a website is required as well. This will serve as a portal and administration tool for the HEQED House. It may also work as a portal for the HEQED project as a whole. During the same meetings, the need for a virtual application is put in uncertainty. It is suggested that this can be used as a possible replacement for video conference applications like Teams and Zoom. This is seen as unnecessary, as current solutions are better than what would be produced by this project. However, as further explained in this- and later chapters, a new solution is found.

At the end of February, it is determined that the platform will eventually be used in teaching, gathering information and general meetings. This concept is closer to the roots of the initial task description, with some key differences. The focus should be on showing content, rather than sharing files.

An expressed wish is for it to enable the user to experience what refugees experience, with a variable degree of severity. A further wish is for the refugee users to be able to express their experience. Hence, the virtual application is required to let users enter different rooms for different experiences. This will allow for future expansion of features. A request from the French NGO, Humanity and Inclusion, is for the app to allow for documents to be available offline. The latter request is more in line with one of the focus points from discussions in January, regarding a file system. As the current file system for uploading documents is difficult to use, there have been requests for something easier. Both the offline request and ideas around the file system will be brought up again in chapter 2.2 Delimitations.

2.1.3 Initial Idea for Solution – Virtual Application

With such an open task description, and from what is understood of the task, the initial idea is to focus on justifying the virtual application. This is because the competition is strong from alternative, already established products, such as video conference services like Zoom and Teams. First and foremost, a virtual application requires a game engine, or another real time rendering framework. Game engines provide few, if any, document renderers for files like PDF or spreadsheets. Implementing this requires a lot of work. However, both Unreal Engine and Unity provide web browsers inside the game engine. As many browsers support rendering of PDFs, the idea is then to use the built-in web browser to display document files.

As potential competition from Meta's Metaverse may be tough, the HEQED House framework must be based on something forward thinking. An example is Epic Games' use of their game engine Unreal Engine for The Matrix Awakens (Anon., 2021), utilising Metahuman, which provides movie quality, animated characters with real time facial capture of the user (Anon., 2021). The ability to animate facial expressions quite accurately is initially seen as important because the application may have to compete with video conference applications. Unreal Engine is therefore, at the early stages of this project, seen as the most viable option. This view will come to change as the project evolves, as later discussed in chapter 3.

With the features the initially chosen game engine provides – in combination with the HEQED project's multinational focus – one of the early ideas is to focus on putting students from different countries together. This is simply to learn from each other, while using their potentially registered interests to connect them together.

During the meetings in December and January, an idea of making the virtual application as an alternative to a conference is raised by parts of the HEQED project. However, because of the doubt about necessary equipment for a VR experience, a request for a live video streaming service is raised. This is rejected, as the feature would end up as a much inferior alternative to current video conference services. The idea is then turned back into how the HEQED House can provide something new.

The question of a virtual conference centre is raised. An application where users of different roles get different privileges inside the application is then discussed. One is automatically sending users to the correct room, or potentially a connected server, when the actual conference starts. This would allow users to mingle without getting lost. The HEQED Project expresses a wish for users to be able to wander around and find more private spaces for conversations, much like in a real conference setting.

2.1.4 Initial Idea for Solution - Website

Websites have come a long way in the last ten years, allowing websites to behave more and more like applications. As this is a Scandinavian project, it is therefore natural to put some Scandinavian Design ideas into it, as well. This should aide the application in being minimalistic and functional, and therefore be easy to use- or adapt for users of all types. In other words, universal design should be easier to achieve with this mind-set.

More concretely, the idea is a website with a simple presentation of what it contains and what its purpose is. In this case, it means a section for the library, and one for the virtual application, which

may be a streaming service. Relevant sub-pages should lay behind these, so the user has a logical starting point without having to browse through a large or complicated menu, or any other hyperlinked text. Tools – for example filtering or altering of items on the page – should be placed in a tidy, non-obtrusive manner. In combination with this, a responsive site with low loading times can create a fluid experience.

The application page on the website will allow the user to access the virtual reality application from anywhere in the world. This page may be tied to the login on the website itself and provide the user with automatic login into the application. This will allow quick and easy access. Using this page for streaming will eliminate the need for downloading the application, which may require more internet access and hardware than available.

Information security is important. Therefore, it must be a site that abides by laws and regulations, and still presents a consent form that doesn't annoy the user. This is also important to the HEQED project as a whole, as it is crucial to not further lessen the security for any users – especially refugees. More on security in the chapter 2.4.

2.2 Delimitations

The initial project description is worded in such a way that it may cover all aspects of health equity, with the limitation of the virtual application being primarily geared towards students, teachers and clinicians. The original plan is therefore to utilise Unreal Engine's ecosystem and library. Animation, as well as 3D- and 2D work, takes time, and sufficient assets for the HEQED House's purpose is included in Unreal Engine's library. Hence the focus will be on the technical aspects of this project. However, this slightly changes, as following paragraphs will go into more.

The inclusion of a fully functional website puts a heavier time constraint on the development of a virtual application. However, the file library is already a part of the original plan, due to the support for a web browser in relevant game engines.

On the 24th of February 2022, the virtual application is further changed to work as an experience centre. Due to the late change, the time becomes a limiting factor. The greater part of two months of planning is undone, albeit presented in a constructive manner to the project owner, who understands the situation well. Hence it is determined that the virtual application will be a framework with a simple experience centre functionality.

As mentioned in chapter 2.1.2– regarding the website – during the February meeting there is a request for documents to be available offline and using the website as a replacement for the current file system. Offline documents are achievable by letting the user download documents in advance. An alternative is creating a downloadable application, but that is outside the scope of this project. The file system cannot entirely replace what is currently used for uploading documents. This is due to many of the documents being research material, required to be uploaded to servers dedicated to research material following regulations for sensitive information.

The developer team does not feel comfortable or qualified in handling sensitive information. This further limits the request for the virtual application to collect experiences from refugees, where there naturally is a lot of high-risk sensitive information involved. One feature request is to allow refugees to enter a private room that records audio from the user's microphone input in the application. This poses security- and legal risks. As the application is meant to allow multiple users to

interact together, there could be a bug causing users to listen to each other when they should not be able to. This can happen for multiple reasons, including lack of calculation accuracy of graphical hardware, though not likely. As such, this feature is deemed too high in risk to include.

Due to development of the HEQED House having to be done mostly at home, version control is of utmost importance. During work with it, it is discovered that it is difficult to work in a group project remotely in Unreal Engine. 16th of March, the developer team is given access to the master lab, together with a desktop workstation at HVL Kronstad. However, this only allows for one worker at a time, and has the additional requirement of rigging up and -down the workstation before and after every use. It is determined that it causes more work than time saved, and Unity is chosen as the new game engine. Unity is more lightweight, allowing for meeting up and working on laptops. It also provides an easier tool for collaboration on projects. However, as previously stated, Unity's library of free resources is limited, limiting the application to a purely technically functional one.

2.3 Resources

From the task description, Microsoft's Azure is one of the suggested cloud services for this project. HVL have an agreement with Microsoft and expresses that this is what should be used for this project. Azure supports both hosting for web applications, as well as servers for streaming of applications, enabling streaming of the virtual reality application. The database system PostgreSQL is also supported, with Azure Files as an option for file system. With the scalability and possibility of maintaining the system from different locations, Azure is an ideal service for the HEQED House. This is suggested to be a part of the HEQED project's budget.

The HEQED project will supply the HEQED House with content. This includes text, graphics, documents and other public files relevant to the website. This also extends to the virtual reality application that is to be used for showing the audience curated experiences of refugees.

In line with the task description, HVL provides a workspace for the HEQED House. However, due to initial lack of space, this is delayed into March 2022. At the same time, HVL provides the necessary virtual reality headset for testing towards a full set of features.

Quixel, purchased by Epic Games, offers their Megascans library of materials and 3D assets for free when used with Unreal Engine. This allows the virtual application to provide a better-looking experience, with movie quality assets. However, the assets can also be adapted for use with lightweight applications.

Unity Asset Store/ Unity provides some code packages and minimal assets for free to kick-start projects. While not as fleshed out as what Epic Games provides, it is sufficient for a framework.

Other asset resources, mainly in the shape of Node packages and 2D and 3D assets, are listed in included *DAT191_Asset-List*. These are code packages to build the website, and other assets to aid in visual presentation of the website and virtual application.

2.4 Literature on The Issue

As the website is primarily aimed at the European area, it must abide by General Data Protection Regulation, or GDPR. These are regulations aimed at giving users more control over their own data and are laid out on the Norwegian Data Protection Authority's website (Datatilsynet, 2018).

However, ever since the implementation of the directive, sites and relevant service providers for the sites are looking for ways to still get their data collected. While sites are technically abiding by GDPR, it is often tiresome for the user to manage the collection of their data. Hence, the terms such as "cookie fatigue" have appeared. This is something organisations like UK's Information Commissioner's Office are discussing and publishing about (Information Commissioner's Office, 2021).

There is a wish for a solution to let the user stay logged in, while also keeping the data safe and away from external parties. Hence, research into cookie alternatives is being done. Using an account that a user already has, seems user friendly and intuitive. This will further enhance the user friendliness of the site. Login on different sites by using a Google account is often possible. Hence research into what Google has done is conducted, where it is discovered that Google claims to anonymise their data (Temkin, 2021). However, a recent case that that has gone through the Austrian legal system puts Google's efforts into protecting its users into serious question (Jennifer Bryant, 2022). This case causes concerns over the ability to protect users. Especially political refugees.

Making the HEQED House handle the data internally makes it easier to abide by future laws and regulations. One example is the European Union's Digital Service Act (European Union, 2022), requiring transparency in the handling of personal data. From the various literature used, it is clear handling information carefully is important.

Taking care of the user's safety is not the only important point to consider. In a project aimed at people with disabilities, universal design is also important. Going back to chapter 2.1.4, Scandinavian Design provides simplicity, aiding creation of a clear, more readable design. However, simplicity is not the only factor. The Norwegian Agency for Public Management and eGovernment (Difi) increased focus on universal design in 2018, when they threatened Scandinavian Airline Systems with daily fines for their web design (Gytri, et al., 2018). As white on dark blue (NTB, 2018) may not look too difficult to read or navigate through, it is important to investigate why this design is a problem. Hence, Uutilsynet (Uutilsynet, n.d.) and Centre for Excellence in Universal Design (Centre for Excellence in Universal Design, n.d.) are being used.

As HEQED House is a project that is supposed to be continued in the future, choosing a good and easy framework is important. As new developer teams take over the project, expertise may vary. The current one is an example, as it has no deep experience or specialisation in web development, but rather graphics and software development. Unreal Engine, which is still the framework of choice early in the project, requires Node.JS as runtime for the server that connects the peers in the WebRTC connection in its streaming package. It is therefore decided to use as similar technology as possible to minimise problems if any are to occur. However, that raises the question of how to build upon Node.JS.

State of JS is a public, online survey aimed at JavaScript developers. With 16 085 respondents in its 2021 survey (State of JS, 2022), measuring what is thought of popular tools and frameworks, this and previous surveys are seen to give good indicators on what to pick. It is especially helpful to not only see whether the different frameworks are good, but also to see if they're widely used, indicating that there may be a good number of resources on problems, as well as extensions online.

An observation made is that the frontend framework React has a good reputation and is widely used. However, it needs a backend framework. Next.JS works as both a backend- and frontend framework,

also called a meta-framework, utilising a lightly modified React as the frontend part. This is discussed in detail in chapters 3.3 and 4.1.1.

3 PROJECT DESIGN

This chapter discusses which choices are made regarding how the project is managed, which approach is used for developing the product, and an overview of the tools required to do so.

3.1 Proposed solutions

3.1.1 Alternative Solution 1: Virtual Application with Integrated File Sharing Solution

The proposed solution is to create a virtual application acting as an educational experience centre for students, teachers, and healthcare workers, which has integrated functionality for sharing documents with others. This means it should be possible to open and view documents, and to show these contents to other participants in the virtual space. This can, for example, take the form of being on a large screen within the virtual space that is rendered equally for all participants.

The virtual application will be made using a game engine. It will include voice over IP (VOIP) to let users talk to each other and provide a framework for virtual rooms or spaces which contain modular experience environments. The actual rendering of document files will be done either natively in the game engine, or by using a web browser within the game engine. This solution minimises the need for and requirements of a website, as the only functions of the website will be to display documents and act as an interface for the virtual application. Users will not interact with the website directly. In this solution, the virtual application represents the majority of the product and is heavily focused on, while the website has minimal functionality.

Being the main focus of the project, the virtual application will contain more functionality. As most of development time is spent building the application, it increases the possibility of implementing more ambitious features, such as multiplayer functionality. This approach also favours the use of Unreal Engine, as a larger scope for the virtual application benefits from a more feature rich engine.

3.1.2 Alternative Solution 2: Virtual Application + Website Portal for File Sharing

This solution aims at dividing the two problems of virtual experiences and file sharing into two separate parts, and then linking them together. In this solution the website takes a more central role and contains more functionality. The idea is to create a website which contains a library where one can view and upload files. The website will also act as a portal to access the virtual application, and possibly the HEQED project as a whole. Meanwhile, the virtual application will be made as a standalone application using a game engine, in a similar fashion as described in solution 1, but with the crucial difference being that the issue of file sharing, viewing, and uploading, is entirely handled by the website. Users will interact directly with both the website and the virtual application.

A more ambitious part of this solution is to make the application stream-able from the website. This will provide multiple benefits, such as making the application accessible on a wider range of devices and ensuring a certain graphical and performance quality, as user hardware would not affect these aspects when the application is running on a server and video output is being streamed to the user. As this aspect of the solution is more difficult to achieve, an alternative is to simply let an installer be available for download from the website, which the user can then run to install the application on their machine and run it locally.

The virtual application will contain a more limited set of features than proposed in solution 1. The limited scope also makes the use of Unreal Engine less favourable, as it becomes a question whether the extra work required to use it would provide any benefits. However, this allows for the website to be expanded.

3.1.3 Discussion of alternatives

The benefits of solution 1 is to keep as much of the product's functionality confined to the same space as possible, the virtual application. The website, if needed at all, will act as an interface for the virtual application, and not be interacted with directly by the user. The point of this is to create a more unified and streamlined experience for the user, as they are in this case only concerned with interacting with a single program. Such an all-in-one solution also requires a minimal amount of programming languages and frameworks to be developed, saving time by decreasing the number of tools needed to be familiarised with. Having a larger focus on the virtual application also allows for a more feature complete experience centre framework to be created, and more time spent on implementing advanced features. The shortfalls of this approach are mainly linked to the increased difficulty in integrating the file sharing- and viewing solution directly into the game engine. Game engines generally do not offer any native support for this feature, hence why a website may be required. It will also be required to implement a GUI in the virtual application for navigating the library of documents. The developer team has little to no experience developing GUI in the relevant game engines.

Solution 2 offers a split approach, where features are divided between a website and virtual application as seen most appropriate. This is beneficial by making the implementation of said features easier. Rendering of PDFs is supported by most web browsers, and there already exists multiple libraries to simplify this task. The user interface for navigating the library will be implemented on the website instead of the game engine, allowing utilisation of HTML, CSS and JavaScript, which the developer team has some previous experience using. React.JS will also be utilised, which is a popular and capable library for building user interfaces. This approach will make the documents library more accessible, as users will not be required to go through the virtual application to access it but can interact with the website directly instead. It is likely to also make downloading of documents easier to implement, which would benefit users wanting to access documents offline, and fulfil Humanity and Inclusion's request, mentioned in chapters 2.1.2 and 2.2. This solution does, however, also have some disadvantages. It will increase the number of frameworks and tools needed to be familiarised with, and split the project into two distinct parts, instead of keeping everything within one single package. It will also limit the scope of the virtual application, resulting in a less feature complete experience centre framework.

From the initial interpretation of the project's purpose and requirements, solution 1 seems to be the most appropriate approach, as the focus is primarily on making a virtual application for meetings. At this stage the developer team is also optimistic about being able to implement most, or all, of the intended features, including rendering of PDF files within the game engine. At the early stages of the project the need for a website is also minimal. However, after the first meeting in January with Maria Nordheim Alme and further discussion, the benefit of having a website to handle file sharing, in addition to acting as a portal to the rest of the project, becomes clearer. As the project evolves, the idea of what the virtual application should include, also changes. The need for a good file publication tool and library, as well as a shift in the project's scope, makes solution 2 look more favourable.

3.2 Chosen Solution

With early prototyping and discussions with the project owner, it is clear that a website with more features is needed. Therefore, a solution based on the one from chapter 3.1.2 is chosen. The focus on communication between users inside the virtual application is changed to focus on conveying information. In combination with there two months into the project not being any available space for the developer team to work together, it is decided that a transition to Unity is beneficial. Unity provides a version control system, allowing work to be done remotely. At the same time, it is possible to run Unity as a development tool on a regular laptop, allowing the developer team to come together to develop.

The chosen solution is expanded to be a full website, with user- and authentication systems. This includes user interface handling of content that is more aimed at human browsing. Meanwhile, it is decided that the virtual application will be a bare minimal framework, where the user enters a virtual experience centre. The user will enter with a first-person view and be able to walk around. While it is not determined exactly what the content should be, images and video will be tested. This framework will limit users' movement to different rooms. This is to provide special rooms for special needs.

3.3 Required Tools

To build the website and virtual application, several programming tools and frameworks are needed. For the virtual application, a game engine and IDE is required. The most relevant frameworks for this project's goals are Unity and Unreal Engine. To start with, Unreal Engine seems to be the preferred choice, for its richer feature set and abundancy of high quality, free assets. Because of this, the initial decision is to use Unreal Engine 5. Metahumans' and Quixel Megascans' free assets, as well as multiplayer capabilities, are especially attractive features of UE5. However, after early prototyping it becomes clear that Unreal Engine has a steeper learning curve than Unity, while Unity also holds the advantage for being more performant than Unreal Engine in some cases, having better VR capabilities, as well as offering the possibility of running the application directly on the website, in a web browser. The scope of the virtual application is also required to be further limited, as project goals are being changed and the website is taking up a lot of time resources. The reduced scope also limits the need for UE5's more advanced feature set. These points are discussed, and the team decides to move to the Unity game engine for developing the virtual application.

For the website, as mentioned in chapter 2.4, it is decided to use a framework that utilises Node.js – a JavaScript runtime environment that lets JavaScript be used for backend services. Next.JS, a React.JS based framework which is compatible with Node.JS, is chosen as the meta-framework. Using Next.JS provides many features, such as easy routing, fetching data, and TypeScript support. In addition to this, Next.JS is flexible when it comes to the method of rendering a page, be it SSR, CSR or SSG. This is also important when having search engine optimisation in mind. The latter may be important for HEQED project's ability to reach out to the public. However, the most important reason for this choice is the combination of client-side rendering and streaming the virtual reality application. More on this in chapter **Error! Reference source not found.**.

In relation to both the virtual application and the website there is a need to store data in a database. This is to allow for functionality such as creating user accounts, logging into user accounts, and storing document and organisation information. PostgreSQL, an object-relational SQL database, is

decided on because of the developer team's previous familiarity with the system, allowing for an easier implementation. PostgreSQL is also chosen for it being a relational database as opposed to a NoSQL database. This allows for relationships between objects to be clearly defined, and for this project it is a wish to relate users to universities and other institutions, as well as enforce constraints on data. For storing document files Google Cloud Storage, and later Firebase Storage, are used.

3.4 Project Method

3.4.1 Development Methodology

From the beginning of the project, it is clear that the goals and requirements are prone to change. As the HEQED project is newly started itself, it seems to be at an exploratory stage. To tackle developing software under these circumstances, it is decided that it will be best to follow the principles of agile software development, as defined by Agile Alliance. One of these principles state "Welcome changing requirements, even late in development" (Agile Alliance, 2022). To adhere to this, focus should be on staying ahead of the curve and anticipating which parts of the project are most likely to change – keeping focus on developing the most essential parts that are least likely to change. The developer team should work incrementally and use tools such as Trello, a web tool similar to Kanban board, to map the ongoing status of the various tasks to be done.

As part of an agile software development method, it is important to keep regular and continued communication between the developer team, the supervisor, and the HEQED project team. This is done through a combination of digital meetings and physical meetings at the HVL Campus Kronstad. Team members work both remotely and together in the same physical space and keep frequent communication to hold all members in the loop and up to date on the project progress.

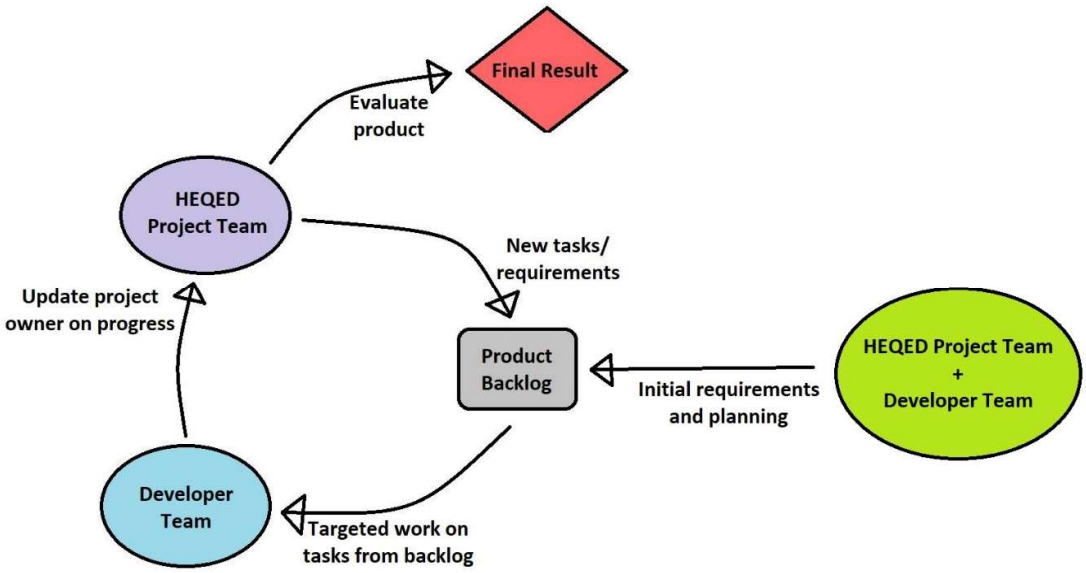

Figure 3.1 Development Cycle

This method of development also takes some ideas from Scrum, a framework for agile software development. Scrum describes a small core team which works in an incremental fashion with regular input from the product owner (Scrum.org, 2022). This model seems to fit well with this project's

circumstances, as there is a small group of people involved with the HEQED House project, and ideas are continuously being received from the project owner. Therefore, Scrum is deemed to be applicable and of value in optimising workflow. One of the ideas adopted is to take input from the project owner into a product backlog. In this case Trello functions as an overview of the product backlog, where tasks received from the project owner are added, and the developer team can track progress and target tasks to complete iteratively. In this way the idea of selecting work from the product backlog to increment value to the product is also incorporated into the development method (Scrum.org, 2022). Figure 3.1 illustrates the intended way the development cycle will work until a finished product is produced.

Going back to the topic of the HEQED project being a relatively fresh project and agile development's welcoming of change, an extra measure taken is to not stick too much with specific solutions. The reason for this being to avoid spending a lot of time on specific features that may be scrapped. In other words, welcoming change. To narrow the project down, attempting to avoid these problems, regular meetings and openness about limitations are important.

With some uncertainty on the HEQED project team, they have gone for virtual reality – a relatively unexplored technology that is far from being a regular household item. However, it shows potential in many sectors, be it healthcare or the engineering industry, so it seems likely that there are still areas where it can be of value. As this project is not aimed at hardcore fans of new technology, but in a learning context for the general public, a survey is conducted to address different aspects of the technology and see if it can help or potentially hinder in specific areas. This survey is sent to people in as different areas of work or life situations as possible. This is to avoid skewing the result by presenting the survey to biased groups of people. An assumption here is that gamers and IT personnel may be groups of people with bias.

With this, the project follows design science, exploring ways of putting a virtual reality, or virtual space at all, in an educational context. This is especially important as the project goal is very open at the beginning of the project, giving the developer team a better ability to give feedback to the project owner on solutions from potential findings and feedback from the survey. As there are different opinions on VR within the project owner, it is also important to try to see why, and if this matches with responses from the survey. Besides asking the members of the HEQED project for their input, there is also a chance the survey can show some aspects that can alleviate scepticism.

As stated in chapter 3.5 of the Vision Document, Meta, previously Facebook, has garnered a lot of criticism. As they are one of the biggest companies in the world, a survey may also help in seeing if Meta's situation (Isaac, 2022) has spread over to the general opinion about virtual reality. The results may further aid in making the virtual application part of the HEQED House product one that the public wants and finds useful.

3.4.2 Project Plan

The developer team decides to create prototypes for the website and desktop application to expose any issues early on with the chosen frameworks and solution, and to get a clear understanding of which elements of the project would require the most amounts of work. The website prototype is started on first, using Next.js, followed by the virtual application prototype made in Unreal Engine. The prototypes are the groundwork for the most essential parts of the website and virtual application. As the prototypes are created, the developer team becomes familiarised with the

different frameworks and discover their strengths and weaknesses. The lessons learned at this stage are used to make choices going forwards in the project. It is at this point, in conjunction with the meeting on March 3. 2022, that Unity is selected to be used for building the virtual application.

After prototypes are created, development continues, with parts of the website and virtual application being created in parallel. The most critical and important issues take priority, according to the team's assessment. When all the essential parts of the project are implemented, focus can shift towards optimising and polishing features, as well as trying to implement features classified as being on the wish list. This, for example, includes streaming or running the application directly on the website.

3.4.3 Risk Assessment

To assess the different risks that can occur, a risk analysis with a risk assessment matrix is used to show the likelihood of the risk to occur and the consequence that can follow. Both likelihood and consequence are ranged on a scale from 1 – 5. These two values are multiplied by each other, resulting in a number ranging between 1 – 25, and determines the level of the risk. The different levels indicate which risks are minor and which risks will have a large impact on the project and must be avoided at all costs.

The risk analysis includes measures to take for each risk. This is to let everyone on the developer team know how to prevent the risk from happening and what to do if it should occur.

Risk	Reason	Likelihood	Severity	Risk-level	Measures
Functionality problems with the library.	Problems with reading, upload or download in the library.	(2)	(3)	6	Early development and testing. Reduce complexity.
Missing functionality in the application.	Some of the functionality in the applications does not get implemented. May be due to time constraints.	(2)	(4)	8	Early development, testing and develop prototypes.
Stakeholder would rather use existing solutions instead of the virtual solution.	Well established companies already have good and working solutions.	(4)	(2)	8	Find interests from stakeholders early. Make a survey. Have frequent dialogs with the project owner regarding the end-product.

(Depends on the virtual application) Pixel Streaming is not working.	Network routing or other problems with the implementation.	(3)	(2)	6	Testing. Possibly substitute Pixel Streaming with a standalone application. Give the stakeholders a realistic expectation.
Lack of motivation and competence.	Project owner has shifted the focus significantly away from the given task. The developer team then lacks competence.	(3)	(5)	15	Talk with the project owner and supervisor for a solution.
No funding for services.	Project owner is not interested in funding services like hosting, database and file storage for the website.	(5)	(3)	15	Perform evaluations and testing on localhost and free limited services.

Figure 3.2 - Risk analysis

		Consequence				
		Negligible 1	Minor 2	Moderate 3	Major 4	Catastrophic 5
Likelihood	5 Almost certain	Moderate 5	High 10	Extreme 15	Extreme 20	Extreme 25
	4 Likely	Moderate 4	High 8	High 12	Extreme 16	Extreme 20
	3 Possible	Low 3	Moderate 6	High 9	High 12	Extreme 15
	2 Unlikely	Low 2	Moderate 4	Moderate 6	High 8	High 10
	1 Rare	Low 1	Low 2	Low 3	Moderate 4	Moderate 5

Figure 3.3 - Risk assessment matrix

3.5 Evaluation Plan

During development, the website is being evaluated through unit testing in Postman, and further integration- and system testing on localhost. The virtual application is being tested through similar methods, and through using Unity's debugging console.

In parallel to mentioned testing, the website is to be initially evaluated running locally on a computer during meetings. From initial feedback, it will gradually be adapted to the project owner's needs. When the project owner provides content, and adaptation of the content has taken place, the website will be deployed online and made freely available for user testing and further feedback. External feedback from contacts with expertise within user interface design development is used to evaluate the design.

An online deployed website will also provide a stable database that can be used for testing loading of content in the virtual reality application, with its experience centre.

4 PRODUCT DESIGN

In this chapter, the project solution is clarified and explained in greater detail, outlining the applied approach to achieve project goals. It is important to note that the final product is not intended to be in a finished or fully featured state, but to be a ground layer and function as a framework for further development.

4.1 Website

The website is the first part of the project and is responsible for file uploading and library functionality. It is also meant to act as a portal to the virtual application, and potentially to the overarching HEQED project.

Error! Reference source not found. maps the general arrangement of the website, while Unity, the virtual application, is incorporated into the HEQED House page.

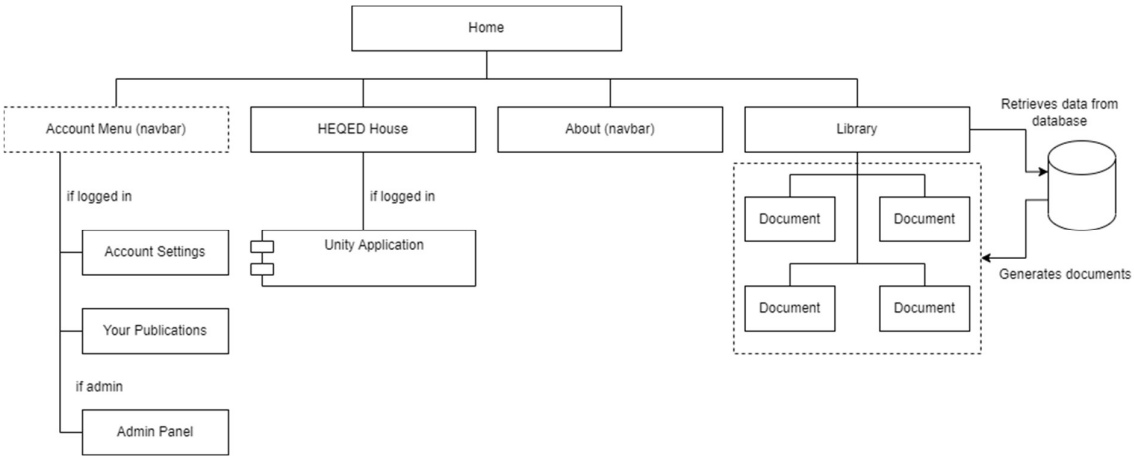

Figure 4.1 Base layout of the HEQED House website

4.1.1 Website Framework - Next.JS

Next.JS is chosen due to being a complete, streamlined package, with a rich set of features and low amount of boilerplate code. This makes it a system that is easy to learn and work with. In a Next.js project, the file structure decides routing paths, as well as providing a good overview of all project parts - be it styling, components or installed Node.JS libraries. During prototyping, the choice of Next.JS as the framework to build the website with is concluded.

Next.JS mostly acts according to the developer team's vision and reasons for picking it, providing a good platform for web development. However, there are some challenges due to choosing Next.JS. An example is data fetching. Being a flexible framework in terms of page rendering means that data can be fetched both on the server- or client side. Client-side data fetching - as described by the Next.JS documentation – proves to be more difficult to implement than fetching data through server-side rendering. In some circumstances it is, as further explained in chapter 4.1.5, desirable to fetch data on the client-side. Consequently, a larger strain is put on the group when trying to implement client-side data fetching, causing some unforeseen use of time. A diagram of how client-side rendering functions can be seen in figure 4.2.

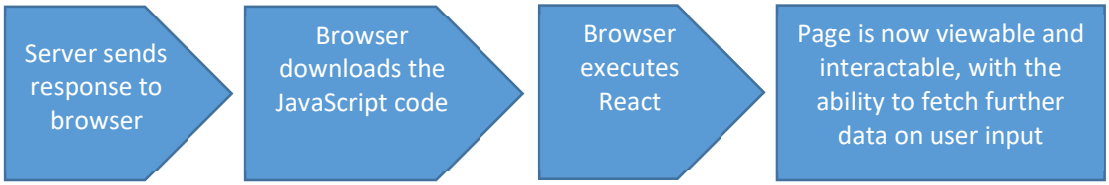

Figure 4.2 - React Client-Side Rendering

The upside of client-side fetching is a very responsive page, that can work efficiently on the database, and give results from work on the data in close to real-time after the first fetch. When correctly used, it will also cause less stress on the server and network, resulting in less fees when using a cloud service, as the data re-fetching is minimised. Client-side data fetching can be experienced in the library pages of the HEQED House Website.

Fetching of data on the client is done using SWR, derived from stale-while-revalidate, a React hook created by the same team behind Next.js. As explained by the Next.js documentation, "it is highly recommended" to use this hook if one is fetching data on the client side (Vercel, 2022). The reasoning for this is that SWR handles caching, revalidation, focus tracking, re-fetching on intervals, and more. The SWR documentation is followed and the implemented approach, as shown in Figure 4.3 and 4.4Figure 4.4, consists of creating a custom hook using SWR and a fetch function. The custom hook returns the fetched data in addition to a Boolean value indicating if the data is still loading, an error object upon failed fetching, and "mutate" – a function that can be called to re-fetch the data on demand.

```
export function useDocuments() {
    const fetcher = url => fetch(url).then( res => res.json() );
    const { data, error, mutate } = useSWR<any[],any>("/api/library/documents", fetcher);

    return {
        documents: data,
        loading: !error && !data,
        error,
        mutate
    }
}
```

Figure 4.3 – Implementation of custom hook using SWR

```
const { documents, loading, error, mutate } = useDocuments();

{function(){
    if (error) return <h2>Error loading documents</h2>
    if (loading) return <Spinner />;
    if (documents) return <FileCardList documents={documents.filter( doc =>
        filterByDocument(doc, title) && filterByOrganisation(doc, org) && filterByName(doc, author) )
    } />
}()}
```

Figure 4.4 – Using custom hook to fetch documents on client-side

Another thing to note is that React supports both functional and class components. The main difference being functional components accepting props as arguments and managing state with hooks, while class components implement state and logic, and can use React lifecycle functions. For this project the functional approach is used for simplicity. A general functional programming architectural style is also used for the rest of the website's logic, where the logic is implemented in standalone functions, grouped together in files as modules in accordance to relating concerns.

To host all this, multiple hosting options are discussed - one of them being Atle Birger Geitung's suggestion, Azure by Microsoft. This is the preferred option by HVL, due to their partnership. However, the HEQED project team is not interested in setting up and paying for a server at this stage in the project. Therefore, it is decided to use Vercel for hosting the website, as they offer a free hosting solution. Vercel is the company behind Next.js, and support all the frameworks' features and optimisations, making it an easy and cost-free deployment option. As Vercel uses serverless functions for hosting, and have a set of limitations related to this, an unforeseen consequence arises where the user is unable to log out after logging in. This is caused by the authentication cookie not being set properly upon a GET request. This unexpected behaviour is quickly fixed by changing the logout request to a POST request. No further issues occur from deploying with Vercel.

4.1.2 Authentication and Password

The authentication is done with a cookie that stores a JSON web token, often called JWT. The JWT is a JSON object that can consist of different attributes. These attributes are then hashed through base64 encoding, a reversible encoding that makes the attributes accessible again. The advantage is using this, packed in a cookie, where the values won't be stored as plaintext. It must be noted that sensitive information should not be stored in the JWT.

The origin here is JWT initially being tested out as an alternative to cookies. As mentioned in chapter 2.4, cookies are problematic. However, with the current knowledge in the developer team, it proves difficult to implement. Putting the JWT into a cookie solves some of the challenges when trying to use a plain JWT, as it proves difficult to find instructions on how to do it right with Next.JS or React. This is still seen as ethical, as the cookie and JWT stores no personal information, but a random, autogenerated hash tied to the user.

For passwords, the HEQED House utilises Bcrypt, a hashing algorithm designed to be slow, getting exponentially slower for each salt round, which the algorithm uses. This makes it slow to crack and is therefore strong against brute-force attacks.

On a sidenote here, sessions are considered. However, they are often ended when the user closes the browser. This is an unwanted functionality, as the user may prefer to not have to log in every time they visit the site. A solution is to store the session for longer on the server, but that would take some information control away from the user.

4.1.3 Database and File Storage

Different solutions for database hosting and file storage are explored. In the prototyping stage there is only need for a local database, as there is no deployed website. As development continues and the website is needed to be deployed, a database hosting solution is required. The selected hosting option will have to support PostgreSQL as this is the chosen solution. Google Cloud SQL is found to

be a good alternative, as it is easy to set up and provides three months of free use. The website then requires minimal configuration to be set to use the Google hosted database in production mode.

However, near the end of the project, all free credits from Google have run out, and therefore a temporary database hosted by Aiven is set up to allow for testing and final evaluation. The website configuration for using the database is mainly controlled through environment variables, making such a change very easy to implement on the website, requiring next to no re-writing of code.

Initially, binary data of files is simply stored in a column in their associated database table. However, this is quickly pointed out within the developer team and supervisor to potentially have a large impact on database performance. Upon investigating different possible solutions, it is discovered that the Next.js documentation recommends using a third-party option for persistent file storage (Vercel, 2022). As the database is already hosted by Google Cloud, this seems to be a good framework to start with. Google Cloud Storage proves to be easy to implement and work with and is therefore chosen as the file hosting solution. This choice is partly taken because of Google's simple to use Cloud Storage API, as well as the benefit of having both the database and file hosting handled by the same account and found in the same dashboard, making them simpler to manage.

In connection to free credits running out, like with the database, there is a need for a temporary switch in solution for file storage. Firebase is selected as an alternative and implemented on the website. Firebase is a platform created by Google, and Firebase Storage uses Google Cloud Storage infrastructure for file storage, making the switch less of a strain on the developer team. Firebase does however have a different API to interact with storage buckets, which means the uploading, fetching, and deletion of files on the website must be partly re-implemented, causing more loss of time. This is still seen as necessary because the developer team wishes to demonstrate the product in its entirety to the HEQED project team members and get an evaluation of the result.

Though the solutions for database and file storage hosting are changed near the end of the project, it is not intended to necessarily be a permanent switch. As such, all original code using the original solutions is still present and updated, making a switch back effortless. This is done in order to allow the product to be able to evolve in parallel with the HEQED House project requirements.

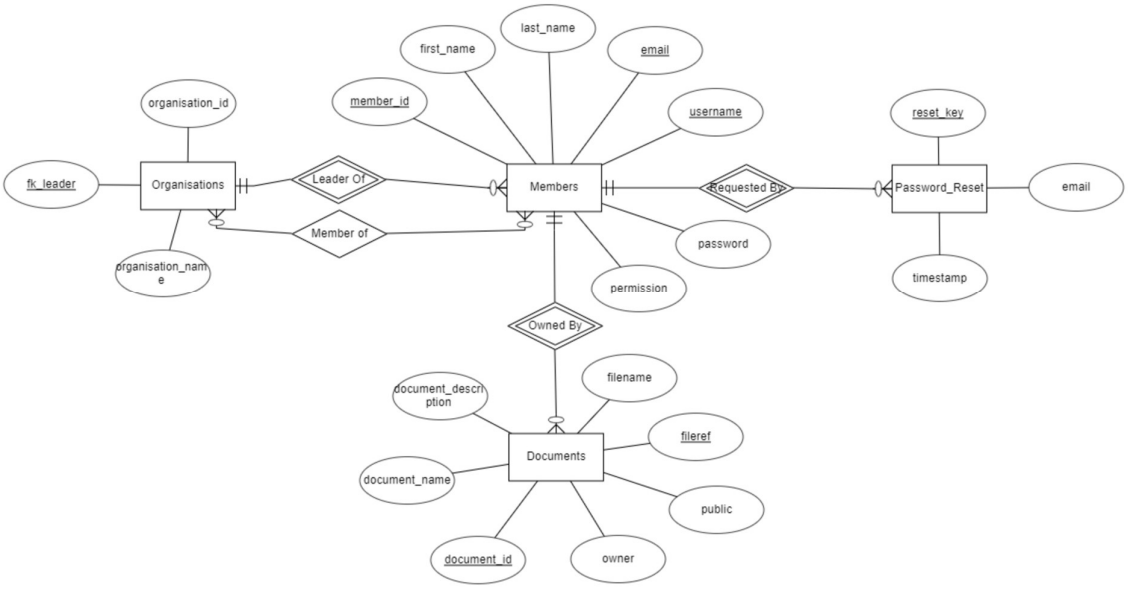

Figure 4.5 - Database Relationship Diagram

Figure 4.5 shows the current structure of the database and how entities relate to each other. There are four entities: Members, organisations, documents, and password_reset. The word member is used instead of user, as user is a reserved word and can therefore not be the name of an entity. Hence, all uses of the word member refer to the database entity representing users. Current design allows a member to be related to multiple organisations, or none. This is to allow for more flexibility in which organisations a user is associated with, and to allow for users that are not directly associated with any organisation. Furthermore, all other entities than members have a reference to a member as a foreign key. For organisations this is a reference to the member who is the assigned leader of the organisation, labelled “fk_leader”. Documents reference the member that they were uploaded by with the “owner” attribute. Password_reset has an “email” attribute which references the email of the member that requested a password reset.

4.1.4 User interface

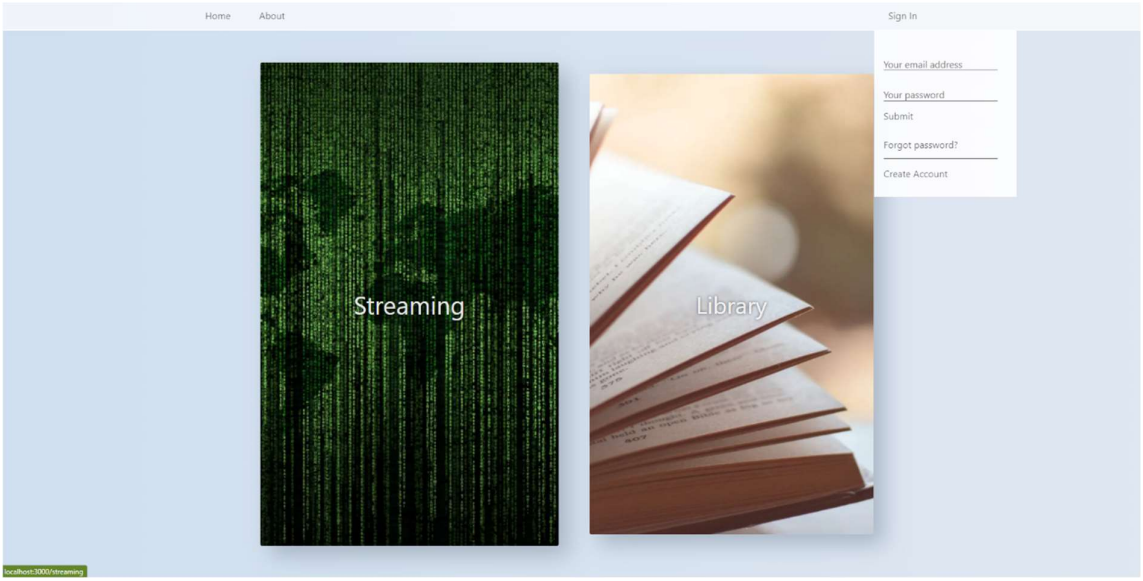

Figure 4.6 - HEQED House landing page

Going back to chapter 2.4, looking at the website of Centre for Excellence in Universal Design, pictorial and tactile presentation of information are some of the suggestions from the guidelines. As images are used to create a sense of what the content behind the button is about, it can be hard to see where the cursor is. Especially over an image. Enlarging the image, as can be seen in figure 4.6, with a quick animation and giving it a mild shadow, has the aim of creating stronger visual feedback, while not looking tacky. By doing this, instead of putting all the focus into contrast, the overall aim is to create a balanced visual. However, this may not be the right decision, which will have to be addressed on user testing and further research.

Users may have different devices, with different aspect ratios on the screen. Especially field workers are likely to use mobiles only. Hence, the CSS code takes this into account, as can be seen in figure 4.6 and 4.7, where the same page is shown. In figure 4.7, the CSS keeps the image buttons large to comply with uniform design. The login dropdown is handled to stay near its activating sign in button.

As the site must be prepared for further changes and enhancements, much of the CSS code utilises variables and rules that allow for expansion and easy changes. The colours and contrasts may be one of the most critical elements of complying with universal design, therefore as many elements as possible on the site are set up to lay on top of a contrasting element. With the variables, these can be further tweaked.

The document viewer that can be accessed through clicking on the cards in the library section is designed to look like a part of the site, with big buttons, placed in an orderly manner. These buttons utilise React and client-side rendering to avoid reloading the page. This solves the problem of having to reload the document on each change. To make the feature more compliant with universal design, and otherwise more useful, it pushes page number and zoom level to the URL, allowing the user to easily get back to the page or share it with others.

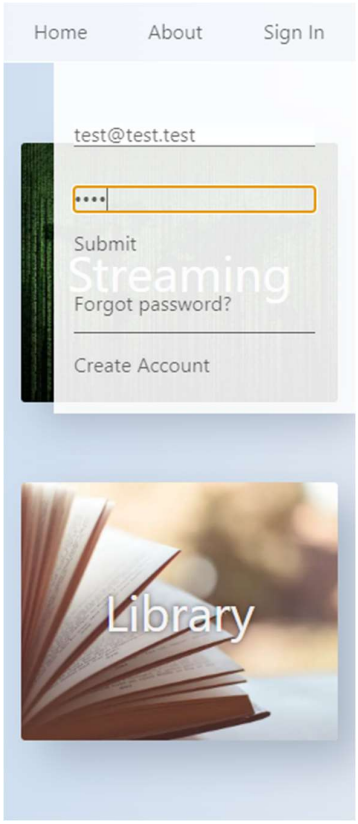

Figure 4.7 - HEQED House on mobile

Further, the document viewer is deemed one of the most important features of the HEQED House. With conversion of the documents to html, it enables rendering of documents inside a potential browser in the virtual application, as there otherwise is no built-in support in the game engine for this.

Delving more deeply into how it is constructed, it utilises the library React-PDF to render the PDF, as it proves to be the easiest to work with together with Next.JS. In parallel with React-PDF, PSPDFKit and PDF.JS are being investigated. PSPDFKit is a paid package, but the attempted implementation does not work. PDF.JS is a library that requires more scratch building and is therefore deemed to take up too much time.

4.1.5 Responsiveness

Responsiveness is an important aspect to keep in mind when building a website, as it is crucial in providing a good experience for the users. To accomplish a responsive website, client-side fetching is used in the library section of the website to let the pages load before the actual data. The data is then fetched on the client, filling the page with document content. The result of this is a faster initial page load compared to if the data was first fetched on the server on each request before rendering the page. This is done because loading thousands of documents means fetching a lot of data, which takes time. The same goes for large files when rendering PDFs. As the library pages were first implemented with server-side fetching, switching to this approach is witnessed as greatly improving page loading times.

Alternatively, it is possible to use incremental static generation to generate static pages at an interval, where fresh data would be reflected on the website. This can easily be done with Next.JS by using the "getStaticProps" function in each page. The benefit of this would be near instant page loading times without any required data fetching from the client, as experienced by the developer team building Illuminem (Ghezala, 2021). This is ultimately the best approach. The reason this is not currently implemented is because with such an approach the free credits provided by the database host would run out quickly, as the database would be queried at a set interval in order to keep the data fresh.

The Header component is also reliant on the user permission being provided to the page component as a prop through server-side rendering. This logic would have to be redesigned on every page using static generation. Meanwhile, as the website has no traffic from users at this stage, client-side fetching causes very few queries to be made to the database. If the website is to generate a lot of user traffic in the future, switching to incremental static generation will not only benefit page loading times, but also limit database querying. It is therefore strongly recommended to be considered at a later stage in the project.

In the library, admin panel, and user publications page, there is implemented search functionality to find documents, organisations, and users. These search functions share the same basic implementation, where content is updated immediately on each typed input. This immediate response is done to provide fast and smooth experience for the user. However, this approach does have a few shortcomings in its current state. For large quantities of items to be filtered, updating the displayed list on each input could prove to be a slow and stuttering experience. A concern is that if the quantity of items is large enough, this approach will lead to the user's input field not acting responsively as the user types. If this is the case, it will counteract the implementation's purpose of improving the user experience. Therefore, an alternative solution is to let the user finish typing their search query and wait until the user presses a button to perform the heavy task of filtering through items.

Another option is to use the react hook "useTransition" to render the page twice according to priority, where the first render will be upon setting the input value from the search query – ensuring the input field is updated as the user types – and the second render will be after filtering through the items. At this stage there is no indication of the quantity of documents, users, or organisations the system will contain. Because of this, no further resources are spent investigating the issue, and alternative solutions are shelved for future iterations.

4.1.6 Error Handling

Because of scope and time limitations, handling of errors is not prioritised at this stage. Many erroneous actions or occurrences are handled by simply redirecting the user to a static error page with a message that states "Woops, something went wrong.". This is not an optimal approach as it does not give the user any sort of feedback regarding what caused the unexpected behaviour, but it does achieve the minimal goal of letting the user know that something did in fact go wrong, and whatever action they were performing did not go as expected.

Some errors are however conveyed to the user. Examples of this include admin actions regarding updating and deleting users, documents and organisations, and users updating their account or inputting incorrect or invalid login details. The actions of logging in and requesting a password reset

are of particular interest, as these pose a security threat if not treated correctly. To avoid malicious actors from gaining users' login details, a uniform message is sent back to the user upon providing either an incorrect email or password while attempting to log in, and upon the user providing a correct or incorrect email when requesting a password reset. Hence, the potential intruder will not get a message on what went wrong – only that something did.

4.2 Virtual Application

The virtual application is the second part of the project and is meant to be a framework for an educational experience centre. The application is accessible through the website by logged-in users. It is intended to be used by students, teachers, and healthcare workers.

4.2.1 Survey

The survey is conducted through Google Forms and shared through LinkedIn and Facebook. To avoid biases and technology cultural similarities to that the developer team might have, the survey is sent to people of different ages and fields of work. While the idea changes during the project, these findings give some insight in behaviour and views on the technology being used in the project.

Despite of reaching over 450 people as of 1st of May 2022, there are only thirteen respondents. It therefore bears no significant value. However, it may still provide some feedback. The full survey can be seen in Attachment "Digital Meeting Survey". Figures show values, where 1 represents a negative answer, and 7 a positive, meaning the questions are asking for the respondents' perceived opinion, and not concrete values.

Starting with topics that applies more to the initial idea for the framework, the survey shows mixed reactions on several aspects. As the virtual application is to initially have the potential as a meeting service, the survey shows that people clearly are using video call services often, and that it has replaced elements of work to a significant degree, as seen in figure 4.8. In other words, the response is strongly skewed towards often, rather than rarely. As one of the initial ideas is to use the application as a replacement for bigger conferences, and the observation is that video calls have replaced and experienced significant amount of work. Especially since 46.2% of the respondents work within- or in relation to health care.

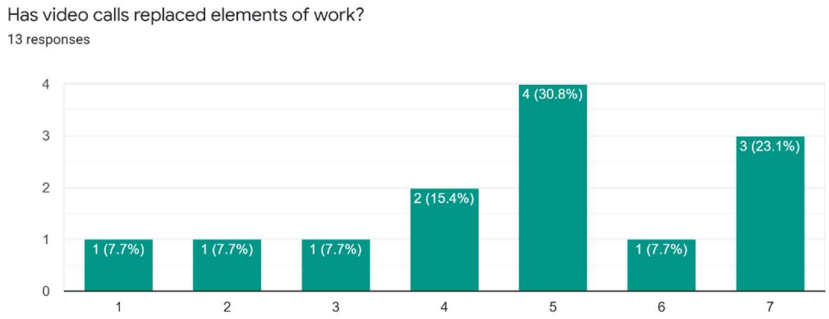

Figure 4.8 - Video conference replacing elements of work

Figure 4.9 - Distractions during video calls

On the questions on distraction during video calls, the results cover the whole scale. However, if this result ratio stays when more people take the survey, it shows that this is perceived as a big problem for a significant amount of people. Following design thinking, it is therefore a clear problem to address. On boredom and tiredness from the environment such calls take place in, the results skew towards the middle, meaning that many of the respondents have this problem to some degree.

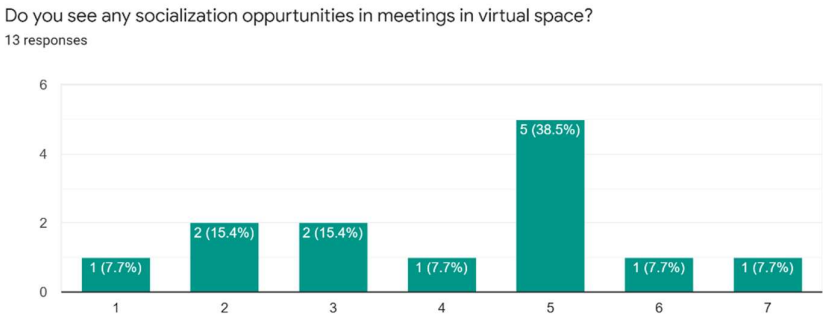

Figure 4.10 - View on socialization opportunities in a virtual space

None of the respondents own virtual reality headsets, while 15.4% have attended virtual concerts. This may show some willingness to experience traditional mediums in new ways. Looking at the amount of people who are open to new socialisation opportunities in the virtual space, there may be a pattern. Especially since 76.9% never use virtual applications as a replacement to physical meetings. This is even more skewed towards a positive attitude on the question of "if productivity features were included in a service for virtual meetings, would it increase your interest?" further strengthening potential openness to virtual spaces as an alternative. With 69.3% of the respondents never or rarely playing video games, this shows that a bias towards technology like virtual spaces is likely avoided.

While the amount of gaming respondents is low, using video games as a meeting platform does occur. This concept is also backed up in chapter 3.5 in the included vision document. Further, the results on respondents' attendance to virtual concerts strongly leans towards never, while a few have tried it.

4.2.2 Unity and Overview

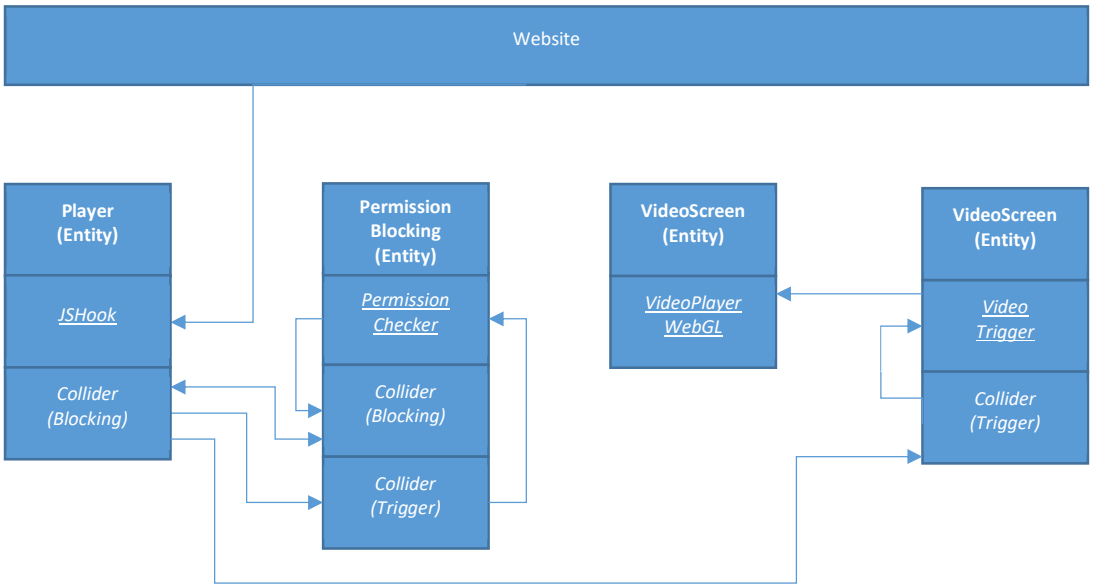

Figure 4.11 - Overview of basic scripts in the HEQED House virtual application

Unity is the final selection for game engine. This framework engine uses a modular component-based system. This means that code is typically added as scripts to an object. In figure 4.11Figure 4.11, scripts are the underlined components, attached to each an entity. The rest are generic components. The way the components can reference each other and other entities, allows cross communication, as shown with the arrows. Colliders are primarily there to stop entities from passing through each other in the 3D space. However, when they are set to trigger, they will let a set volume pass through, and call an event instead – like an infrared alarm sensor. This is utilised in some of the scripts, seen in figure 4.11Figure 4.11.

This differs to the originally chosen engine, Unreal Engine, where the work happens more directly onto each entity and their code. However, the principles are the same. When triggering events, information about the sender can be obtained. This is utilised to establish a connection between the entities. Another option is to hard reference with pre-set variables in each module. In figure 4.11 the colliders, set to triggers, obtain information about the entity that enters its volume. This is used to make the environment dynamic, and filter between types of entities potentially intersecting with the volume.

To prepare the application as much as possible for multiplayer, and otherwise make changes easier, the component system's programming is set up to use as few component inputs as possible. This is similar to the idea of avoiding boilerplate code. The one or few component inputs should instead be traversed to find child- or parent components. This allows for development where the instance of each player's data can be updated- and tailored according to the player's actions. An example is targeting a canvas, part of the user interface, for the player. This is possible because the canvas is a child of the player entity, and not set globally. If a player triggers an event that makes an UI overlay of information necessary, the overlay should only be visible to that player alone, and not the rest. Figure 4.12 shows an example of this, where a function belonging to the canvas – attached to the

player – is invoked. Note that the canvas contains a CanvasActivation, the component that the code is referencing.

```
// Event triggered when a player, c, enters the trigger volume
async void OnTriggerEnter(Collider c) {
        if (c.gameObject.tag == "Player") {
            ps.Play();

            new WaitForSeconds(1);

            c.gameObject.GetComponent<CanvasActivation>().ActivateCanvas();
        }
    }
```

Figure 4.12 - Event when player colliding with a trigger volume

Out of the box, Unity 2021 LTS utilises a new system for controller inputs. Documentation seems to be lacking on this, as the system just got out of beta. When searching for answers, many suggest allowing the application to use both the new and old system. This is attempted but caused crashes. One thing to note is that the old system targets concrete controller inputs, like the keyboard's spacebar. However, as this is an application that is to implement VR, it is important to be aware of controllers being different. Hence, the used solution targets the action the player does instead, so that the same code is agnostic to what the button the input comes from. The exact inputs are instead handled by Unity's systems, which may also be modified when needed. Figure 4.13 shows code where the jump action is tied to an event and its delegate, agnostic to hardware.

```
private void Awake()
    {
        playerInput = transform.GetComponent<PlayerInput>();

        playerAction = playerInput.actions.FindAction("Jump");
        playerAction.performed += DoJumpAction;
    }
```

Figure 4.13 - Input action

Controllers are based on the norm for games – using the keys W, A, S, and D (WASD) – for movement, and mouse/ trackpad for camera pivoting. However, it also gives the user the option of using arrow keys if preferred, aiding in Universal design. While this iteration of the HEQED House does not go into deep research into universal design, especially for the virtual application, it does try to address it. This is therefore built as a platform that can gather feedback on the topic. Hence it is programmed to not let the user move before basic information about controllers are read and acted upon to test the experience. Chapter 5.1.1 will go more into findings, followed up by suggestions for further work in chapter 7.2.2.

Going back to VR, this complicates the topic of controllers, and must be built separately to the web build. While it is possible to let the user use VR headset for web, detection – if possible – for different controllers must be implemented into the website to provide the user with correct instructions. However, this falls out of the scope of providing core functionality: There may be headset and controllers that work for the application, but may have different input layouts, causing potential misinformation to the user.

To make the application run fast, static lightmaps are generated. This means that shadows from given light sources are baked on forehand, and not calculated on the go. If the light is to hit a moving object, the shadow will therefore not change. The light source can, however, be set to light up dynamically, but this requires more graphical power from the users' hardware. An alternative when needed is therefore to use a hybrid system, where some lights have the static shadows, and some dynamic. Since this application is meant to be ran at a as wide as possible number of different computers, this set to purely static. DXT compression, a GPU compatible industry standard, is used to compress both lightmaps and other textures. This aids in not requiring more GPU memory than available.

Further performance savings come from using universal render pipeline (URP) instead of a more detailed, high-definition render pipeline (HDRP) in Unity. This means going for a simpler rendering pipeline, instead doing extra work of probing for reflections, and often lowering precision on features both pipeline supports. As an example, URP does one pass for generating real-time shadows, compared to HDRP's ability to do multiple. While WebGL doesn't support HDRP, it is still important to be aware that URP can also become heavy to run due to the graphical effect. When compiled to web assembly, it is comparatively slower than when compiled as a desktop application (Unity Technologies, 2022). Hence, only ambient occlusion is activated.

Due to the use of WebGL, mobile is generally not supported, while it may work on some, according to Unity (Unity Technologies, 2022). This can be solved by adapting and building the application for mobile, which is supported in the Unity editor.

4.2.3 Database and Content

While the original idea is to utilise the main database for providing content to the virtual application, when building the application for web assembly, it is discovered that the Unity game engine requires files to be placed in a directory called Streaming Assets, placed in the root relative to the application. Further, the virtual application must be placed in the public folder of the website, allowing users to access files without having to forego any login or limiting measures.

This limits the flexibility to provide content dynamically. While it is still possible, it is determined that it requires further research to find solutions, including one that won't make the user have deal with any form of mitigation of this problem. This is further complicated by the Unity game engine when building for a standalone VR application. During this build, files are required to be present during the build process, which happens on the export of the virtual application from the development tool. This means that files will be compressed and/ or packed into custom files. The developer team does not know how to do this dynamically or in real-time with the VR build. Hence, the current solution is using a predefined scene, where the content is set for each build of the virtual application.

However, when building the application for web assembly, files are stored as conventional file types, like mp4, indicating that a dynamic solution is easier to do in this instance. Connecting to the right API on the web application, Unity can fetch data on which files to load. Hence, with the potential implementation, the administrators may change the exhibition. If this is to be done, it is important to be aware of the aforementioned differences in the build types in terms of how the files are handled.

4.2.4 Networking and Alternatives

The survey from chapter 4.2.1 shows that people are interested in alternatives to attending physically. Hence, the general accessibility to the application is considered. This goes back to the idea in the included vision document about streaming the virtual application and making it multiplayer.

Unity's Netcode is selected to provide the multiplayer part. Netcode features both the ability to create a client-side server, as well as a dedicated server. The player entity is referenced by the added instance of Netcode, which then distributes an instance to each user. As mentioned in chapter 4.2.2, this is where it is important to separate entities and data correctly, so that relevant data is only visible to the relevant player. Other than this, the application is built as if there is no multiplayer, with some exceptions regarding the playing of in-application videos, which this chapter will discuss later.

Streaming is another feature that Unity allows with extensions. This means using a dedicated server to run the application and stream it through WebRTC to the user. This means that the user does not require any special hardware, but an internet bandwidth good enough for the stream. However, as the current iteration of the HEQED House has no budget for servers, and otherwise a budget that prioritises other posts, these features fall through. The multiplayer is still possible, but time constraints and other discoveries makes it a high-risk feature to implement. Netcode is also an experimental feature that must be downloaded via Unity's GitHub.

A solution, which is strongly believed to lower the cost, is to make the virtual application a part of the website itself, through using WebGL and web assembly. This means compiling the code as web assembly, and launching it through the site, running it on the user's computer. This should lower server cost in terms of both upload and running the virtual application. On the other hand, it demands slightly more from the users' hardware. However, focus on making the application lightweight should make it runnable for most. The performance savings come from the light baking and – in cases where the GPU memory is low – from texture compression, mentioned in chapter 4.2.2. Bigger performance impacts will come when the artistical detailing part of the application is done. More on this in chapter 7.2.2.

One performance problem may be playing multiple videos at the same time. This will happen if multiplayer is implemented, letting players globally start the running of each video. This is a potential showstopper, and a reason why this feature is delayed. Despite this, the code in script *VideoTrigger* is prepared for the case where multiple videos playing is okay. This is done by counting the players in the entering volume that starts the video, as seen in figure 4.14Figure 4.14. The video player is hard referenced, meaning that each trigger volume that is connected to said video player, has an instance of this code. Hence, it is not a global trigger. This code will also not reactivate a video if another player is already watching. However, as the media files are put in a streaming folder with the WebGL build, it may be that the content is played individually for each user. This needs to be tested if implemented.

Another potential problem with multiplayer is the need for a server. As there is no budget for it, the option falls on making a local server. This is believed to require extra systems for handling the sessions, like a planned meeting for a video conference. Another issue is that the user who gets selected to host the session may not have hardware capable of the necessary performance.

```
void OnTriggerEnter(Collider c){
        if (c.gameObject.tag == "Player") {
            playerCount++;

            if (!videoPlayer.active) {
                videoPlayer.SetActive(true);
            }
        }
    }

// similarly in OnTriggerExit(), with playercount-- and the following:
// if (playerCount == 0) videoPlayer.SetActive(false);
```

Figure 4.14 - Video activation and tracking of players watching

4.2.5 User Experience

Going back to chapter 4.2.1, the survey shows distractions during meetings. While this is not a video conference, it shows that it is important to engage the user and catch their attention. Therefore, a simple minefield is implemented to provide some shock factor and see how the user reacts. This is something that refugees experience, making it potentially provide a strong, educational experience for the user, in line with the vision of the project.

This is an example of content that may not be suitable for all audiences, and therefore utilises a blocking volume to stop unauthorised users from entering. This is done by using a collision box that collides with the user character's own collision box. However, by the website sending a message with user permission to the virtual application after loading is done, the collision box is deactivated if the user has logged in with an account with high enough credentials. An example of this blocking volume, and explosion from the mine, can be seen in figure 4.15. Note the "Permission Level" and "Given Permission" in the panel to the right. The given permission is updated by the message from the website.

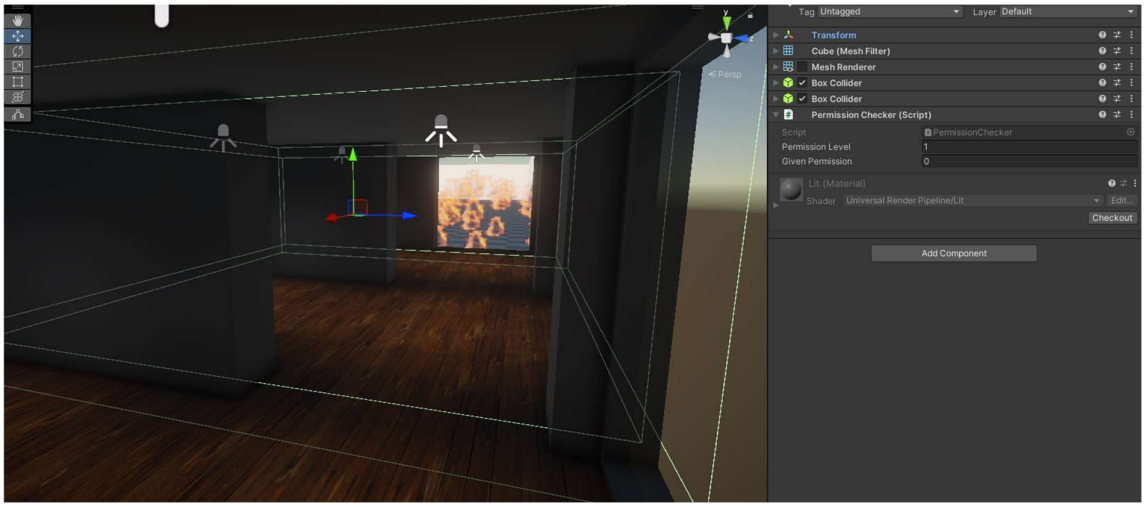

Figure 4.15 - Blocking volume and permission system

As mentioned in chapter 4.2.2, when the application start, the user must read through the initial message, and push the jump button to continue. These five buttons, plus the movement of the mouse are all the controls the player needs use the application. It is not only the standard setup for Unity and general game industry, but also there to make the application as simple as possible for what is needed. Including when the player steps on the landmine, and needs to confirm to respawn, after having read through the short message. The latter is meant to be a message, customised to the HEQED projects liking. The controller instruction should also be done for VR controller, but it may also require more detailed drawings of the controllers. The VR controller are not as easily represented by letters as on a keyboard. Hence, the current focus in on keyboard and web application that is available to the public.

The videos are set up to automatically play once the player steps into a logical area, the semi closed space the screen for the video is placed in. As mentioned in last paragraph, the automation is done to keep the controllers as simple as possible. The automation is shown in Figure 4.14.

4.3 Product State and Limitations

The scope of this project changes over time. Notably, the product goal goes from being more limited in scope and a more feature complete application – which will allow meetings to take place in a virtual space with sharing of documents – to being a broader concept. The product is now intended to be a framework consisting of two parts: A virtual application which acts as an experience centre for healthcare workers, teachers, and students – to allow for an immersive and interactive learning experience – and a website which hosts a document library and acts as a portal to the virtual application.

This shift in scope largely impacts the finished state of the product. It becomes clear that because of a strict time limitation, the product must make compromises to which features, above the most essential, will be implemented or completed. Therefore, some features are either left incomplete or shelved entirely.

One lacking feature is users being able to join or associate oneself with organisations. Still, the database allows for a user to be related to multiple organisations. However, the library does not take

this into account when fetching and displaying documents. It will require a lot more work to first implement the library in such a way that it adjusts for varying numbers of organisations associated with each document through its owner, as well as implementing a way for users to either join organisations or to allow organisation leaders and admin users to add users to an organisation. As the library functions in its current state, and other admin and user tools are implemented and working as intended, this feature is not seen as essential and therefore not prioritised as the project nears completion.

Another note to make is the lack of quality control and polish. As the project continues, the initially allocated time for polishing is instead used to finish more crucial parts of the framework. Consequently, much of the product is in an unpolished state. This can be seen by the lack of validation on some user inputs, as well as the database structure not being fully up to date with current needs and ideas. There are also security and performance measures missing, such as limiting the number of documents a user can upload and limiting how many times and how often a user can request a password reset. These are weak points against malicious attacks wishing to overstrain the server by bombarding it with requests, as well as filling the database with garbage data.

Parts of the front-end also require further work to function better on a wider range of devices. Currently, some parts of the UI are best suited for high resolution screens with an aspect ratio of 16/9 or similar and have not yet been updated to follow the guidelines mentioned in 4.1.4. The CSS files containing the stylesheets are currently somewhat unorganised and unoptimised, creating another source of lacking quality control and polish.

4.4 Promoting Health Equity

4.4.1 Experience Centre

The experience centre is there to engage users in a virtual environment to further their understanding and empathy towards people in difficult situations. This is the main focus of the virtual application. While features like streaming and multiplayer are not implemented, streaming is replaced with an alternative that may work even better, letting the project owner reach a broader target group on a smaller budget. Multiplayer is kept in mind, but is put aside in favour of getting the website deployed for testing, as well as other features.

While viewing 2D content like videos in a 3D space is a part of the requirement, the framework also implements more dynamic content like the mines and necessary component to create a complete user experience, as mentioned in chapter 4.2.5. The goal of letting the project owner try out different things in the future.

4.4.2 Open Library

Open and accessible information is inherently important in the context of health equity through education. The library contains all published documents from users. A user does not have to be signed in and authenticated to enter the library page and freely view public documents. The library has a simple design, with easy-to-use search functionality, with the purpose of making finding and viewing documents as easy and accessible as possible. It does not, and should not, require much technical knowledge to find information, broadening the audience group. All PDFs can be viewed

directly in the web browser or downloaded to the user's device. All this combined is meant to create an easy to navigate library of open and free information.

As finding and consuming information should be an accessible and easy task, contributing to the library should follow the same principles. From the library page where users can find documents, they can also find a simple layout that requires minimal input for uploading documents. To upload a document, the user submits the document file and enters a title and description of the document. The user then decides to either leave the default option of making the document publicly available checked or unchecks it, and then uploads the document by clicking a button. A caveat here is that a user must have a verified account to upload documents. This is to avoid uploading of unwanted and malicious material. Such material could for example be pornographic imagery or depictions of graphic violence. In an ideal setting all users would be able to contribute to the library to make it truly open for the public, but this measure is a necessary compromise to ensure safety.

5 RESULTS

This chapter describes how the evaluation of the result is done. It is described which types of evaluation methods are used to ensure the quality of the project and the evaluation results.

5.1 Evaluation method

Throughout the project different evaluation methods are used. The first is continuously running and testing the website and virtual application when changes and new solutions are implemented. Second method is to present the project owner with the current version of the website whenever a physical or online meeting is held. Third method is to get evaluated by deploying the website online. In this way the website is easily accessible for testing for both the project owner and others.

5.1.1 Evaluation by Code Testing

Continuously testing the website is used to ensure that changes and newly implemented solutions work as intended. This method is used throughout the entire development of the project and plays a pivotal role in ensuring the quality of the website. By using this method each developer can help evaluating each other's implementations and alert each other to discovered faults. This creates multiple layers when it comes to both quality control and security. As mentioned in the risk assessment in chapter 3.4.3, this type of testing is used as a measure to avoid different risks from occurring.

One example of testing is the use of the program Postman, which can be used to test the APIs of the website. At a later stage, developer tools in Chrome and Firefox take over to test further implementations of features like cookies, or behaviour of React features. Using developer tools in a web browser is also done to ensure that the website works on mobile phone aspect ratios.

5.1.2 Evaluation from Project Owner

During meetings the project owner and supervisor are presented with the current version of the website, and ideas and thoughts on future development. This gives the opportunity for the project owner and supervisor to observe, test, and evaluate the current state of the website. From initial feedback, the website will gradually be adapted to the project owner's needs and wishes. This method is very important as it ensures that the development and ideas for the website are aligned with the project owner's vision. In the beginning, the website can only be run locally on a machine whenever a meeting is held. At a later stage the website is deployed online, making it easier and more accessible for the project owner.

For the final evaluation, a meeting is held on the 23. May 2022 with the developer team, Clark Nowack, and Maria Nordheim Alme. Nowack and Alme represent the project owner and are given full access to the website and virtual application. Alme cannot stay for the entire meeting, but initial impressions seem mostly positive, although some issues do occur while accessing the virtual application in relation to moving around and transferring of the correct permissions from the website to the application. The meeting is ended, and Nowack is given more time with the product and some questions for the evaluation, which he provides answers to on the 26. May 2022. Following are the evaluation questions given and the comments provided by Nowack.

Website Evaluation	**Comments from Clark Nowack**
How intuitive is navigation and placement of user controls?	Found it really easy to navigate. Managed to upload a PDF.
How is the user experience regarding loading times and performing actions/inputs?	Loading times and typing feels okay. The ”Public” button is not explained, but feels logical.
What kinds of feedback on actions/inputs does the user expect compared to what is received?	Got no feedback when entering a wrong password. Reset of password works well.
Which actions can the user not perform successfully?	Searching by organisations causes an error during one of the runs. Another error message when leaving the virtual application.
Additional comments	Would be nice if you could see the first page of the document.
Virtual Application Evaluation	**Comments from Clark Nowack**
How intuitive are the movement controls?	The letters W, A, S, D do not feel logical. F for moving forward, L for moving left etc. is more logical. Did not understand “Jump”, difficult to understand for someone who does not play video games. Moving around with multiple buttons, spacebar, touchpad and mouse requires more experience. Not sure whether the lots of different future users, with different IT literacies, can work with it. Walking through different rooms can have more information. What is exposed in the different lanes?
To what degree does the user encounter unexpected behaviour in the application?	Video sound can be a little bit louder.
Which actions can the user not perform successfully?	Everything seemed okay.
Product Evaluation	**Comments from Clark Nowack**
How does the product differ from initial expectations?	In relation with the Vision Document, it is expected more functionality from both the website and the virtual application, but it is understood that more work must be done. It is important that the next group of developers know what is covered and what needs to be done.
Does the project owner see value in the product?	There is value to the product. Some work needs to be done still, but that is outside of this group’s scope.

Can the project owner use this product in place of, or, in addition to current solutions?	The basis is quite new and useful for this project.
Does the project owner wish to continue the HEQED House project to further develop the product?	There is a wish to continue the project.

5.1.3 Evaluation by Deploying

Deploying the website online gives the opportunity for the project owner to test and evaluate the website whenever it is requested. This makes it easier for external feedback from external users. Since the project owner lacks competence in software development, receiving feedback from external users with expertise is very valuable. As an example, contacts with expertise within user interface design development is used to evaluate the design. An online deployed website will also provide a stable database that can be used for testing of loading content in the virtual reality application, with its experience centre.

5.2 Project Owner Evaluation Results

Using these different evaluation methods is critical for the development of this project. The results of the evaluations show a website and virtual application that can be further developed. The website is up and running and the most critical functionality is implemented. This discussed in detail in the following paragraphs of this chapter.

The answers given by the product owner, represented by Nowack, give valuable insight into what state the product is in, how it compares to the project owner's vision, and how to proceed from here. Regarding the website, it is found that navigating the site and performing various actions and inputs seems natural and easy enough to use. Most actions seem to work as intended. During testing, Nowack does not receive any feedback on entering a wrong password. This is an implemented feature and can therefore be categorised as an error and should be investigated. It is also discovered that searching by organisations on the library page causes an error. Another error observed by Nowack occurred upon navigating to the home page whilst loading the virtual application. The last two issues mentioned are since fixed. This indicates that there are still issues to be found and more polish to be done. An additional comment was made about the possibility of showing the first page of documents as a preview in the library page.

As it stands now, documents are presented in a very simple fashion, with an organisation name, author name, title, and description. It is specified by Nowack that this feature will be for the next group to implement, and this is therefore of note to future work.

When entering the HEQED House virtual application, the user is presented with text explaining the different controls. Nowack does not find these controls to be natural or intuitive. He comments on the jump feature, explaining that this is not something he is familiar with, but thinks it is common in video games. Nowack continues explaining that he is unsure users with many different backgrounds will be able to work with the many controls for moving around in the application.

This presents a very important point: The divide between users with different backgrounds and experiences. The keyboard keys W, A, S and D (WASD) for movement, and mouse or touchpad for

camera/ looking direction is the standard for computer games. Nowack gives an example of controls he finds to be more intuitive, where F is for forward, L for left, R for right, and B for backwards, where the action's first letter corresponds with the letter of the key. The problem with this approach is that these keys do not match the natural positioning of the fingers of most people using a keyboard. WASD is standardised and used precisely because of their positioning on the keyboard, and how it mimics the arrow keys: W is at the top and moves you forward. S is at the middle-bottom and moves you backwards. A and D are left and right, respectively. Most people are right-handed and subsequently use a mouse or laptop touchpad with their right hand.

The virtual application also allows to move with the arrow keys, which should account for users who are left-handed and would use their right hand on the keyboard. The current layout of controls feels obvious and natural to the developer team because of having a lot of previous experience with it, but from the comments presented by Nowack it becomes very clear that this is not the case for every user. This represents a clear divide between different users, and must be accounted for, either by simplifying the controls and/ or explaining them better.

Another comment regarding the virtual application is that the different rooms can be better explained and provide more information about what they contain. At this point the virtual application is a barebone framework and has very limited content. Going forward it is meant to be filled with more rooms and experiences. The point about what these rooms contain being better conveyed to the user then becomes increasingly relevant. Furthermore, the questions about unexpected behaviour in the application and actions not being performed successfully reveal that there are few issues in this regard, but also that the audio of video can be slightly louder. It should be noted that audio volume is typically possible to change on the computer by the user themselves. At the same time, it underlines the importance of editing content properly before publishing.

Lastly, the group seeks an evaluation of the product as a whole and its usefulness to the project owner. On how the result differs from initial expectations Nowack answers that based on the vision document, which is provided as an attachment to this report, he has higher expectations to how much functionality both the website and the virtual application should have. However, he goes on to explain that he understands there is a lot of work to be done, and it could not all be finished during this project. It is therefore important for the next group of developers to get an overview of what is done, and what needs to be done.

Nowack states that there is value to the product. More needs to be worked on still, but this is outside of this developer group's project scope. Furthermore, the product is something new and useful to the HEQED House project going forward. Finally, Nowack states that they do wish to continue the HEQED House project and further develop the product.

5.3 Overall Results

5.3.1 Website

Testing by the developer team shows that the website behaves as expected, and the detected bugs are dealt with. After deployment, it's also tested on mobile, where the site works well. The only feature that doesn't work on mobile is the virtual application, but this is due to lack of support of WebGL.

The website is seen to provide simplicity and being easy to navigate. It lets project owner members register and become administrators of the site during testing, to then upload documents, fulfilling some of the most complex features in the requirements for the website. While most requirements are fulfilled, the HEQED Project desires more document types to be shown. This requires paid extensions or significant development efforts. Chapter 6.2 will go more into the possible causes and solutions to this.

The issue related to the virtual application and searching by organisations is reproduced by the developer team, otherwise none of the other issues are reproduced yet. It must not be dismissed, and instead be investigated further, as it can indicate a user interface that is not optimal, another lacking optimisation, or simply be a bug.

5.3.2 Virtual Application

The virtual application is the one that requires the most development further, and the one that has the least amount of work done to it. This shows in the evaluation. At the same time, it is seen as new and useful. Decisions on how to implement alternative solutions to streaming, seem to have affected the project positively, as it provides relatively accessible and easy navigation.

The project owner sees that the application provides value for the HEQED project, which in combination with it being seen as new, indicates that it may be an innovative product. As the current iteration is a framework for future iterations, further feedback shows that the basis works, but also that expertise outside programming is required to build the application. This is further underlined by the need to accommodate a wider range of users with different levels of IT literacy and experiences.

5.4 Project Implementation

The implementation part shows different difficulties in implementing some of the features, and further limitations due to lack of resources. This is mostly dealt with ongoingly during this phase.

Where the limitation is due to lack of resources, free extension alternatives or mock example files are used. The same applies to the virtual application, where content often must either be made by hand, or a free alternative used. The focus falls on coding and using free, ready-made assets or mock-ups in this case. In the virtual application, this may contribute to it looking like a very unfinished product, but also falls back to the emphasis on this iteration of the project being a framework.

The virtual application is also a product of severe time constraints. Unreal Engine, used during the initial stage of the project, requires a source control management service that accepts very large files. This is hard to acquire for free, and the lighter Unity Engine ultimately takes over, after a reset of virtual application following the gathering of the project owner members on 24th of February. The Vision Document is redone according to the new plans from this meeting. The remaining part of the project acts upon this, in respect to the time that is left at the point.

With usability, overall project goal, and a tight budget in mind, potentially better solutions are found during implementation. The implementation of the virtual application into the website itself is one example of this, eliminating the cost of using multiple servers to run the virtual application, which chapter 6.1.2 goes deeper into.

6 DISCUSSION

6.1 Product

6.1.1 Website

The backend of the website shows to be a good fit for the project. With Node.JS providing a runtime that is compatible with a significant number of extensions, it makes it easy to find suitable solutions for occurring problems. While Next.JS is a relatively fresh framework, it builds upon React, which is one of the current industry standards for frontend. This combination of well tested parts and Next.JS's more forward leaning backend, enhances the development time significantly. It also makes implementing the virtual application into the website relatively problem free. While there are other limiting factors when doing the latter, it still shows that it is possible to extend with features that are not often seen on, or associated with, websites.

An example of Node.JS working well for the HEQED House, is that it not only provides a very scalable runtime for the website itself, but also the virtual application. If the traffic gets too large, Node.JS works asynchronously, allowing for horizontal scaling and making it technically easy to put into an educational context. In addition, the virtual application can simply be deployed together with the website and run. Because of this, the integration of the virtual application within the website is seen as a success. Furthermore, Next.JS lends itself well to a serverless hosting model, which removes much of the concern about the server, provides easier scaling, and lets the developer team focus solely on writing code and developing the product.

Chapter 6.1.2 will go more into the application itself, but for the website's general user experience, it provides a complete experience with one portal for the whole project. This lets the users access information without having to access the virtual application first, making it more accessible than the original idea and project description. It also covers a wider range of needs, later proposed by the project owner.

The downside is that the focus gets significantly bigger, and fewer features get time to be polished or fully realised. At the same time, the website also needs to be further tailored to the content the project owner wants to publish. This must happen when the content is ready, which it is not at this stage of the project. Therefore, the current delivery is seen a good start.

Though some issues do occur as a result of having to rely on free credits, the use of Google Cloud as a platform for the database and file storage is seen as a good choice. The APIs to implement these services on the website are simple and easy to understand, and work as expected. There is however no guarantee that Google Cloud will be used as the database and file storage provider further in development. Hence, some degree of modularity has been accounted for to allow for easier migration to other platforms. This proves to be important when a necessary switch to Aiven and Firebase is done during the last stage of the project. The database shows to be easy to migrate to other platforms, as it mainly relies on environment variables, while file uploading and downloading demands some more work to function with the new platform, and can be improved upon. More on the setup can be found in the included *System_Documentation_DAT191* document.

Lastly, there are some lacking qualities to data structure and the page rendering methods. The database structure has gone through multiple iterations, but no major overhauls since its first

inception. Most noteworthy is the fact that there is a disconnect between what the website expects the link between members and organisations to look like, and what the database provides. This is only an issue in relation to the library page, which wants to display the associated organisation name with each document. It does not know how to compensate for no associated organisation, or multiple. This will have to be addressed in the future. Additionally, static generation is currently not possible on any pages because of server-side rendering being relied upon to deliver the necessary prop to render the Header component. As explained in 4.1.5, static generation will greatly improve page loading times, and reduce database querying under heavier traffic. It is therefore also worth looking into refactoring the Header component to not rely on server-side rendering.

6.1.2 Virtual Application

During development, unforeseen challenges are discovered. As mentioned in chapter 4.2.4, both multiplayer and streaming is dropped. This proved to potentially be a good choice, as evaluation indicates that the focus needs to be on getting the users to feel more comfortable with the controls and orientation. Multiplayer will not solve this, but rather introduce development delay and more disturbances to the users.

Implementing streaming in a way that lets user easily test and evaluate the application is unpractical. With no budget for servers, the solution is to bring a computer with a locally running development build to the end-users. The choice of implementing it with the website instead, enables testing at a much larger scale, as the link to the website is all the user needs. Compared to streaming, it may also be a financial benefit to the project owner, as it is cheaper to run and seems to solve enough of the problems. Furthermore, it is easier to operate and develop as it uses the same server as the website, requiring no special routing.

In comparison to initial expectations and visions of the project, the pivot and result is different. Unreal Engine speeds up the visual development significantly, making it easier to make a pretty scene. The necessary extensions seem to be more mature and production ready, too. An example of this is an in-engine web browser, which is seen as necessary to share and present documents within the virtual application. The same goes for Unreal Engine's multiplayer framework. However, Unity works well for the current iteration, and can be built into the website, which Unreal Engine currently cannot.

Another reason for dropping multiplayer is that a clear, acute reason for having it, has not appeared. Multiplayer makes sense in terms of having an experience together with someone, and potentially having contact between clients and health personnel. However, the latter conflicts with GDPR and legal information protections the users must have. Relaxing the security will likely require a one-on-one solution, where others are closed out, making it counterproductive. This may still not be enough, as extensive bug testing and validation must be done to secure this, no matter what the solution is when involving this kind of sensitive data.

Going back to controllers, and as mentioned before in this report, universal design must also be a part of this application. During the evaluation, some of the testers have problems with orientation and starting to use the application. This is despite the controllers being set up the traditional way. However, as mentioned in chapter 5.1.2, there are differences in how familiar people are to this way of interacting with a computer. This means that universal design is lacking or should have a stronger focus. In addition, when the standard way is not good enough, this means that some research and

further user testing is required. While it causes a limitation for the player, an idea is to limit the angle up or down the player can look. The reason for this idea stems from the tester looking into a unicolor roof, unable to identify that it is a roof. Hence, the tester gets lost, despite only having to move the camera slightly, either by mouse or by touchpad movement. This again points towards the differences in what users are familiar with.

This also highlights the need for an improved visual aspect of the virtual application. The user needs more visual cues to orient him or herself – which another tester also points out – in the context of using the application in a learning setting, where exhibition pieces need to be labelled. These problems partially fall outside programming and may be better addressed by developers with more expertise in visual design.

The feedback regarding using the application in an educational setting may hint at the need for multiplayer functionality again. For multiplayer to make sense, audio- or even written communication may also need to be implemented. This is where it is important to not let the project escalate to something that is too big per iteration. Especially when combined with development of a website.

6.2 Development

Working with a clear idea and problem to tackle proves to significantly increase the speed of development. Even when the idea might not be a correct one. Hence, the communication and preparedness are important from both parties to get the development going. As an example: During development of the original idea, the tasks are clear. However, this did not align with what the project owner ultimately needs. Recognising this requires more effort into researching how to handle this kind of situation: A client that is not exactly sure about what they want or need, should still be able to have their problems solved.

During development, some members of the developer team have become consultants in tech corporations. The cases in the consultancy industry can be very similar, but typically have a more thorough pipeline that this project should aim to follow examples of. This includes signing off on requirements and working more closely with the project owner. The latter is attempted in this project, but should have been done even more proactively. However, this is realised too late in this iteration.

When the project goals are not clearly defined, it shows that agile development is important. The clear requirements need to be addressed first, as others may be thrown out. In other words: Prepare for- and welcome change. Hence the website, where the requirements are much more established, becomes the focus during the period of uncertainty. This is also the case regarding the promised server, where eventually it is found that the budget does not allow for it to be provided at this stage, requiring other solutions to be found.

The main problem is the virtual application, which has had its needs and purposes questioned multiple times. Partially due to lack of IT literacy and VR headsets not being a typical item most people own. Therefore, the survey is conducted – simply to find different views and opinions on use cases that may fit in this project. During 24th of February, the brainstorming during the international gathering of the project owner members in Bergen proves very productive. It becomes an arena for

ideas, previous experiences, and direct feedback on ideas, where the final idea for the virtual application is shaped for this iteration.

Ultimately, solving these challenges from the beginning, will likely lead to a more feature rich, more polished set of products, without the need for creating a second vision document. The vision document is shared with the HEQED group to work as both a short- and long-term vision for the HEQED House. After getting feedback from testing, seems the vision document's long-term vision may give some false, over ambitious expectations of the short term product. This highlights the importance of better clarifying and planning limitations for- and with the project owner.

6.3 Workplace

The HEQED House is being entirely made through home office. While this is a luxury in the sense of work hour flexibility, no travel, kitchen, privacy, and more, there are consequential challenges of this. The biggest issue may stem from general communication. People are not necessarily working simultaneously, causing delays in feedback and problem solving. This again leads to lower work motivation and -morale, as the perceived social aspect may be lost.

On the other hand, being able to work flexible hours solves some problems the pandemic causes. The lower tolerance of showing up to work with symptoms causes part time jobs to demand developer team members to work extra. This causes additional stress on other developer members. Therefore, while habits may have changed during the pandemic, it is still important to establish some routines for work. It is not a given that work can easily be delegated, and the result may be several developers working on parts of code that relate to each other.

It seems the project owner has adapted to the COVID-19 pandemic and works internationally. This is a group that has worked together before, and who works in a field where the understanding for each other seems stronger. During the 24th of February meeting, the developer team sees that there is a gap between itself and the project owner. Physically working more closely with the project owner may increase the ability of getting feedback and establishing a common understanding of the different parties' needs. This means the project owner needs a clear case to be solved, and the next group of developers need to find the correct solution more proactively. Design thinking is a good example of a process that can help with this. Achieving some level of this may strongly increase the ability to create a useful product. Especially at a stage where the ideas and requirements are not clearly defined.

7 CONCLUSION AND FURTHER WORK

In this chapter there will be a conclusion of the project and discussion of potential further work. This includes a conclusion of the results, problem, and the requirements from the vision document. It will also include what the developer team would do with the possibility of further work on the project and suggestions to other groups that could potentially take the project further.

7.1 Conclusion

The developer team has been able to develop a website using Next.JS and a framework for the virtual application using Unity. The website consists of file-uploading, library functionality and information about the HEQED-project. On the website users can navigate around to access the library to read papers that are published, and in the future read about who the HEQED-group is and what the HEQED project is about. The website also acts as a portal to the virtual application. The virtual application is a framework for now, but is meant to be further developed and give users interactive and immersive lessons and experiences.

The product of this iteration of the HEQED House project contains examples of content that can be further extended on, including more dynamic content that can be adapted for different scenarios. Some code adaptations are necessary, but it is mostly about importing new content, and adapting the exhibition to said content.

The goal is to create a fully functional website as well as a fully featured virtual application, but because of different circumstances some of the work had to be cut out. At the same time, potentially better alternatives are found, and other discoveries documented. This aids the continuous development of the HEQED House, potentially further improving it. The project owner also sees the products as innovative, meaning that the end goal of finding ways to improve health equity is a possibility with this project.

7.2 Further Work

7.2.1 Website

While the framework for the website is present, there is a need for more work on the frontend. For this, it may be a good choice to get hold of a team with more experience and education in work with user interfaces and universal design. An example of an important feature to implement is aria labels, to make the site more usable for people with visual impairment. The implementation of this can be done with translating the site. A suggestion is to make a file that contains the strings for the different elements of the site. Therefore, it may be well organised to combine these with belonging aria labels.

If the site is worth investing more than the minimum, and file viewing solves a more significant problem, there are extensions that, with some knowledge of JavaScript, are easy to implement. These can offer far better viewing experiences, including allowing viewing of different file types. However, the quality of the extension may differ, and may therefore not offer the same adaptation to universal design in some cases. Hence, further investigation must be done.

The cost of running and maintaining the website must be evaluated. While the work in this project does focus on keeping the costs low, there may be some aspects that are not caught with current

testing, and improvement. This also applies to security. While the current site utilises strict APIs for much of the communication between user and server, there still may be loopholes. One of these may be taking measures against potential DDoS attacks. This is, however, something that the server provider normally takes care of. Due to no control of how many times password resets can be requested, a known issue is potentially causing a victim user to receive an excessive number of emails.

7.2.2 Virtual Application

The virtual application, as a framework, is meant to be further built upon. It shows possibilities of viewing different content, whether it is videos placed as an exhibition, or more traditional game-like content, typical for a game engine.

As mentioned in chapter 5.3.2, a solution to allowing individual players to pass the blocking volumes is not found. The current system has the credentials attached to the player. The conflict with having a multiplayer feature occurs when one player with the correct credentials to pass deactivates the blocking globally, allowing the rest of the players to pass as well. This means that the player system is prepared for multiplayer but requires a different or enhanced blocking system.

At the same time, the player system is a placeholder, though prepared for multiplayer. Unity is also working on a high-quality character system (Silvia Rasheva, 2022) that competes with Epic Games' Metahuman. This may be worth considering in due time, but it must be noted that it is of a quality that often requires high performance computers. Meanwhile, Epic Games' Metahuman allows for qualities suitable for low performance computers, or even mobile phones. Unity may choose to compete in this regard, as well. A different alternative of sufficient quality will likely not be free or will require a significant amount of work from skilled people.

The multiplayer system provided by Unity is in an experimental stage, where it is not even published as a preview on Unity's website, but rather must be downloaded from GitHub. Therefore, it is recommended evaluating the need for this feature at a later stage and consider if waiting for the technology to mature is worthwhile. However, it must be noted that if a multiplayer system is to be integrated, it should always be kept in mind from the beginning. Full control of where the code should run independently for each instance of the application or on the server is required for a well-functioning multiplayer application.

As the virtual application is built using a game engine, it is natural to take advantage of this, and add more dynamic content. This can be a fully interactive experience that simulates the experience of being a refugee to a better extent. Here it must be noted that the project then evolves from being programming oriented, to requiring 3D artists and animators. This type of work can be very labour intensive, and therefore evolve to be a costly project. With Epic Game's focus on providing 3D assets and functionality for free, it can therefore be worth re-evaluating Unreal Engine as the framework used for this part of the product.

The universal design in the virtual application must not be underestimated. Controls are very important to the user, and the target audience may not be familiar with interacting with a computer in the required way. The virtual environment also needs to be easy to orientate oneself in. Hence, extensive user testing must be done.

7.2.3 Server, Database and Storage

As it stands the website and virtual application is deployed using Vercel, connected to an Aiven hosted Database and using Firebase storage. These work well for now since the website is not fully operational, giving the project owner the opportunity to test and evaluate both the website and virtual application in the final stages of this project iteration. If the project is further developed there will at some point be necessary to upgrade the server, database, and file storage.

The current server is a serverless solution that scales well, meaning that it should be able to handle a lot of traffic. It handles both the website and virtual application. This is great for the development and testing phase, but if the website and application is to go fully operational this could cause problems and performance issues if there is very high traffic.

The current database and storage are free, limited solutions. At some point the database and file storage will eventually run out of credits and storage. This means that new free solutions must be implemented or there must be investments made from the project owner for paid versions. Either way, if the project owner wants to make this a fully operational website and virtual application, upgrades to the server hosting plan, a database without limitations and a scalable storage is necessary.

7.2.4 Further Work for a New Group

The entire project is being documented with a final report, vision document, project manual, system document and requirements document. In addition to the documents, the code for the project is public and can be found at GitHub with instruction. This provides a new group with all the information, instructions, and code they require to continue work on this project.

Even though a new group will be provided with all the documents and code, as mentioned in previous chapters, it is critical that both the project owner and developer team decides on both content and concrete goals for the virtual application before taking this project further.

8 References

Agile Alliance, n.d. *12 Principles Behind the Agile Manifesto.* [Online]
Available at: https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/
[Accessed 27 February 2022].

Anon., 2021. *Introducing The Matrix Awakens: An Unreal Engine 5 Experience.* [Online]
Available at: https://www.unrealengine.com/en-US/blog/introducing-the-matrix-awakens-an-unreal-engine-5-experience
[Accessed 27 February 2022].

Anon., 2021. *Metahuman Creator.* [Online]
Available at: https://www.unrealengine.com/en-US/metahuman-creator
[Accessed 27 February 2022].

Brooks, S. K. et al., 2019. *BMC Psychology.* [Online]
Available at: https://bmcpsychology.biomedcentral.com/articles/10.1186/s40359-019-0360-6
[Accessed 26 February 2022].

Centre for Excellence in Universal Design, n.d. [Online]
Available at: https://universaldesign.ie/what-is-universal-design/
[Accessed 07 02 2022].

Datatilsynet, 2018. *Datatilsynet.* [Online]
Available at: https://www.datatilsynet.no/personvern-pa-ulike-omrader/internett-og-apper/cookies/
[Accessed 3 5 2022].

Edward Winter, n.d. *Humanity & Inclusion.* [Online]
Available at: https://www.hi-us.org/advocating_for_disabled_refugees
[Accessed 4 May 2022].

European Union, 2022. *European Commission.* [Online]
Available at: https://ec.europa.eu/commission/presscorner/detail/en/ip_22_1978
[Accessed 14 04 2022].

Ghezala, S., 2021. *Using Next.js and Vercel to instantly load a data-heavy website.* [Online]
Available at: https://tinloof.com/blog/using-next.js-and-vercel-to-instantly-load-a-data-heavy-website
[Accessed 18 May 2022].

Gytri, A., Askeland, A. S. & Brakstad, T., 2018. *NRK.* [Online]
Available at: https://www.nrk.no/vestland/sas-kan-fa-dagboter-pa-150.000-kroner-1.14168232
[Accessed 05 02 2022].

Information Commissioner's Office, 2021. *Information Commissioner's Office.* [Online]
Available at: https://ico.org.uk/about-the-ico/news-and-events/news-and-blogs/2021/09/ico-to-call-on-g7-countries-to-tackle-cookie-pop-ups-challenge/
[Accessed 3 5 2022].

Isaac, M., 2022. *New York Times.* [Online]
Available at: https://www.nytimes.com/2022/02/03/technology/facebook-meta-challenges.html
[Accessed 02 05 2022].

Jennifer Bryant, 2022. *International Association of Privacy Professionals.* [Online]
Available at: https://iapp.org/news/a/far-reaching-implications-anticipated-with-austrian-dpas-google-analytics-decision/
[Accessed 03 05 2022].

Jong, J. T. V. M. d., 1995. World Health Organization. *Preventing stress among refugees,* 6(November-December), pp. 12-14.

NTB, 2018. *Aftenposten.* [Online]
Available at: https://www.aftenposten.no/okonomi/i/rLjK6A/sas-kan-faa-tvangsmulkt-for-daarlig-tilrettelagt-nettside
[Accessed 06 02 2022].

Scrum.org, 2022. *What Is Scrum?.* [Online]
Available at: https://www.scrum.org/resources/what-is-scrum
[Accessed 21 May 2022].

Silvia Rasheva, 2022. *Unity.* [Online]
Available at: https://blog.unity.com/news/introducing-enemies-the-latest-evolution-in-high-fidelity-digital-humans-from-unity
[Accessed 18 05 2022].

State of JS, 2022. *State of JS.* [Online]
Available at: https://2021.stateofjs.com/en-US/
[Accessed 16 02 2022].

Temkin, D., 2021. *Google.* [Online]
Available at: https://blog.google/products/ads-commerce/a-more-privacy-first-web/
[Accessed 15 01 2022].

Unity Technologies, 2022. *Unity Documentation.* [Online]
Available at: https://docs.unity3d.com/Manual/webgl-performance.html
[Accessed 23 05 2022].

Unity Technologies, 2022. *WebGL browser compatibility.* [Online]
Available at: https://docs.unity3d.com/Manual/webgl-browsercompatibility.html
[Accessed 23 05 2022].

Uutilsynet, n.d. *Uutilsynet.* [Online]
Available at: https://www.uutilsynet.no/english/information-english/252
[Accessed 05 02 2022].

Vercel, 2022. *Client-side data fetching.* [Online]
Available at: https://nextjs.org/docs/basic-features/data-fetching/client-side
[Accessed 05 May 2022].

Vercel, 2022. *Static File Serving.* [Online]
Available at: https://nextjs.org/docs/basic-features/static-file-serving
[Accessed 26 April 2022].