



**UNIVERSITY  
OF LONDON**

**CM2020 Agile Software Projects  
Final Project Report  
Team Number: 91**

**Team Members:**

**Tan Wei Xiang (200553076),  
Darren Ang Jun Rong (200553559),  
Prasanna Palaniappan (200644857),  
Mohan Krishna (210220108),  
Hao Fan (200628868),  
Ye Myat Oo (220253387).**

**Date: 05/09/2022**

**Project Title: Hospital Management App  
Word Count: 9921**

## **Abstract**

Previous market research has shown that the proportion of the elderly population in Singapore is increasing year on year, and with it, the demand for healthcare services such as online healthcare. Many seniors are not adept at using technology such as online appointments to access healthcare services or are unable to visit hospitals in person due to their health conditions. However, survey data shows that demand for online healthcare services has nearly doubled since the outbreak of the Coronavirus pandemic, with the use of telemedicine services surging from 17% to 32% in 2020. Hence, our team has decided to alleviate this issue and aid the elderly.

During the course of the project, our team investigated some of the existing telehealth applications in the market as depicted in the first report and found that few of them provide convenient services for the elderly population. Thus, we have decided to develop a more user-friendly and easy-to-read web application to meet the needs of the elderly population for online health care services.

# Table of Contents

<b>Background</b>	<b>4</b>
Introduction	4
Aim & Objectives	5
Scope	7
Limitations	9
<b>Planning and Research</b>	<b>11</b>
Project Management	13
<b>Prototyping and Iteration</b>	<b>21</b>
Background	21
First Prototype	21
Second Prototype	28
Third and Final Prototype	42
<b>Design</b>	<b>55</b>
<b>System Development</b>	<b>60</b>
Conceptualisation of Application	60
Git and the GitHub repository	61
Software Structure and Components	63
Black Box Testing	67
Code Rundown	72
<b>Analysis</b>	<b>83</b>
<b>Evaluation &amp; Potential Project Scope</b>	<b>85</b>
<b>Conclusion/Summary</b>	<b>87</b>
<b>References</b>	<b>88</b>

# **Background**

---

## **Introduction**

The goal for this phase of the project is to develop the hospital management application for the elderly's use and ensure it runs the required functions our team have planned to implement successfully. The team is divided into three front-end and three back-end developers. The back-end team is responsible for building the database, refining the data content as well as connecting the database to the front-end. The front-end team on the other hand focuses on the presentation of the required information in the web pages and their functionality. A lead scrum master was also assigned to ensure that each member was on time with their tasks and that the team was on track with the project development. Throughout the course of the project, the team consistently made references to the content of our original report and followed closely our proposed prototypes and ideations.

This report will contain the process of how the team carried out the project from start to finish. It includes how we planned for the project in the beginning, the development of our code and application over time and different iterations, user testing review and feedback of our prototypes, and an overview of the technologies used in assisting our development and elaboration on the code and how they achieve their corresponding functions. The report concludes with an analysis of the entire project's journey as well as an overall evaluation of what the team has achieved during this time and possible future improvements if we were to continue working on this project.

The current version of the application allows users to register for an account or log in with an account already created before and stored in our database. When registering for a new account, the user has to fill in a form of their personal details and medical information which will be uploaded and stored in the database. Once logged in, users will be able to carry out various functions such as scheduling appointments online, managing them by rescheduling or deleting them if needed, calling for emergency medical assistance, and viewing previous past medical records in the application.

# Aim & Objectives

## Aims

As mentioned in the beginning, the main aim for this phase of the project is to successfully develop the hospital management application as planned previously. To achieve our aim, the team has discussed and chosen a few basic and essential functions that are to be implemented for our users in the application.

The aim of our application is to provide

- An online appointment booking system where the elderly can make a medical appointment to see a doctor when they fall ill so that they can just head down to the hospital or clinic when it's their turn to prevent unnecessary waiting which might be tiring for them.
- A medical records page which would allow the elderly to look through past medical records and track previous appointments
- An emergency SOS page to assist those elderly who live independently in times of emergency to seek immediate help when required
- A simple to look, easy to understand and use application that even the elderly would have no difficulty navigating and using
- A remember-me system so that once the user is logged in for the first time, logging in would not be necessary again in the future. This is to assist the elderly who might have poor memory and have a hard time remembering their login credentials.

## **Objectives**

To achieve the aims we have planned for our project and application, the team has agreed to set the following objectives to work on

- Develop a user interface for users to register an account and log in
- Creating a functionality whereby if once a user is logged in, it will be remembered and the user will be logged in automatically in the future
- Develop a home page for user to navigate over the various functions the application provides
- Develop an appointment booking system where the user is able to create multiple appointments with a time and date which will be recorded
- Develop an appointment management system where the user can amend the appointment date and time of appointments they booked or delete them
- Ensure the proper working functionality of the booking system where bookings of the same time and date cannot be repeated.
- Creating a database for the storage of user information as well as appointment information
- The proper linkage of the database to the web pages to send and retrieve any data required correctly
- Develop a simple Graphic User Interface (GUI) system that is not complex and easy to understand for our application.

# **Scope**

## **Project scope description**

Our Hospital Management Application is a low-barrier, simple, easy-to-understand application with the essential functions of making medical appointments online and retrieving medical information. As booking medical appointments offline require physical effort and time, our application with the online appointment system will help solve this problem by making it more convenient for the elderly to do so and protect their health and safety. The application can be further improved and enhanced in the future.

## **Acceptance Criteria**

1. The application should be easy to understand and navigate for the elderly who are not tech-savvy
2. The application's design should be visually suitable for the elderly eyes, being simple and clear
3. Low requirements from the user once the user has registered and provided their personal and medical details
4. The application should provide an accurate display of available and non-available appointment sessions for booking in real-time

## **Deliverables**

1. One fully working website of our application
2. One source code of the application
3. One schema of the database
4. One project report detailing the project process and the application details
5. One video showcase of our application

## **Project Exclusions (from what was proposed in the first report)**

1. Doctor's part of the application
2. Changing font size function
3. Language changing function

## **Constraints**

1. Time as every team member still had their individual commitments aside from this project such as other school modules or work
2. Resources as the team had to self-source and learn how to implement various parts of the application
3. Covid-19 caused one of our teammates to be stuck overseas, resulting in a change in work style as everything had to be held online

## **Assumptions**

1. All of our application users either understand or are English-educated
2. All of our application users have a simple knowledge of using technology such as a phone or a computer to access and navigate a website
3. All of our application users have easy and convenient access to a technology device such as a phone or computer to use our application

# **Limitations**

As a team, we encountered various limitations and challenges during the project's execution.

## **Teamwork limitations**

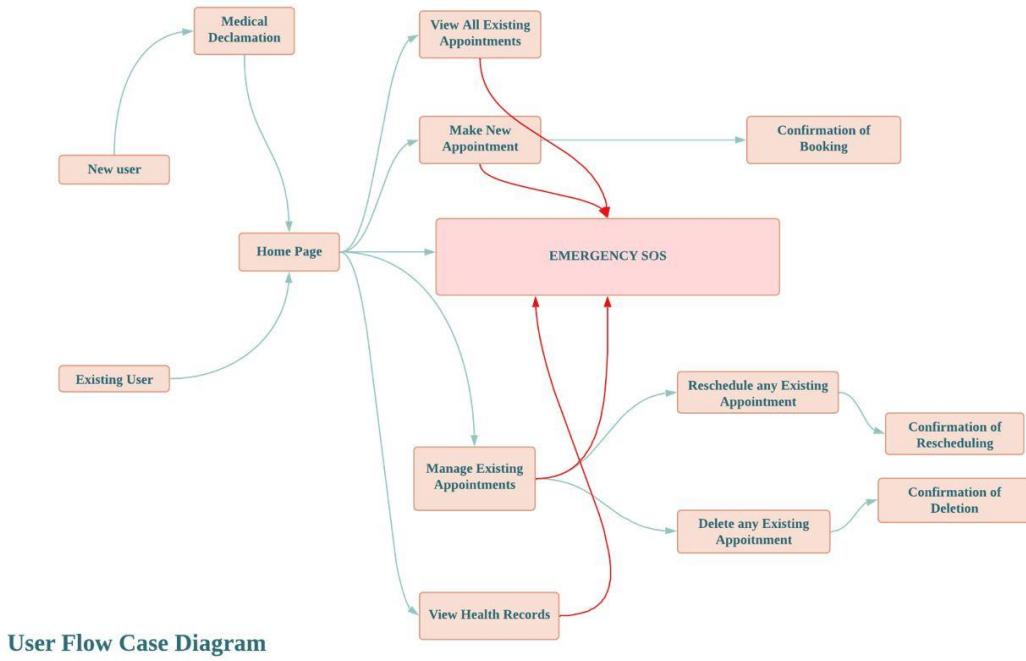
As it was the first time for all of us to be working on a coding project together, there were many issues we had encountered. Most of us were used to working individually on coding projects or programs which meant that all of the work regardless of front-end or back-end is done on one system which is easy to compile. However, in this project, we were tasked to take up different parts of the application which would mean that different parts of the application were done on different systems and in different coding styles based on the individual. We had to learn how to coordinate our coding styles and learn how to compile different parts of the program separately and ensure the code runs smoothly which we encountered many errors but overcame in the end.

## **Knowledge limitations**

Also, as this was a group project and each team member came from a different background, this meant that each member had a different level of programming skills and different knowledge of various programming languages. This created difficulties in terms of deciding what language and dependencies to work on the project as it required some of us to compromise to pick up and learn a new language or dependency that we agreed on using which was not an easy or short task.

## **Technical Limitations**

Since working on different components of the application were assigned to different team members, this would result in a time delay or a time lag during the project as there were times we had to wait for one to be done with their part and uploaded before being able to continue as our task was reliant on theirs. Our team decided to minimise this limitation by planning out a user flow case diagram ahead of starting our project so we had a good knowledge of what task was dependent on which task. This allowed for a better allocation of the flow of tasks to be done and avoid delay and time lags as much as possible. The diagram can be seen in the image below.



*Figure 1. User Flow Case Diagram*

## Communication and time limitations

As mentioned above, the Covid-19 pandemic also provided us with some constraints on carrying out the project. Our team is comprised of individuals who come from various countries, although most of us reside in Singapore, one of our team members resided in China. The Covid-19 situation in China prevented him from travelling to Singapore as there was a lockdown which resulted in us adapting most of our project process to be carried out online to ensure that he could be updated. He also had to use a virtual private network (VPN) to access communication applications we were using such as Telegram for time-to-time updates which meant a delay in information relayed to him. This resulted in adapting strategies such as daily scrum meetings to be online instead of offline and major sprint reviews and team meetings to be online as well. Coupled with the fact that every individual had their personal commitments outside of the project as well, it was hard to coordinate and find a time to come together for discussions and meetings but as every member was willing to compromise and make sacrifices, we were able to do so.

# Planning and Research

---

Mildly reflecting on the first report of our market research, it is evident that the telemedicine industry is growing, ever more so with the debilitating effects of the pandemic on the crowding situation in hospitals. We knew from that that this project could potentially alleviate the stress weighing down on our local hospital staff, therefore we decided to continue with this project. Carrying on further research, this time, we delved deeper into matters that have a greater impact on our product such as the styling choices and features that are more relevant to the local community. An article titled "Color in Designing Technology for Seniors" [1] talks about the effects of ageing on vision and how elderly are less capable of discerning between colours of lower contrasts. It was also mentioned that navy blue, sky blue or aquamarine are the colours of choice by seniors. These research findings will be proven substantial in the later sections where the prototypes are covered in further detail. In addition to the styling choice, one particular feature that was not seen in any other similar telemedicine applications in the market was the emergency SOS feature. A statistics report on the number of one-person households aged 65 and older in Singapore was done recently [2], and at a glance, it was obvious that the number of elderly living alone was increasing over the years. The team believes that having this emergency appointment could help redirect some of the crowd at the emergency department to the appropriate departments.

Our ultimate goal is to develop a hybrid hospital administration in which all hospital information and services are consolidated and unified into a single online browser that patients can access using a range of multi-platform devices.

The most thorough overview of their appointments, access to their medical information, and the ability to schedule urgent SOS visits with doctors are all requirements for our target customers, the pioneers.

The main project **targeted** to address the following issues mainly:

- 1) How do we create advanced appointment management booking systems? was one of our key objectives.
- 2) How can we deal with and lessen obstacles like age, languages, gender, geographic accessibility, etc.? How can we create a thorough user interface (UI) system that allows even elderly users to use the app quickly and efficiently?
- 3) How do we look through and keep track of doctor's appointments and reports?

Upon reflection, our group is confident that we successfully addressed the identical aim points listed above:

- 1) We think that we have greatly improved accessibility for scheduling, managing, and deleting appointments. Users may easily navigate through the features because they are all within reach of one another.
- 2) Our earnest planning and front-end designers have successfully overcome challenges associated with the target users, such as age, accessibility, etc., as described in detail under the design part.
- 3) Due to time restrictions, our team was unable to code out the doctor's end of the application. However, its architecture would have been comparable to the following: With their login information, a doctor from the designated organisation can access the system and view all of their scheduled appointments for the day. Additionally, they would have access to and be allowed to change patient medical records (micro-segmented privileges only). Additionally, they would be able to update the database and prescribe medications to patients. Doctors would also be able to provide patients with medical certificates for the right reasons and dispose of medical reports at the patients' request.

# Project Management

Software development is a lengthy and complicated process, and it is easy to lose track of time or get lost in the process without proper planning or management. To facilitate this project, the team used the following tools and strategies.

## Gantt Chart

Having a Gantt chart gave the group a better sense of the progress expected in order to complete the software project. The following Gantt chart below was gradually filled along the team's journey for development:

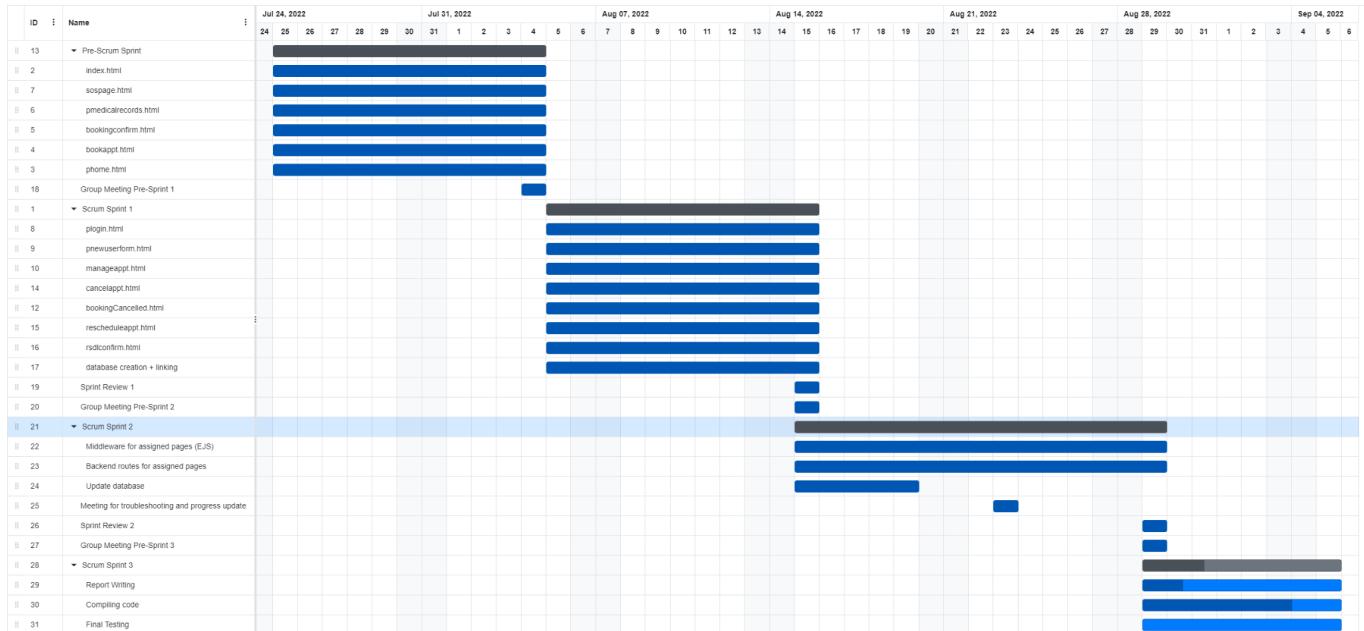


Figure 2. Gantt chart of progress to date

The development of the application began around mid-July when a few of our members began working on the app. However, due to unmatching schedules and members falling ill, we were unable to proceed with the first sprint at that time. Therefore, the members that were available went ahead with familiarising themselves with the technologies that we have decided on using to aid in the development of the web application. This took slightly longer than a week before we began setting up the environment for development and working on the application itself.

## **Team Composition**

Wei Xiang - Scrum Master/Developer  
Prasanna - Developer  
Ye Myat - Developer  
Mohan - Developer  
Oliver - Developer  
Darren - Database Manager/Developer

## **Agile Methodology**

The agile framework that our team has decided to adopt is scrum. Firstly, due to the limited amount of time left for submission from the start of development, the group agreed that following scrum's methodology would spur us to develop the codebase more rapidly with the regular deadline to meet which was either the story points (number of hours estimated to complete a task) or the sprint review at the end of every sprint. Following scrum also helps each group member be more accountable for their contributions by having short daily scrum meetings to update one another on progress.

## **Sprint Planning**

Around the start of August, the team gathered for the official start of our consecutive sprints by having a sprint planning for sprint one following agile principles and procedures, we then discussed the user stories gathered during the midterm software project proposal phase and then broke them down into product backlogs. Some of these product backlogs were then chosen to be that sprint's backlog based on how crucial the functionality is to the product as well as its order of priority on the stakeholder's list. The following diagram below are examples of our scrum planning board during different sprint plannings:

TO DO	IN PROGRESS 10 ISSUES	DONE 6 ISSUES
	<p>plogin.html (basic html functionality) HAA-7 MK</p> <p>pnewuserform.html (basic html functionality) HAA-25 HF</p> <p>manageappt.html (basic html functionality) HAA-12 YO</p> <p>cancelappt.html (basic html functionality) HAA-13 YO</p> <p>bookingCancelled.html HAA-26 YO</p> <p>rescheduleappt.html (basic html functionality) HAA-14 WT</p> <p>rsdlconfirm.html (basic html functionality) HAA-16 0 WT</p> <p>database creation HAA-19 DA</p> <p>database linking HAA-20 DA</p> <p>database schema creation HAA-27</p>	<p>index.html (basic html functionality) HAA-6 ✓ MK</p> <p>phome.html(existing) (basic html functionality) HAA-8 ✓ MK</p> <p>bookappt.html (basic html functionality) HAA-10 ✓ WT</p> <p>bookingconfirm.html (basic html functionality) HAA-11 ✓ WT</p> <p>pmmedicalrecords.html (basic html functionality) HAA-17 ✓</p> <p>sospage.html (basic html functionality) HAA-18 ✓ HF</p>
		+

Figure 3. Sprint Board for Sprint 1

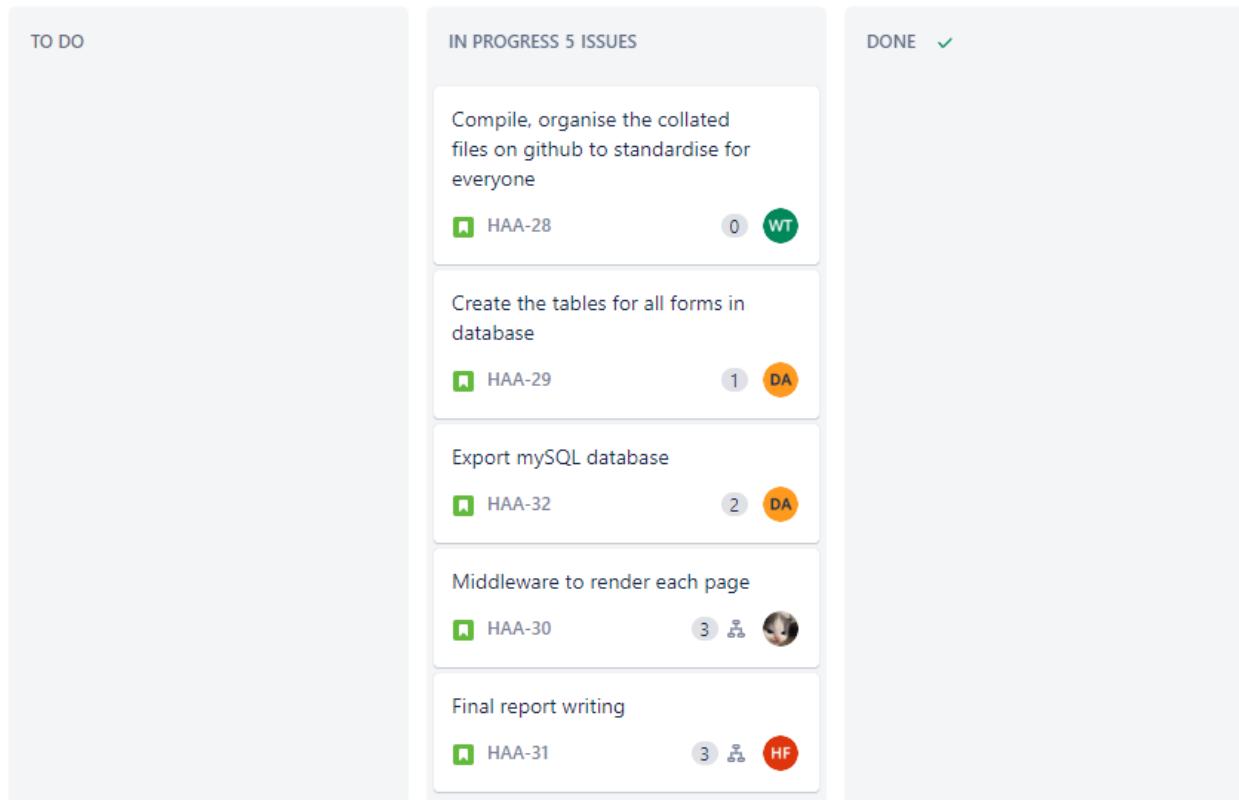


Figure 4. Sprint Board for Sprint 2

With the board ready and the team members aware of their tasks, we then proceeded with the sprint.

## Daily Scrum Meetings

Part of the scrum framework requires that we have a daily scrum meeting to update the team on individual progress using the 3 questions: 1. What have you done today? 2. What are you going to do the next day? 3. Are there any issues faced and troubleshooting assistance needed? In light of the pandemic situation and the varying schedules that the team members had, the scrum master decided to hold one on the chatting platform 'Telegram'. We have also decided to exclude weekends and holidays. The following images below detail some of the scrum meetings that we had:

**Pinned message**

1. What have you done yesterday/before the daily meeting? 2. Any issues you faced with the tasks?

YE Myat joined the group via invite link

1. What have you done yesterday/before the daily meeting?  
 2. Any issues you faced while working on the tasks?  
 3. What will you be doing from now till the next scrum meeting?

Wei Xiang Terence pinned "1. What have you..."

Oliver Fan joined the group via invite link

Mohan Krishna  
Sprint 1 time duration: 4th - 14th August 9:26 PM

August 5

Mohan Krishna  
Mohan -  
 1. I have done the basic html code for the login page. It includes the username, password input fields.  
 2. I made the design simple and plan to add styling and complexity in the next stages.  
 3. I will work on form validation for the login page which will involve javascript code. This is to ensure the user has entered valid inputs.

YE Myat  
Still working on bookingcancelled.html page 8:43 PM

1. I made the html pages for rsdiconfirm and rescheduleappt, also made the elements on the pages neater.  
 2. Had some minor git commit message mishap but remedied it.  
 3. I will be looking through the rubrics and prepare to document our progress. Also will be on standby in case any members need help with their tasks

Prasanna (SIM)  
Prasanna -  
 1) Today, I had to do some research as to how i can implement the google calendar into the website rather than a hardcoded one. I think I have a rough idea now having found a way as to how i want to incorporate it into the existing code for the same page. I also did the basic outline for the appointment management but not done though.  
 2) Uh not any issue cuz i was a bit distracted and got offboard a bit.  
 3) I will finish the google calendar tomorrow and TRY to go further with the other page assigned to me

Darren (SIM)  
 1. Today I did research to learn more on the connection of mysql database to webpage via nodejs and php  
 2. Only issues faced were understanding how it work and the slight complexity of it  
 3. I will begin testing and coding out a basic database to webpage connection and getting it to work before i begin on the actual webpage

Figure 5. Scrum meetings for 5 Aug

**August 6**

**Mohan Krishna**

1. Today I researched form validation methods using javascript. I have learnt how to implement form validation for the login form.  
 2. I faced some difficulty trying to get access to the form input fields. I took help from stackoverflow to understand and clear my doubts.  
 3. I will now implement the form validation logic using javascript for the login page

10:36 PM

**Prasanna (SIM)**

1) Implemented calendar in the website.  
 2) Got very annoying with the bugs but they were very easy to spot after a while. So nothing big.  
 3) Ill begin and try to finish the other page assigned tomorrow. I couldnt start today because the medicalrecords page assigned to me took too much of time. I also have to work with Darren to get the dbs scheme and push to Mohan asap

10:39 PM

**August 8**

**Oliver Fan** Reply

1) done basic html code for SOS page.includes the basic Alerting information, checkbox and redirecting button  
 2) Tried simple design with CSS code  
 3) will apply the checkbox value confirmation function with code in JS file

9:13 PM

**Darren (SIM)**

1. Started on a basic html page and simple database and attempting the connection and linkage of both  
 2. Faced difficulty with the proper connection of both, soemtime it works, sometime it doesnt when i edit things on the webpage  
 3.Will continue working on it and establishing a stable and proper connection before beginning on our actual webpages

11:16 PM

**YE Myat**

Started working on html codes today. Was going through coursera weeks.

11:25 PM

1. Continued working on the report rubrics  
 2. No issue so far.  
 3. Will prepare to look at implementing functionalities for our webpages

11:29 PM

**YE Myat**

1. Today i started on html codes.  
 2. No issues encounter till now.  
 3. Will continue to work on another after this html page.

11:30 PM

**August 9**

**Oliver Fan**  
**OLIVER**

1. Coninue with CSS and JS code, done the checkbox confirmation function  
 2. will finish CSS and Js code by tmr  
 3. will starting on html code for medicalForm page

11:26 PM

Figure 6. Scrum meeting from 6 Aug - 8 Aug

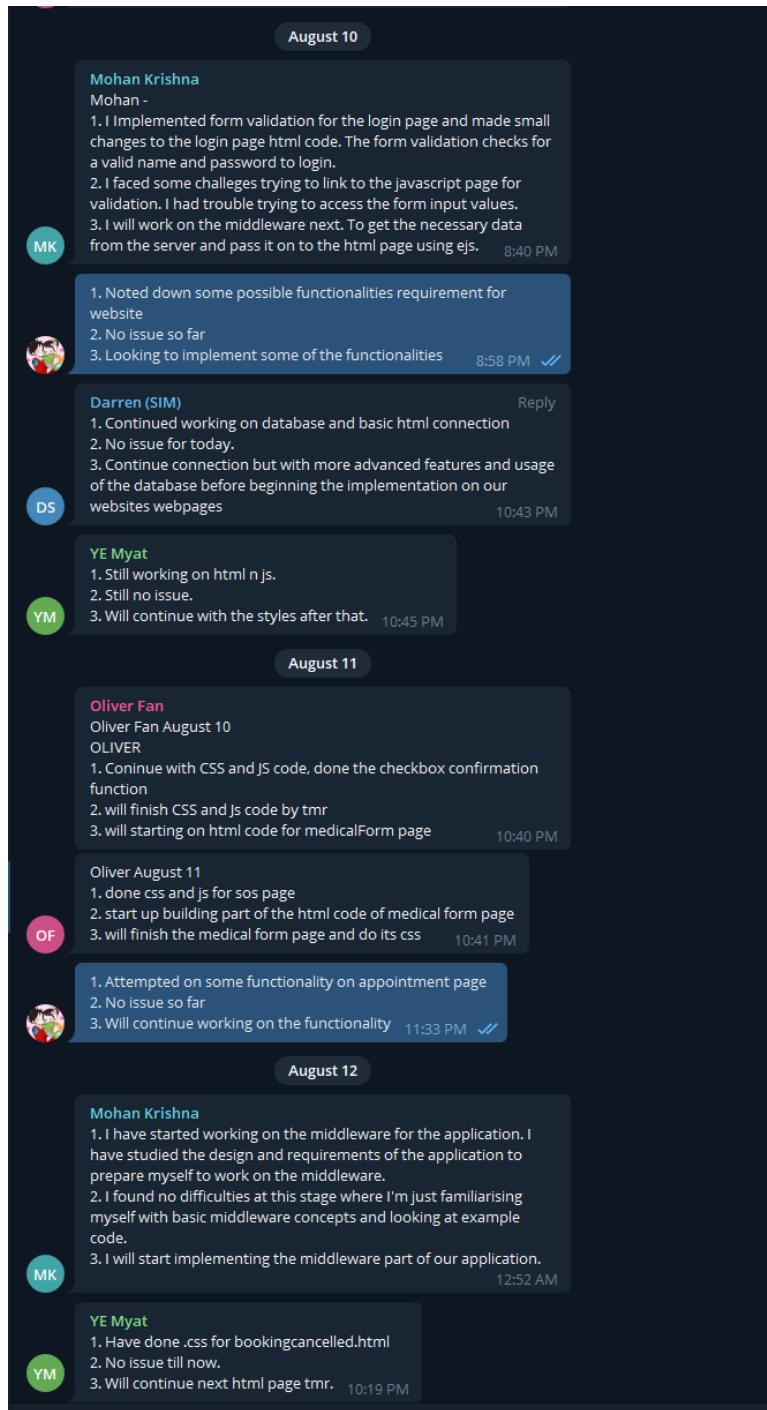


Figure 7. Scrum meeting from 10 Aug - 12 Aug

Having these daily scrum meetings not only allows the scrum master to keep track of development progress but also allows individual developers to flag any issues they may have so that other developers can assist in troubleshooting.

## **Sprint Review**

At the end of every sprint, the team will gather together to have a sprint review where code is reviewed, the sprint backlog is considered for either completion, a shift to the next sprint or to be put on hold for other higher priority backlog that could potentially be inserted midway by the product owner or stakeholders.

# Prototyping and Iteration

---

## Background

While working on the prototyping of our project, we first worked towards one that resembled as close to the medium-fidelity prototypes as shown in the wireframes in the first proposal. After completing that, we improved our prototype further over the course of several iterations based on user testing and input from our intended audience. Below, you can see how we developed via each iteration of our prototypes.

## First Prototype

Firstly, we came up with a prototype with every functionality we required for our application listed in the first proposal. This was without much focus on its design, mostly working on HTML, Javascript and basic CSS. This included the displaying of each page's required information in basic HTML and CSS, the redirecting of pages in the right order, connecting the database to the webpage, storing data from the submission in forms on the webpages and the accurate display of data on the webpages from querying the database. The images below show our first prototype

[E-Health Application](#)

### Your, E-Health Application

Make and edit appointments without hassle, skip queues and enjoy the perks of having all your medical records at your convenience. Afterall, your health is in your hands. Fret not, it's never too late. [Sign up here!](#)

Patient Particulars

User Name   
Password

Doctor Particulars

Doctor's Identity Number   
Password

**Book and Manage Appointments**

Waiting for your doctor when at medical unease can be very frustrating. Book an appointment before hand to avoid the hassle of queuing.



**View Medical Records**

Your health at your very fingertips. Have access to all your medical information with just the press of a button. You now don't have to pay consultation fees just to just get updates on your health.



**Emergency SOS Appointment Featuring**

Need an immediate appointment? Don't be afraid. Our SOS feature helps you get priority access to an expert without having to wait. You're never alone.

---

*Figure 8. First Prototype of Index Page (Main Log-in Page)*

E-Health Application

- [Take me back I do not wish to sign up](#)

## You're only one step away.

Access a full suite of health and wellness services to look after every aspect of your healthcare needs.

### Medical Declamation Form

Please help us by providing some details so our services can be tailored to better suit your medical needs.

<input type="button" value="▼"/>	Gender
dd/mm/yyyy	<input type="checkbox"/> Date of Birth
Your Age	Age
Nationality	Nationality
PH	PH
emergency_PH	Emergency Contact
Your residential address:	
Your Residential Address	
Note that we will use your location to contact your nearest hospital in case of an emergency and other purposes	
<input checked="" type="checkbox"/> Family history of any illness	
<input checked="" type="checkbox"/> Are you on any medications	
<input checked="" type="checkbox"/> Any Medical Allergies	
<input type="button" value="▼"/> Universal vaccinations you have received	
aware	Add any specific notifications to doctor (if you wish to)

[Sign me up!](#)

By submitting, you agree to our [Terms & Conditions](#)

*Figure 9. First Prototype of Sign-up Page (Medical Declaration Page)*

# Welcome, Prasanna.

Access a full suite of health and wellness services to look after every aspect of your healthcare needs.



## **View My Appointment(s)**

Access the status of your scheduled/active appointments.

[Appointments](#) →



## **Book New Appointment**

Schedule yourself a new appointment at your hospital here.

[Book New Appt Now](#) →



## **Manage Appointment(s)**

Reschedule or Cancel your active appointments here.

[Manage Now](#) →



## **Access my personal Medical Record(s)**

Access, read and manage your personal medical records here.

[View My Health Records](#) →



## **Emergency SOS Appointment**

Schedule a priority appointment for medical emergencies here. **In case of severe emergency call an ambulance!**

[Book SOS now](#) →



Sign out of your account here.

Figure 10. First Prototype of Patient Home Page after signing in

## My Appointments

The following is (are) your existing appointments. If you wish to make or manage any existing appointments please proceed to the respective page sections.

Service	Date and Time	Status
Blood Health Checkup	22/09/2022, 22:03	Confirmed
Blood Health Checkup	22/09/2022, 22:03	Confirmed
<a href="#">Add New Booking →</a>		
<a href="#">Reschedule Booking →</a>		
<a href="#">Delete Booking →</a>		

Figure 11. First Prototype of View Appointments Page

## Book a New Appointment

Make a new appointment at your nearest (registered) hospital. Do not worry, we will find the one suited to your needs with your registered details.

29th August 2022

**Proceed to fill the form below.**

Reason for Visit

dd / mm / yyyy  Appointment Date

--- : --- : ---  Appointment Time

Enter your name...  First Name (as registered)

Enter your name...  Last Name

PH number  PH

By submitting, you agree to our [Terms & Conditions](#)

Figure 12. First Prototype of Book Appointments Page

## We've got you confirmed!

We just added an appointment for you. Please find the details below!

Service	Date and Time	Status
Blood Health Checkup	22/09/2022, 22:03	Confirmed
<a href="#">Reschedule Booking →</a>		
<a href="#">Delete Booking →</a>		
<a href="#">Add another Booking →</a>		

Figure 13. First Prototype of Booking Confirmed Page

## Manage Appointments

The following is (are) your existing appointments which you can manage. Please proceed as to which service you require at the moment.

Service	Date and Time	Status
Blood Health Checkup	22/09/2022 22:03	<a href="#">Reschedule</a> → <a href="#">Delete</a> →
Blood Health Checkup	22/09/2022 22:03	<a href="#">Reschedule</a> → <a href="#">Delete</a> →

Figure 14. First Prototype of Manage Appointments Page

## Reschedule Appointment

Reschedule your appointment to better suit your convinence. Do note that we do not recommend rescheduling within an hour's range before your appointment for the in-convinence caused to other patients.

29th August 2022

Reschedule by revising the form below.

Reason for Visit  Reason for Visit  
dd/mm/yyyy  Revised Date  
--- --- --  Revised Time  
Enter your name... First Name (as registered)  
Enter your name... Last Name  
PH number PH  
  
By submitting, you agree to our [Terms & Conditions](#)

Figure 15. First Prototype of Reschedule Appointments Page

## Booking Rescheduled!

We just updated the appointment for you. Please find the details below!

Service	Date and Time	Status
Blood Health Checkup	22/09/2022,22:03	Updated & Confirmed
<a href="#">Add New Booking</a> → <a href="#">Delete Booking</a> →		

Figure 16. First Prototype of Reschedule Confirmation Page

## Cancel Appointment

Cancel your appointment to better suit your convenience. We do not recommend cancelling within an hour's range before your appointment **unless given an emergency** for the inconvenience caused to other requiring slots.

Service	Date and Time	Status	
Blood Health Checkup	22/09/2022, 22:03	Confirmed	<a href="#">Delete →</a>

[Figure 17. First Prototype of Cancel Appointment Page](#)

## Appointment Cancelled!

We just cancelled that appointment for you. Please find the confirmation for the same below!

Service	Date and Time	Status
Blood Health Checkup	22/09/2022, 22:03	CANCELLED
<a href="#">Add New Booking →</a>		
<a href="#">Manage other Bookings →</a>		

[Figure 18. First Prototype of Cancel Confirmation Page](#)

## Past Medical Records

View your past medical history, and request for official data from the hospitals where you may possibly have records at.

### Retrieve Records from Hospitals

Find basic records from hospitals here. Includes (Inpatient Date, Outpatient Date, Unique ID, and Case of Admission)

Select Hospital	Retriece →			
Patient Unique ID:	Case:	Last InPatient Date: dd/mm/yyyy	Last OutPatient Date: dd/mm/yyyy	Medications

### Request for Detailed Medical Report

If you wish to retrieve professional medical reports, including (but not limited to CTs, Blood Tests, Surgery Reports/ Rich Media Statements) proceed with the form below.

Please note this request will take atleast **5 business days** to respond to. The results of your application will be sent to the email address you have registered with us. **For urgent requests**, contact our helpdesk

Enter your name... ▼   Hospital Name ▼   Reason for request
Enter details of what is it: Can you provide us more info?

Submit →  
By submitting, you agree to our [Terms & Conditions](#)

#### Personal Particulars

- Patient Name comes here
- Age
- Gender
- DOB
- Address

#### Long Term Prognosis

Retrive prognosis from H

#### Vaccinations

- Inactivated poliovirus vac
- DTaP Vaccine, 30th Marin
- Tuberculosis Vaccine, 4f
- TT and DTaP (Booster), 1
- Pfizer Covid (2 Complete)
- Pfizer Covid (Booster Dc)

Figure 19. First Prototype of Patient Medical Records Page

## Emergency SOS

Book a quick appointment in cases of emergency.

Please note that this is **NOT AN APPOINTMENTS BOOKING PAGE**.

The following conditions apply for the same:

- Emergency appointments are meant for situations which **cannot wait until the next day**.
- Please take note to not abuse this service so hindrance to actual emergencies do not happen.
- Appointment period will last only for 10 minutes
- No medical certificates or repeat prescription will be given at the appointment.
- The doctor will attend to your emergency problem only.
- **If you are in severe pain or the situation worsens, please contact an ambulance @995 immediately**

By submitting, I have read and agreed to both, the above conditions and the pre-existing [Terms & Conditions](#).

Successfully requested an Emergency appointment!



We have set your request as top priority.

Please take note that our staff will contact you (registered phone) with further details and proceedings within the next hour.  
**If you are in severe pain or the situation worsens, please contact an ambulance IMMEDIATELY**

Figure 20. First Prototype of Emergency SOS Page

## **Second Prototype**

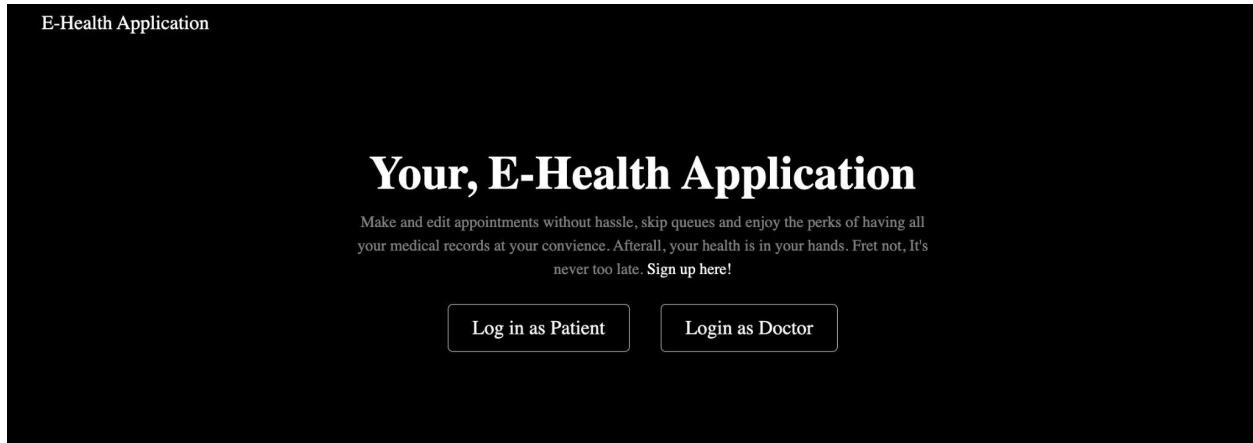
Next, once we tested the functionality of our application from our first prototype and ensured that everything was in place, we shifted our focus to the styling and designing of our prototype. This included the implementation of advanced CSS techniques as well as the use of Bootstrap which we decided to incorporate to assist with our design.

Also, between the duration whereby the first prototype was completed and the beginning of working on the second prototype, we conducted our first round of user testing and feedback via an interview. An interview was chosen over a survey since our target audiences were the elderly and an interview would be better to conduct face-to-face to explain to them our project rather than having them view our prototype and do survey feedback on it online by themselves. This was carried out by having each group member approach the elderly (65 years and above) in their family and showcasing the images of our first prototype and explaining to them the process of our application to obtain their feedback. Some questions that were also crafted by the team were asked to the elderly after we reviewed our first prototype and felt that there were some changes that could have been implemented to improve the application.

The user testing feedback and images of our second prototype can be seen below.

Questions	Possible Response	Response	Feedback and Observation
<b>For the index page as seen in Figure 8, are you confused by the login system if it was not explained to you how to log in?</b>	Yes	12	Most of the elderly were confused and thought they had to fill in the boxes for both patient and doctor's log-in or thought that they could log in via either of the two
	No	6	
<b>For the form on the sign-up page as seen in Figure 9, is every information required displayed? Did we miss out on anything and which should be added?</b>	<b>Yes, everything is displayed and nothing is missed out.</b>	18	The elderly agreed and understood every information that was needed to sign up for the application and felt that nothing else was required to be added
	No, something is missing	0	
<b>For the past medical records page as seen in Figure 19, is the process to retrieve your medical records confusing?</b>	Yes	14	The elderly found the process to retrieve their records from hospitals to be confusing. They stated that they would be unable to remember or know their patient ID and case number and it would be difficult for them as well to remember their inpatient and outpatient dates.
	No	4	
<b>Do you understand the meaning of the abbreviations used in the application such as the one (appt) seen in Figure 10</b>	Yes	8	We observed that although almost half understood the abbreviations, a majority did not which means to us that it would be better not to use any in the application to make it elderly-friendly.
	No	10	
<b>After explaining to you how our application would work, are you confident to use it on your own and is it simple enough for you?</b>	Yes	16	We received good feedback that our application would be easy to use and simple to use once a run-through and explanation of it were carried out but they still needed to judge after they know how the application would look like once the design was implemented
	No	2	

*Table 1. Results and feedback from the questions asked during the first interview with the elderly*



**Book and Manage Appointments**  
Waiting for your doctor when at medical unease can be very frustrating. Book an appointment before hand to avoid the hassle of queuing.

**View Medical Records**  
Your health at your very fingertips. Have access to all your medical information with just the press of a button. You now don't have to pay consultation fees just to just get updates on your health.

**Emergency SOS Appointment  
Featuring**  
Need an immediate appointment? Don't be afraid. Our SOS feature helps you get priority access to an expert without having to wait. You're never alone.

*Figure 21. Second Prototype of Index Page (Main Log-in Page)*

## Changes made

Following the user feedback response we received from the elderly on the logging-in process on the home page being confusing, we have decided to remove the log-in fields from being displayed on the home page. The user will now have to click on one of the buttons depending on whether they are logging in as a patient or a doctor. This will then lead to a pop-up as seen in Figure 23 to prompt the user to key in their username and password in the required fields to log in. The pop-up will also display either "Patient Particulars" or "Doctor Particulars" at the top to allow the user to know which button they clicked on and if they are logging in via the right user.

# Get your health in check

Join and Sign Up Here!

First Name



As in NRIC (or Identity Card)

Last Name



Email address



Password

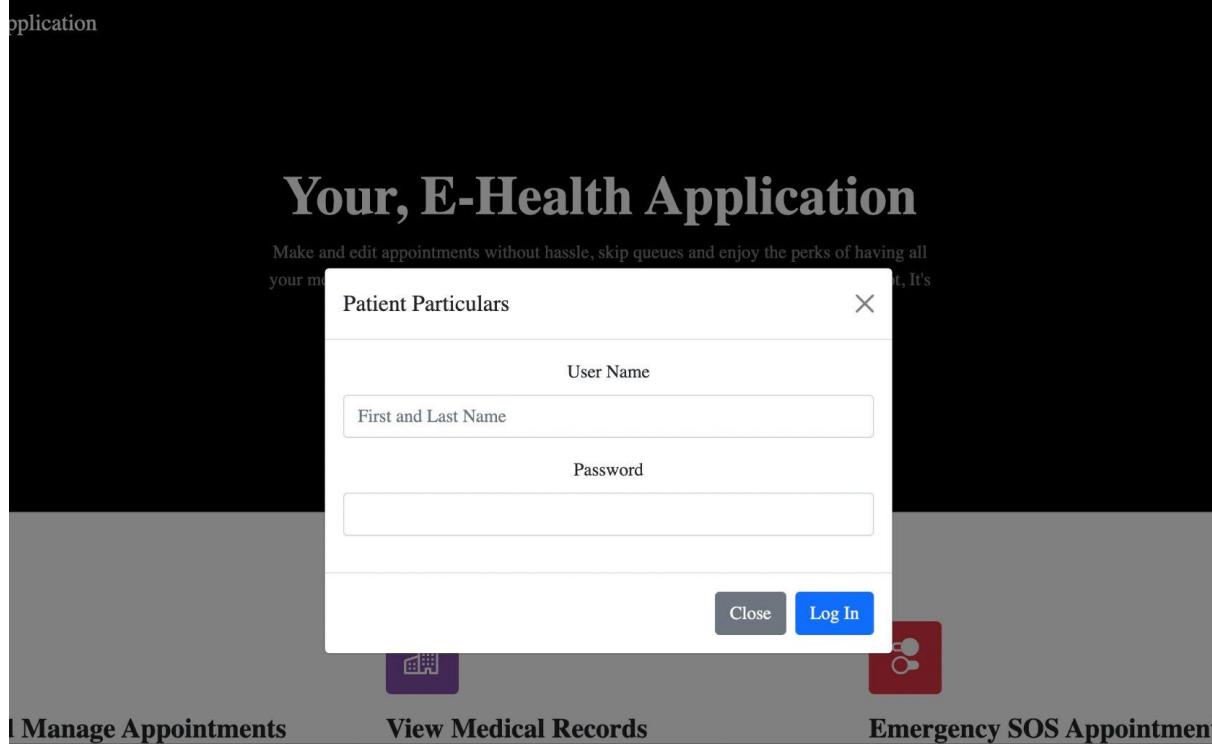


Password must be 8-20 characters long, contain letters and numbers, and must not contain spaces, special characters/ emojis.

Sign me up!

By submitting, you agree to our [Terms & Conditions](#)

Figure 22. Second Prototype of Index Page (Main Log-in Page)(continued)



*Figure 23. Second Prototype of Login modal popup of Index Page*

## You're only one step away.

Access a full suite of health and wellness services to look after every aspect of your healthcare needs.

### Medical Declaration Form

Please help us by providing some details so our services can be tailored to better suit your medical needs.

Gender	
Date of Birth dd/mm/yyyy	
Age	
Nationality	
Phone number	
Emergency Contact	
Your Residential Address	
Note that we will use your location to contact your nearest hospital in case of an emergency and other purposes	
Family history of any illness	
Are you on any medications	
Any Medical Allergies	
Universal vaccinations you have received	
Add any specific notifications to doctor (if you wish to)	
<b>Sign me up!</b>	

By submitting, you agree to our [Terms & Conditions](#)

Figure 24. Second Prototype of Sign-up Page (Medical Declaration Page)

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

## Welcome, Markus Barson.

Access a full suite of health and wellness services to look after every aspect of your healthcare needs.

The screenshot shows the patient home page with a dark header bar. Below it, a large central section displays a welcome message and a brief description of available services. At the bottom, there are six cards arranged in two rows of three, each representing a different service or feature with an icon and a title.

Service	Date and Time	Status
Unwell, Communicable	Sat Sep 10 2022 03:30:00 GMT+0800 (Singapore Standard Time)	Confirmed

**View My Appointment(s)**  
Access the status of your scheduled/active appointments.  
[Appointments →](#)

**Book New Appointment**  
Schedule yourself a new appointment at your hospital here.  
[Book New Appointment Now →](#)

**Manage Appointment(s)**  
Reschedule or Cancel your active appointments here.  
[Manage Appointments Now →](#)

**Access my personal Medical Record(s)**  
Access, read and manage your personal medical records here.  
[View My Health Records →](#)

**Emergency SOS Appointment**  
Schedule a priority appointment for medical emergencies here. **In case of severe emergency call an ambulance!**  
[Book SOS now →](#)

**Log Out**  
Sign out of your account here.

Figure 25. Second Prototype of Patient Home Page after signing in

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

## My Appointments

The following is (are) your existing appointments. If you wish to make or manage any existing appointments please proceed to the respective page sections.

The screenshot shows the 'My Appointments' page with a dark header bar. The main content area features a heading and a descriptive text about existing appointments. Below this, a table lists the patient's existing appointment details. At the bottom, there are two buttons for managing appointments.

Service	Date and Time	Status
Unwell, Communicable	Sat Sep 10 2022 03:30:00 GMT+0800 (Singapore Standard Time)	Confirmed

**Add New Booking →**  
**Manage Appointments →**

Figure 26. Second Prototype of View Appointments Page

## Changes made

From the first prototype of this page as seen in Figure 11, we decided to remove the reschedule and delete booking functionality on this page. It is true that it would be more efficient and time-savvy to include the functions here as the user would not need to go back to the home page to access these functions of the application. However, the team agreed that it should be removed to avoid unnecessary confusion for the elderly and the result of accidental rescheduling or removal of appointments made by them. Thus, although the implementation would be more efficient, as our target audience is elderly, this would ensure maximum effectiveness and user satisfaction by avoiding potential problems.

The screenshot shows a dark-themed web page for an E-Health Application. At the top, there is a navigation bar with links for Home, Appointments, Health Records, and Book SOS. The main title "Book a New Appointment" is centered in large, bold, white font. Below the title, a subtitle reads: "Make a new appointment at your nearest (registered) hospital. Do not worry, we will find the one suited to your needs with your registered details." To the left, there is a calendar for September 2022, showing the days from 1 to 30. The number "4" is highlighted in a teal box, indicating it is the current date. On the right side, there are several input fields for booking: "Reason for Visit" (dropdown), "Appointment Date" (text input with placeholder "dd-mm-yyyy"), "Appointment Time" (text input with placeholder "hh:mm"), "First Name (as registered)" (text input with placeholder "Markus"), "Last Name" (text input with placeholder "Barson"), and "Phone number" (text input). A large blue button at the bottom right says "Book my Appointment". At the very bottom, a small note states: "By submitting, you agree to our [Terms & Conditions](#)".

Figure 27. Second Prototype of Book Appointments Page

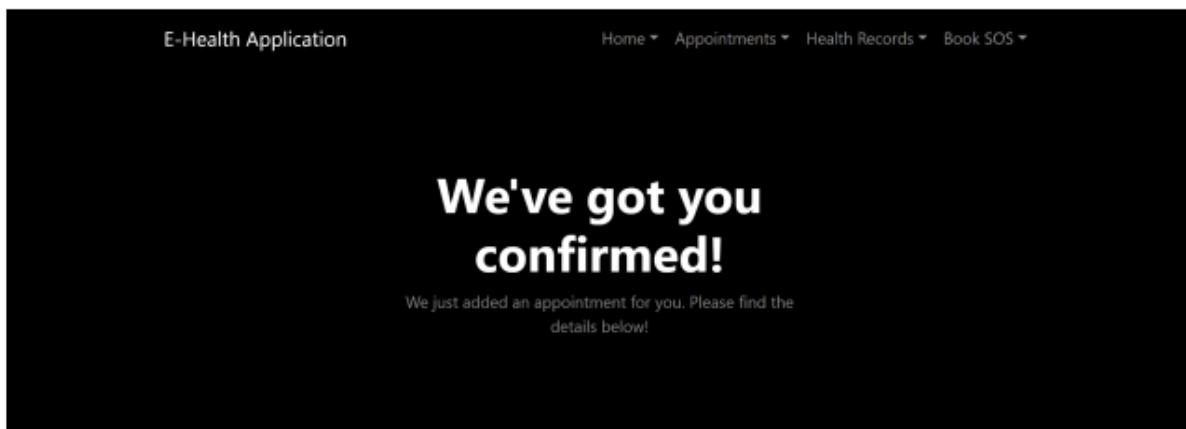


Figure 28. Second Prototype of Booking Confirmed Page

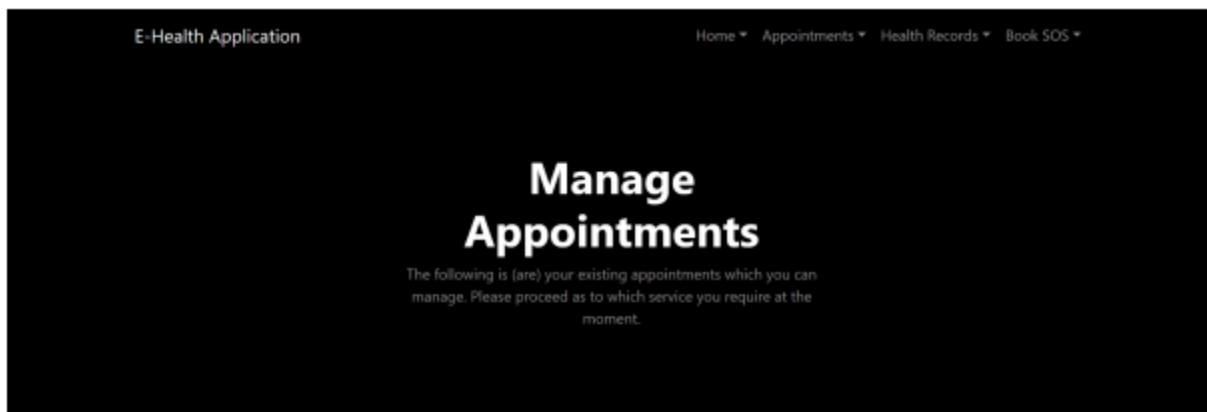
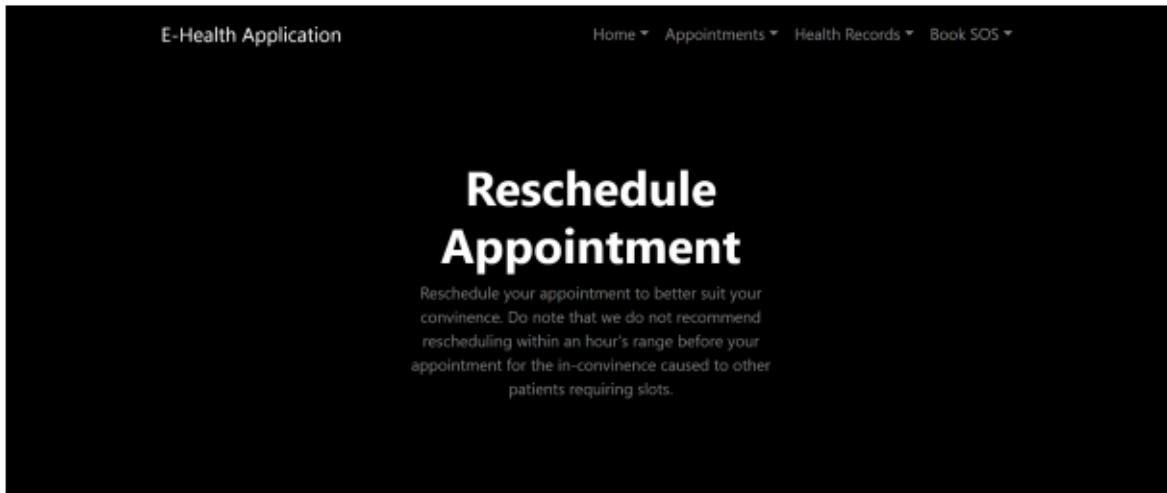


Figure 29. Second Prototype of Manage Appointments Page



29th August 2022

## Reschedule by revising the form below.

M	T	W	T	F	S	S
1	2	3	4	5	6	
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Reason for Visit  
Unwell, Communicable

Revised Date  
dd-mm-yyyy

Revised Time  
--:--

Name (as registered)  
Markus Barson

Phone number  
3249923

**Update my Appointment**

By submitting, you agree to our [Terms & Conditions](#)

Figure 30. Second Prototype of Reschedule Appointments Page

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

# Booking Rescheduled!

We just updated the appointment for you. Please find the details below!

Service	Date and Time	Status
Unwell, Communicable	Sat Sep 10 2022 03:30:00 GMT+0800 (Singapore Standard Time)	Confirmed

[Add New Booking →](#)

[Manage Appointment →](#)

This screenshot shows a confirmation message for a rescheduled appointment. The title 'Booking Rescheduled!' is prominently displayed. Below it, a message states 'We just updated the appointment for you. Please find the details below!'. A table lists the appointment details: service 'Unwell, Communicable', date and time 'Sat Sep 10 2022 03:30:00 GMT+0800 (Singapore Standard Time)', and status 'Confirmed'. At the bottom, there are two buttons: 'Add New Booking →' (blue) and 'Manage Appointment →' (green).

Figure 31. Second Prototype of Reschedule Confirmation Page

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

# Cancel Appointment

Cancel your appointment to better suit your convenience.  
We do not recommend cancelling within an hour's range  
before your appointment unless given an emergency  
for the inconvenience caused to other patients requiring  
slots.

Service	Date and Time	Status
Regular Checkup	Mon Sep 12 2022 17:00:00 GMT+0800 (Singapore Standard Time)	Confirmed

[Delete →](#)

This screenshot shows a page for canceling an appointment. The title 'Cancel Appointment' is at the top. Below it, a note says 'Cancel your appointment to better suit your convenience. We do not recommend cancelling within an hour's range before your appointment unless given an emergency for the inconvenience caused to other patients requiring slots.' A table lists the appointment details: service 'Regular Checkup', date and time 'Mon Sep 12 2022 17:00:00 GMT+0800 (Singapore Standard Time)', and status 'Confirmed'. A red button labeled 'Delete →' is located to the right of the table.

Figure 32. Second Prototype of Cancel Appointment Page

The screenshot shows a dark-themed web application. At the top left is the 'E-Health Application' logo. To its right is a navigation bar with links: 'Home', 'Appointments', 'Health Records', and 'Book SOS'. The main title 'Appointment Cancelled!' is centered in large white font. Below it is a smaller subtitle: 'We just cancelled that appointment for you. Please find the confirmation for the same below!'. A table follows, showing a single row of booking information. The table has three columns: 'Service', 'Date and Time', and 'Status'. The 'Service' column contains 'Regular Checkup'. The 'Date and Time' column contains 'Mon Sep 12 2022 17:00:00 GMT+0800 (Singapore Standard Time)'. The 'Status' column contains 'CANCELLED' in red text. At the bottom of the table are two buttons: a blue 'Add New Booking →' button and a green 'Manage other Bookings →' button.

Service	Date and Time	Status
Regular Checkup	Mon Sep 12 2022 17:00:00 GMT+0800 (Singapore Standard Time)	CANCELLED

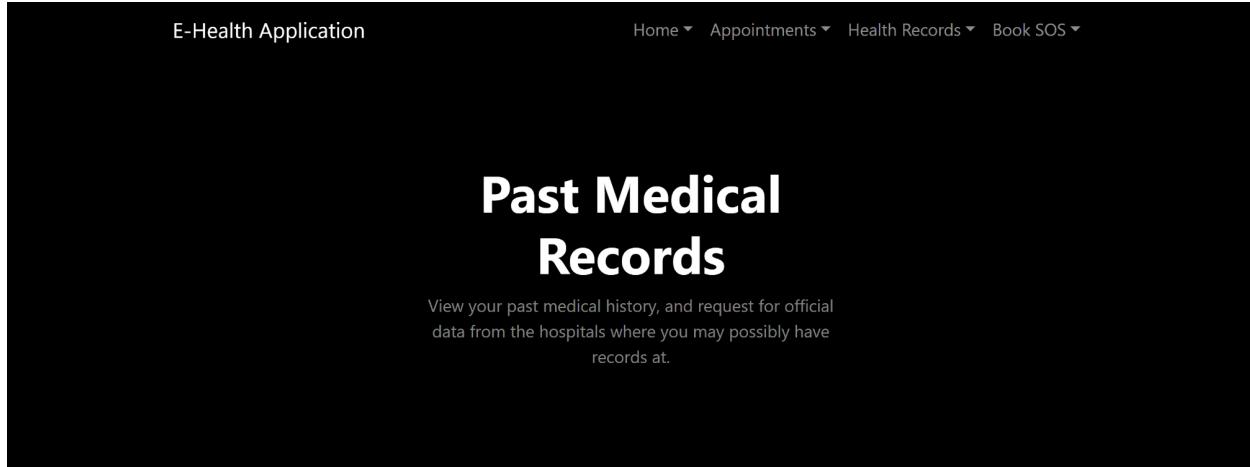
[Add New Booking →](#)

[Manage other Bookings →](#)

Figure 33. Second Prototype of Cancel Confirmation Page

## Changes made

Based on the user feedback response we received on this page on the original prototype having to request the patient's unique ID, case number and more from the user to retrieve records from the hospital was too confusing and hard to remember, we have decided to change it in our second prototype. The team decided to remove the fields and the requests from the user and we have replaced it with a button which upon pressing would display the records of all the records of when the user was admitted or had a medical visit to the hospital. How this is done is that when the retrieve button is pressed, it would lead the user to another page where all the records are displayed. The records are retrieved from the database with the user's particulars. This makes the application more efficient and more user-friendly as they need not remember and fill in the information required from them previously which is hard to remember.



### Retrieve Records from Hospitals

Find basic records from hospitals here. Includes (Inpatient Date, Outpatient Date, Unique ID, and Case of Admission)

[Retrieve →](#)

### Request for Detailed Medical Report

Personal Particulars
Markus Barson
20
Male
Sun Dec 08
Fourth Avenue, 476 Gove Street

Figure 34. Second Prototype of Patient Medical Records Page

If you wish to retrieve professional medical reports, including (but not limited to CTs, Blood Tests, Surgery Reports/ Rich Media Statements) proceed with the form below.

Please note this request will take atleast **5 business days** to respond to. The results of your application will be sent to the email address you have registered with us. **For urgent requests**, contact our helpdesk at +65 66654455

Full Name (First, Last)

Hospital Name

Reason for request

Can you provide us more info?

[Submit →](#)

By submitting, you agree to our [Terms & Conditions](#)

### Long Term Prognosis

Details of Patient's prognosis

### Vaccinations

Inactivated poliovirus vaccine (IPV), 27  
DTap Vaccine, 30th March 2001  
Tuberculosis Vaccine, 4th April 2001  
TT and DTaP (Booster), Vaccine, 3rd Jar  
Pfizer Covid (2 Completed Doses), 3rd  
Pfizer Covid (Booster Dose), 3rd April 2

Figure 35. Second Prototype of Patient Medical Records Page (Continued)

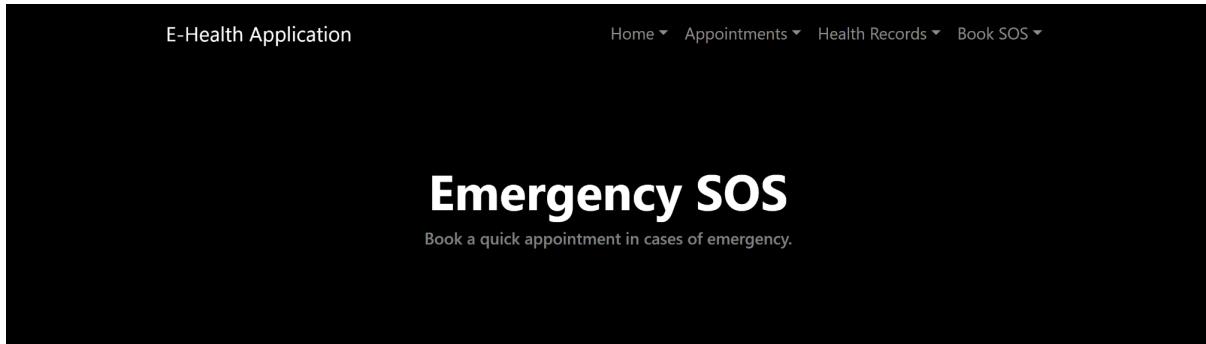


Figure 36. Second Prototype of Emergency SOS Page

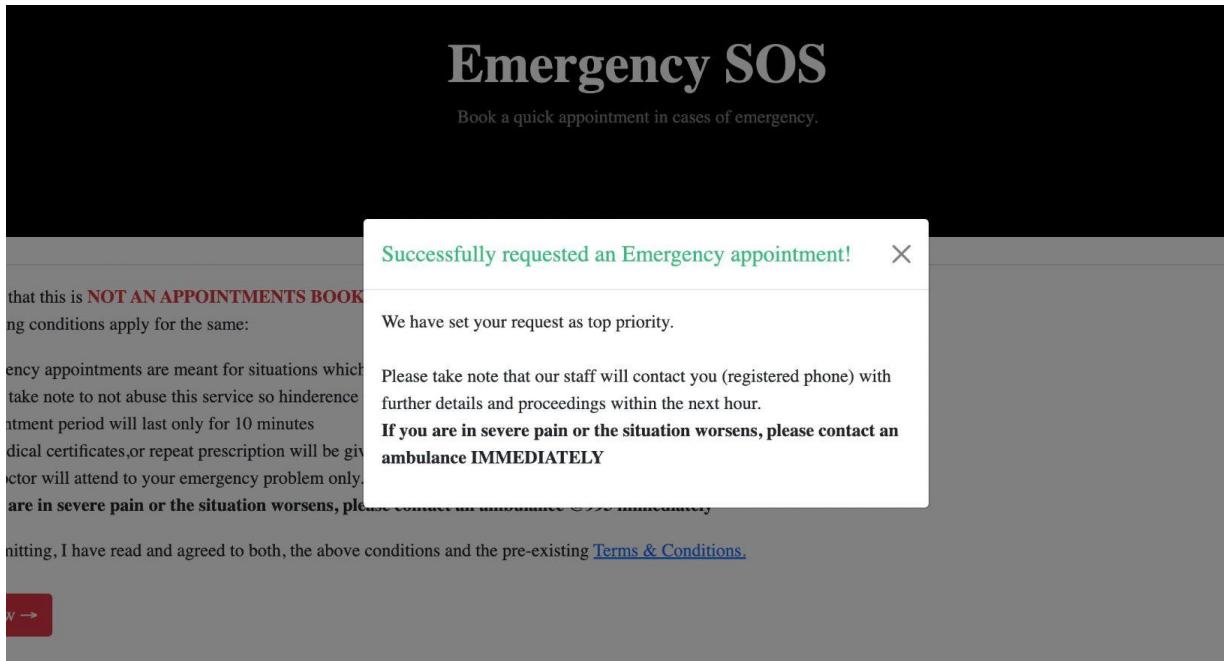


Figure 37. Second Prototype of Emergency SOS Page when booked

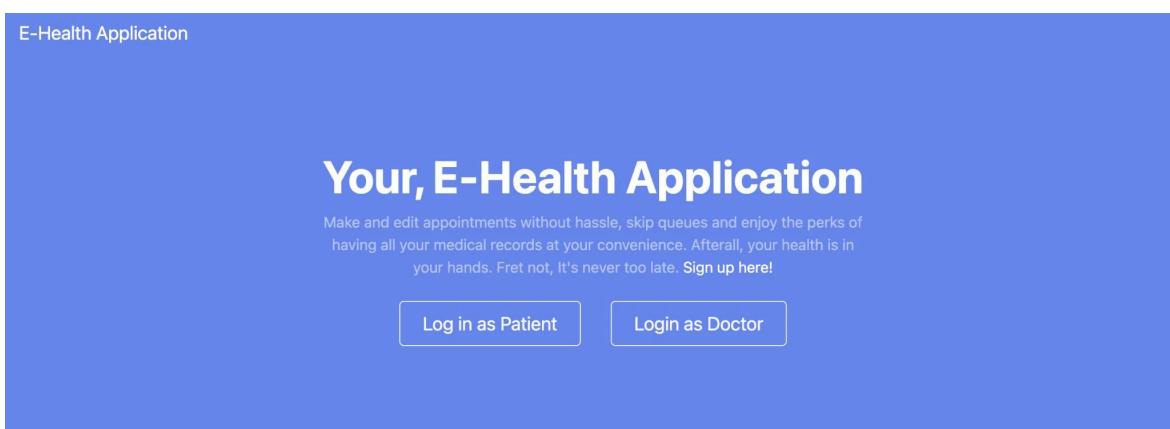
## **Third and Final Prototype**

By the completion of the second prototype, the design and functionality we wanted to achieve for our application would be mostly completed. Thus, for this phase, we decided to continue enhancing our application based on another round of user testing and this time, our focus for the changes in our third and final prototype would be on its design and overall usability.

Hence, another interview was conducted after the completion of the second prototype where it was shown and explained to the same interviewees from the previous interview. Questions were also crafted for this interview beforehand. The interview questions and results are shown below along with pictures of our final prototype.

Questions	Possible Response	Response	Feedback and Observation
<b>With the newly added designs to the application, is it better to understand the application?</b>	Yes	18	All of the elderly understood the application better as compared to the first prototype with the visual designs now in place
	No	0	
<b>With this new prototype, after explanation, do you understand how to use the application and where to access each function you need to do or use?</b>	Yes	16	Most of the elderly understood how to use the application and where to click on to access the function they wanted to carry out with the exception of some who took longer to understand.
	No	2	
<b>Would you be confident to use the application now on your own?</b>	Yes	14	Most of the elderly were confident to be able to use the application on their own while some voiced that they might forget and need some explanation or help in the future again.
	No	4	
<b>Do you like the visual design of the application overall?</b>	Yes	18	All of the elderly enjoyed and liked the design of the application.
	No	0	
<b>Are the colours we used for the design okay for your eyes? Is it easy to view the words in the application or would another colour be better?</b>	Yes	9	Half of the elderly said the colours were okay for their eyes while half of them pointed out that some parts of the application were hard to see and that they had to strain their eyes.
	No	9	

Table 2. Results and feedback from the questions asked during the second interview with the elderly

The image shows three main features of the application: "Book and Manage Appointments" (green icon), "View Medical Records" (purple icon), and "Emergency SOS Appointment Featuring" (red icon).

**Book and Manage Appointments**  
Waiting for your doctor when at medical unease can be very frustrating. Book an appointment before hand to avoid the hassle of queuing.

**View Medical Records**  
Your health at your very fingertips. Have access to all your medical information with just the press of a button. You now don't have to pay consultation fees just to just get updates on your health.

**Emergency SOS Appointment Featuring**  
Need an immediate appointment? Don't be afraid. Our SOS feature helps you get priority access to an expert without having to wait. You're never alone.

---

### Get your health in check

Join and Sign Up Here!

ⓘ  
As in NRIC (or Identity Card)

ⓘ

ⓘ

ⓘ  
Password must be 8-20 characters long, contain letters and numbers, and must not contain spaces, special characters/ emojis.

**Sign me up!**

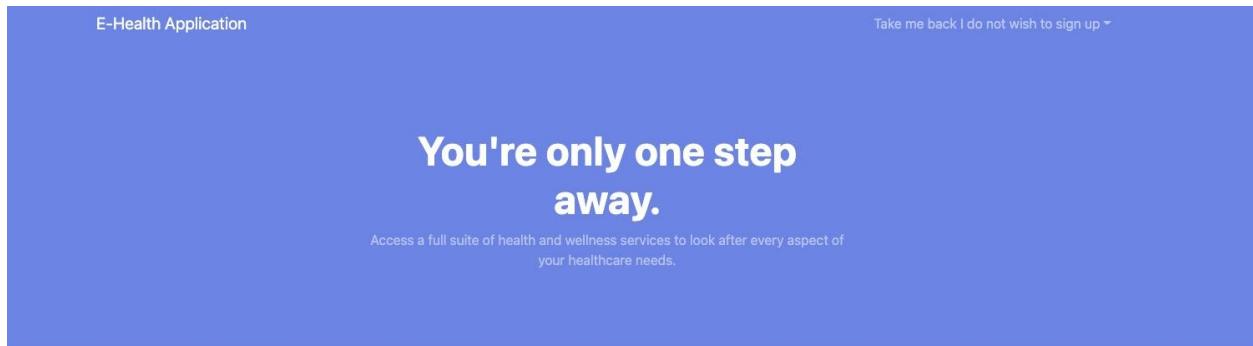
By submitting, you agree to our [Terms & Conditions](#)

Figure 38. Final Prototype of Index Page (Main Log-in Page)

## Patient testimonials



Figure 39. Final Prototype of Index Page (Main Log-in Page)(continued)



**Medical Declaration Form**  
Please help us by providing some details so our services can be tailored to better suit your medical needs.

Gender:

Date of Birth:  dd/mm/yyyy

Age:

Nationality:

PH:

Emergency Contact:

Your Residential Address:

Note that we will use your location to contact your nearest hospital in case of an emergency and other purposes

Figure 40. Final Prototype of Sign-up Page (Medical Declaration Page)

Family history of any illness

Are you on any medications

Any Medical Allergies

Universal vaccinations you have received

Add any specific notifications to doctor (if you wish to)

**Sign me up!**

By submitting, you agree to our [Terms & Conditions](#)

Figure 41. Final Prototype of Sign-up Page (Medical Declaration Page)(continued)

E-Health Application      Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

## Welcome, Prasanna.

Access a full suite of health and wellness services to look after every aspect of your healthcare needs.



### View My Appointment(s)

Access the status of your scheduled/active appointments.

[Appointments →](#)



### Book New Appointment

Schedule yourself a new appointment at your hospital here.

[Book New Appointment Now →](#)



### Manage Appointment(s)

Reschedule or Cancel your active appointments here.

[Manage Appointments Now →](#)



### Access my personal Medical Record(s)

Access, read and manage your personal medical records here.

[View My Health Records →](#)



### Emergency SOS Appointment

Schedule a priority appointment for medical emergencies here. **In case of severe emergency call an ambulance!**

[Book SOS now →](#)

**Log Out**

Sign out of your account here.

Figure 42. Final Prototype of Patient Home Page after signing in

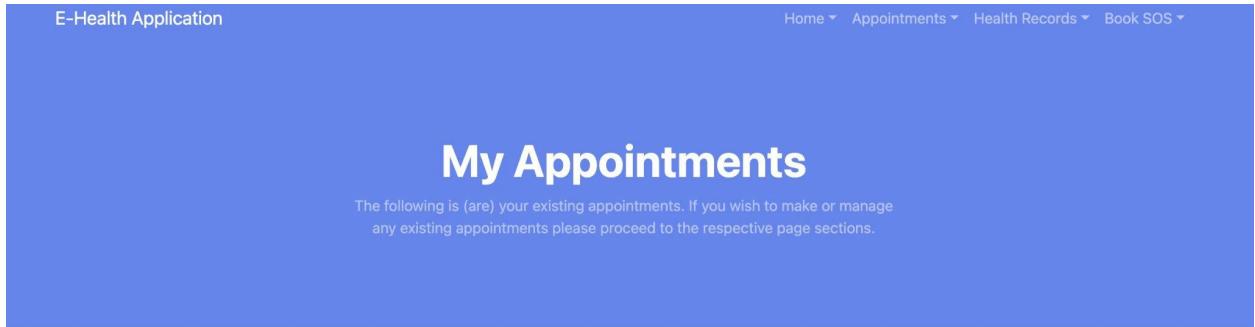


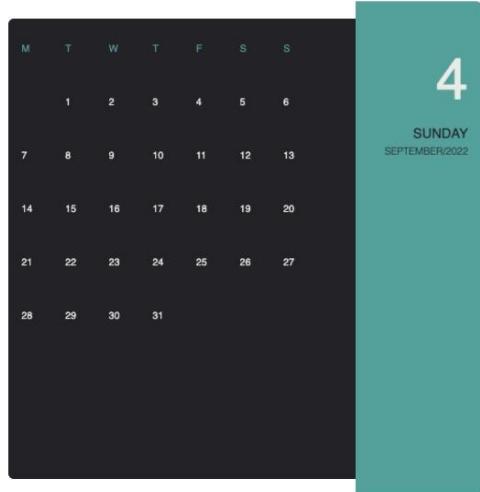
Figure 43. Final Prototype of View Appointments Page

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

## Book a New Appointment

Make a new appointment at your nearest (registered) hospital. Do not worry, we will find the one suited to your needs with your registered details.



29th August 2022

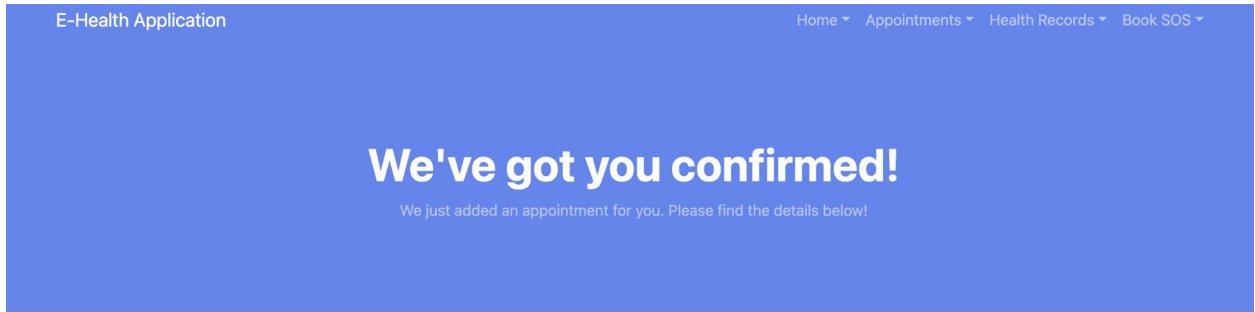
### Proceed to fill the form below.

Reason for Visit
Appointment Date dd/mm/yyyy
Appointment Time --:-- --
First Name (as registered)
Last Name
PH

**Book my Appointment**

By submitting, you agree to our [Terms & Conditions](#)

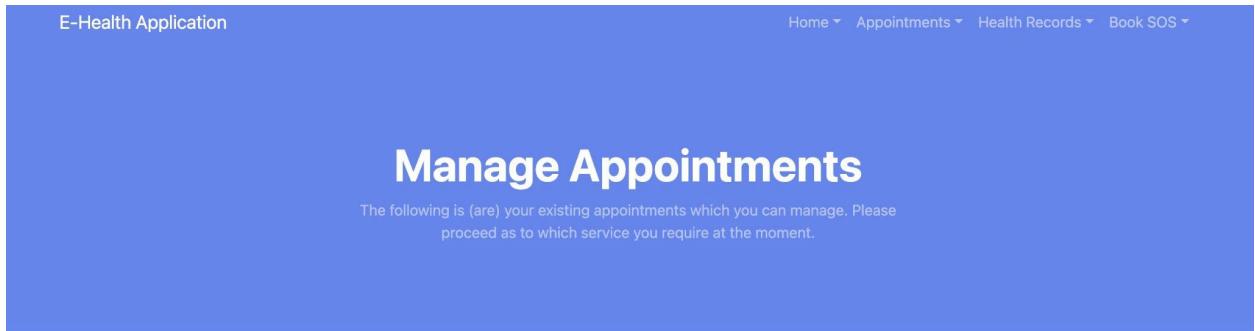
*Figure 44. Final Prototype of Book Appointments Page*



Service	Date and Time	Status
Blood Health Checkup	22/09/2022,22:03	Confirmed

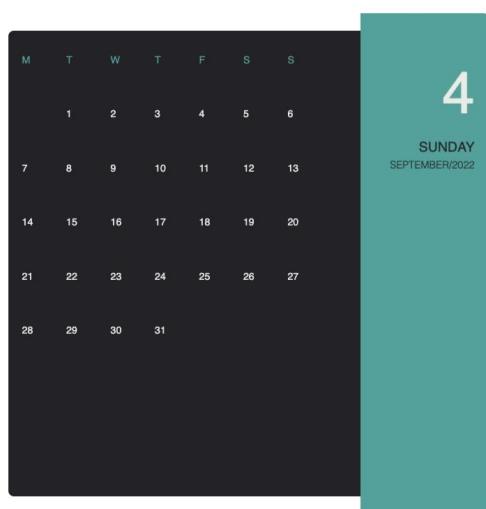
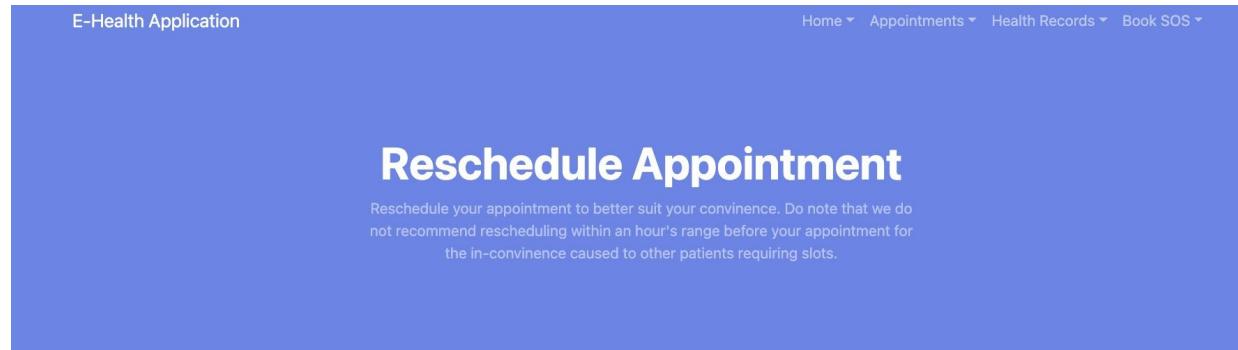
[Add Another Booking →](#) [Reschedule Booking →](#) [Delete Booking →](#)

Figure 45. Final Prototype of Booking Confirmed Page



Service	Date and Time	Status	Actions
Blood Health Checkup	22/09/2022	22:03	<a href="#">Reschedule →</a> <a href="#">Delete →</a>
Blood Health Checkup	22/09/2022	22:03	<a href="#">Reschedule →</a> <a href="#">Delete →</a>

Figure 46. Final Prototype of Manage Appointments Page



## Reschedule by revising the form below.

Reason for Visit

Revised Date  
dd/mm/yyyy

Revised Time  
--:-- --

First Name (as registered)

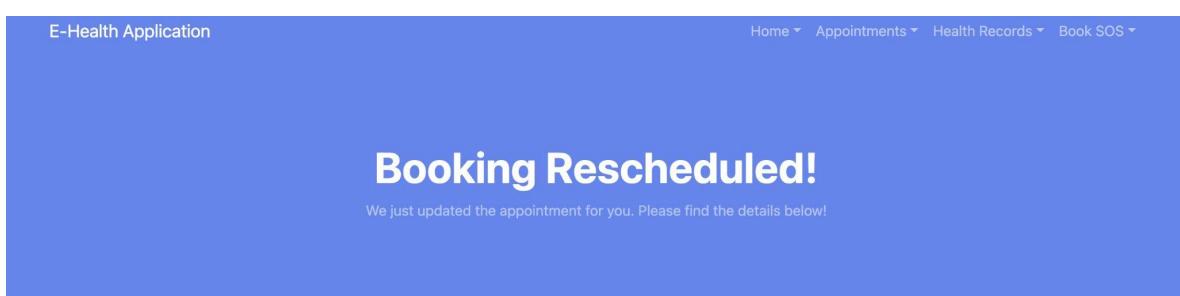
Last Name

PH

[Update my Appointment](#)

By submitting, you agree to our [Terms & Conditions](#)

Figure 47. Final Prototype of Reschedule Appointments Page



Service	Date and Time	Status
Blood Health Checkup	22/09/2022,22:03	Updated & Confirmed

[Add Another Booking →](#)
[Delete Booking →](#)

Figure 48. Final Prototype of Reschedule Confirmation Page

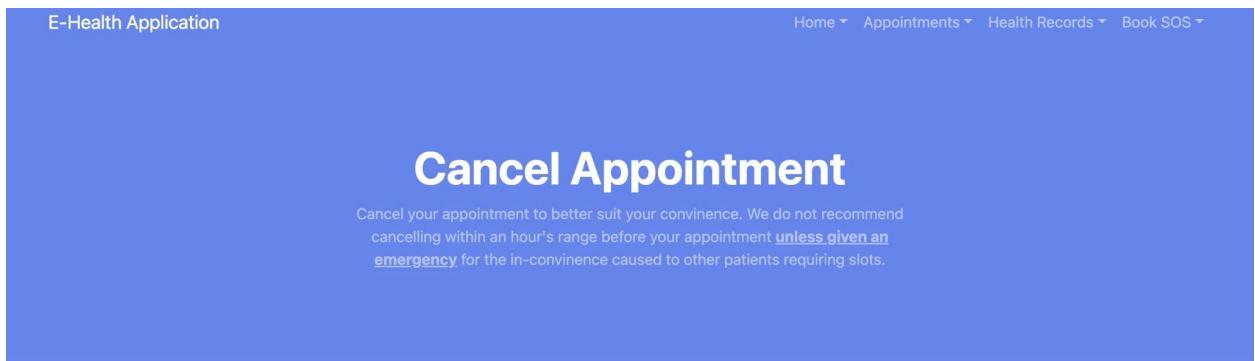


Figure 49. Final Prototype of Cancel Appointment Page

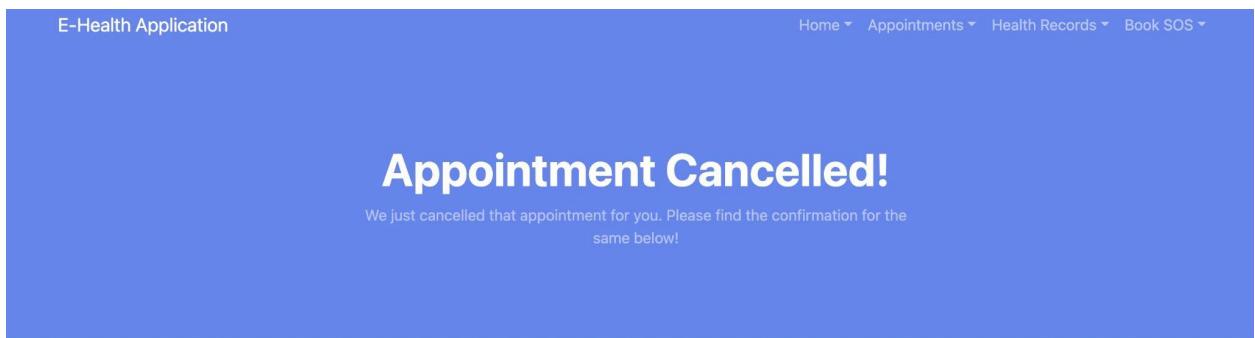


Figure 50. Final Prototype of Cancel Confirmation Page

E-Health Application

Home ▾ Appointments ▾ Health Records ▾ Book SOS ▾

## Past Medical Records

View your past medical history, and request for official data from the hospitals where you may possibly have records at.

### Retrieve Records from Hospitals

Find basic records from hospitals here. Includes (Inpatient Date, Outpatient Date, Unique ID, and Case of Admission)

Retrive →

**Personal Particulars**

Patient Name comes here  
Age  
Gender  
DOB  
Address

### Request for Detailed Medical Report

If you wish to retrive professional medical reports, including (but not limited to CTs, Blood Tests, Surgery Reports/ Rich Media Statements) proceed with the form below.

Please note this request will take atleast **5 business days** to respond to. The results of your application will be sent to the email address you have registered with us. **For urgent requests**, contact our helpdesk at +65 66654455

Full Name (First, Last)

Hospital Name

Reason for request

Can you provide us more info?

Submit →

By submitting, you agree to our [Terms & Conditions](#)

**Long Term Prognosis**

Retrieve prognosis from the dbs and display

**Vaccinations**

Inactivated poliovirus vaccine (IPV), 27th March 2001  
DTap Vaccine, 30th March 2001  
Tuberculosis Vaccine, 4th April 2001  
TT and DTaP (Booster), Vaccine, 3rd January 2017  
Pfizer Covid (2 Completed Doses), 3rd January 2022  
Pfizer Covid (Booster Dose), 3rd April 2022

Figure 51. Final Prototype of Patient Medical Records Page

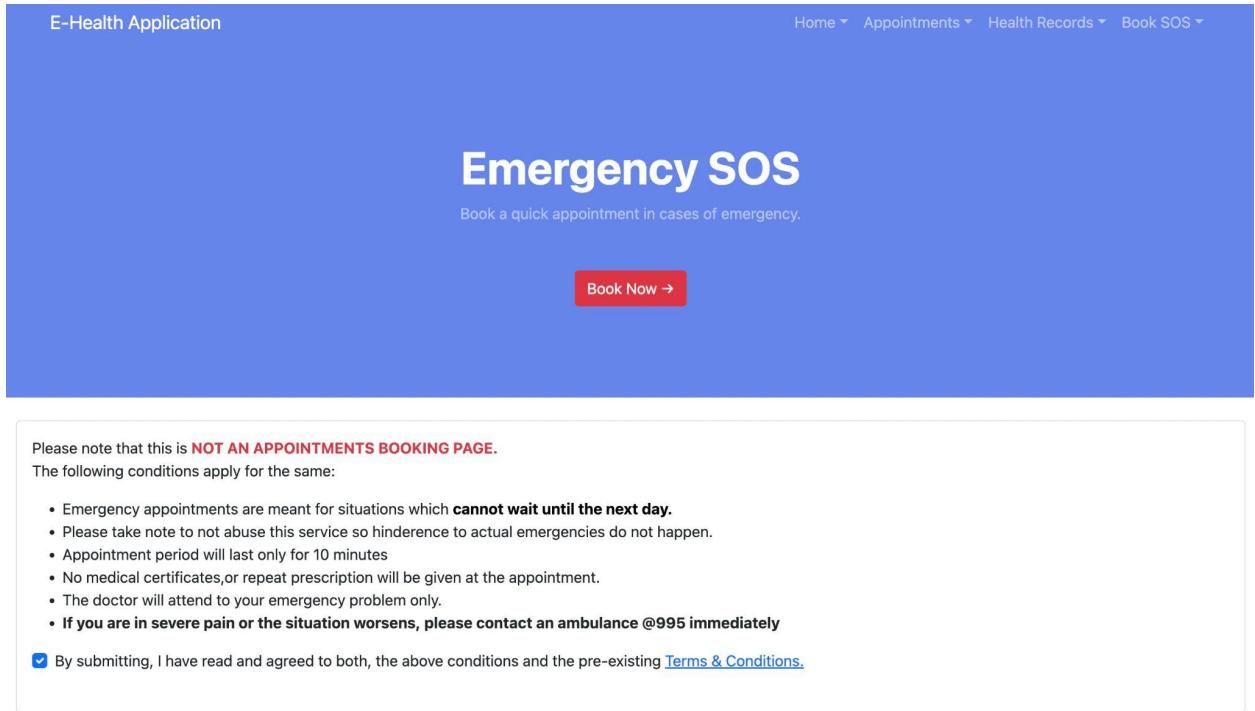


Figure 52. Final Prototype of Emergency SOS Page

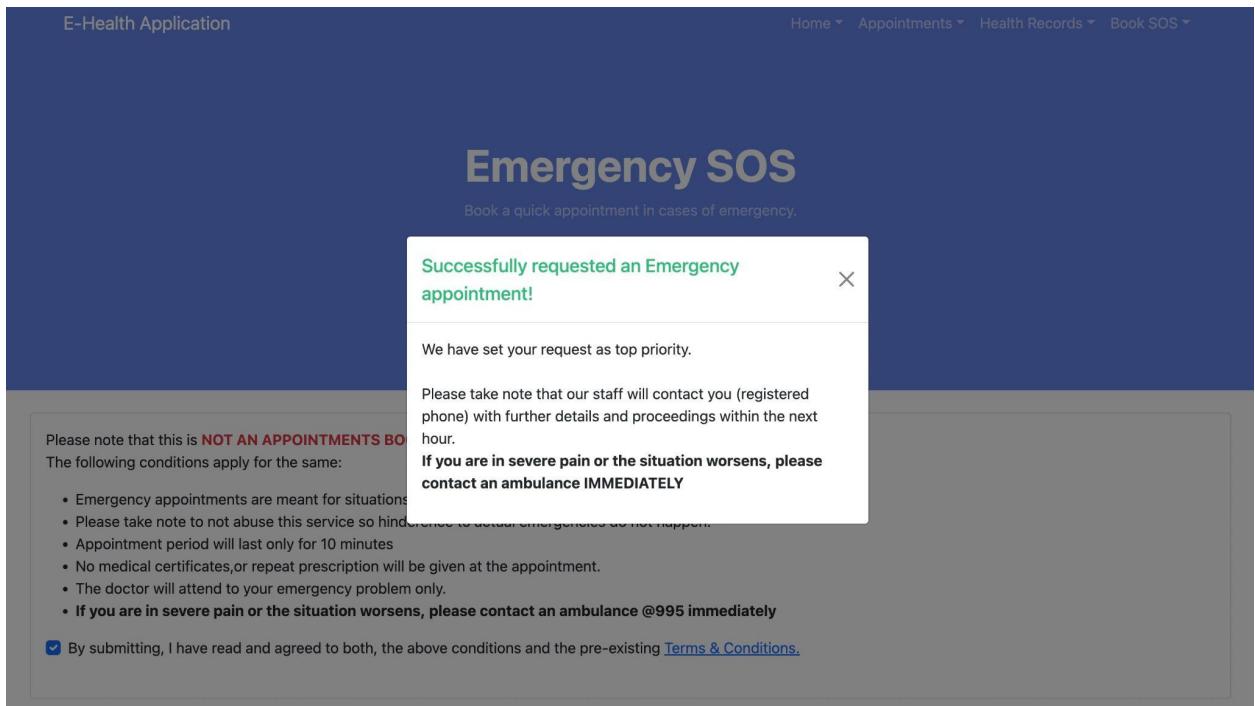


Figure 53. Final Prototype of Emergency SOS Page when booked

## **Changes made from the second prototype to the final prototype**

The main change between the two prototypes was the design aspect of our application based on the user feedback. As seen from the images of the final prototype, the colour scheme of all pages has been changed to better suit our target audience, making it more aesthetically pleasing and easy on the eye of the user. The reasoning behind the design and appearance changes are explained in the following section.

# Design

---

It is quite visible that we as a team have decided and implemented very significant changes from what we had discussed and agreed on in our initial prototyping. We prefer, as previously mentioned, and as stated in the supporting documents, to have the main skeletal website working first. A further stage of development was followed with respect to the design. We realised that the same could be improved from user feedback from our target audience and that multiple different aspects could be developed in order to make it more usable for them in the future.

The main changes or highlights of our revised design were as follows:

**1) Visual Appeal:** Observe from the screenshot (Figure 54) below that an initial design without CSS is not something a software developer would even consider delivering to a client. Responsiveness, architecture, consistency, web navigation, colour scheme, and usability are some of the most basic characteristics of a design that can be rated low. Once we had thoroughly analysed the resources available to us, we decided to use bootstrap. We meticulously designed our 'cards' system, 'grid system', 'positioning and margins', 'feature scaling and sizing', and many other factors to make them feasible for senior users. To illustrate: we used the inbuilt bootstrap utilities class and CSS to address the majority of the issues above. (as shown in Figure 54).

The screenshot shows the 'Your, E-Health Application' sign up page. At the top, there's a header with the application name and a 'Sign up here!' link. Below the header are two login buttons: 'Log in as Patient' and 'Log in as Doctor'. The page is divided into sections for 'Patient Particulars' and 'Doctor Particulars', each containing input fields for 'User Name' and 'Password', and 'Close' and 'Log In' buttons. There are also sections for 'Book and Manage Appointments' (with a note about avoiding queues), 'View Medical Records' (with a note about fingertip access), and 'Emergency SOS Appointment Featuring' (with a note about priority access). The overall layout is simple and lacks the visual appeal and structure provided by CSS.

Figure 54. Sign up page without CSS

(sign up page has been used as an example but note that this was the plain layout for all pages, can be referred to in 'beforecss.pdf file')

# Your, E-Health Application

Make and edit appointments without hassle, skip queues and enjoy the perks of having all your medical records at your convenience. Afterall, your health is in your hands. Fret not, It's never too late. [Sign up here!](#)

[Log in as Patient](#)

[Login as Doctor](#)



## Book and Manage Appointments

Waiting for your doctor when at medical unease can be very frustrating. Book an appointment before hand to avoid the hassle of queuing.



## View Medical Records

Your health at your very fingertips. Have access to all your medical information with just the press of a button. You now don't have to pay consultation fees just to just get updates on your health.



## Emergency SOS Appointment Featuring

Need an immediate appointment? Don't be afraid. Our SOS feature helps you get priority access to an

## Get your health in check

Join and Sign Up Here!

First Name

As in NRIC (or Identity Card)

Last Name

Email address

Password

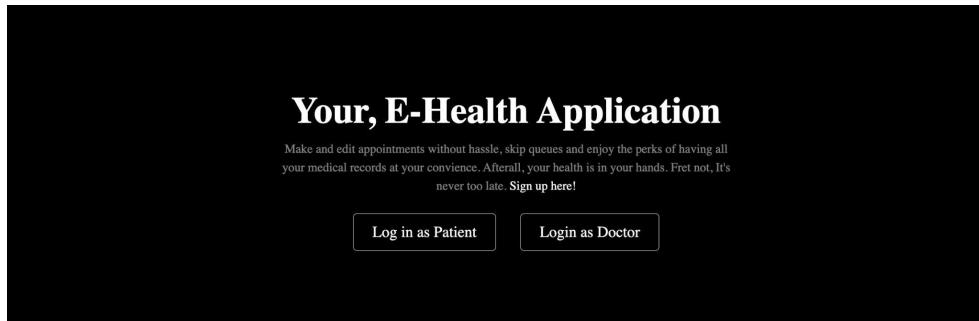
Password must be 8-20 characters long, contain letters and numbers, and must not contain spaces, special characters/ emojis.

[Sign me up!](#)

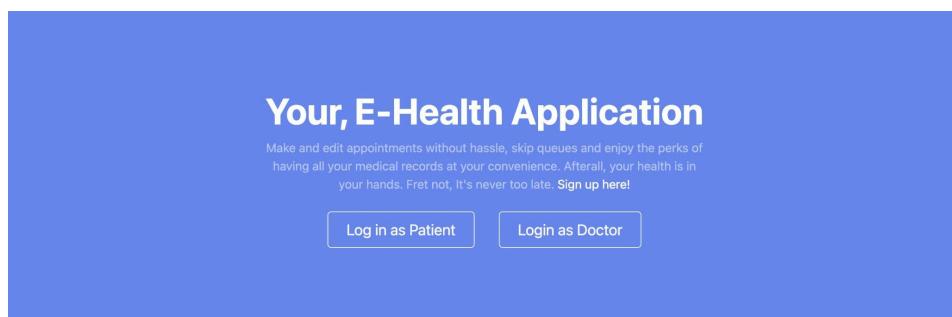
By submitting, you agree to our [Terms & Conditions](#)

Figure 55. Sign-up page with bootstrap and CSS

**2) Colour schema:** Our e-health app primarily targets senior citizens. In light of this, we made sure that our UI/UX designs were comfortable for them and best adapted to their demands. The initial colour schema as shown in Figure 56 is very high in contrast and definitely would not be comforting to their eyes which are undeniably weaker given their age. As a result, brighter and gentler hues, particularly green, blue, and grey, were chosen (but not too light, or else it would affect distinguishability).



*Figure 56. High contrast schema*



*Figure 57. Softer, and better-suited schema*

**3) Fonts:** Given the same overall project scope, we selected sans-serif fonts (a type style without serifs) to facilitate easier and longer-lasting use for the elderly. Sans serif typefaces come in a range of sizes and forms and are thought to be more contemporary. This typeface's "sans" suffix refers to the absence of strokes at the extremities of the letters. This lack of further complexity is regarded to be what makes the type category represent simplicity. Although character edges can be either sharp or rounded, sans serif styles have a direct and accurate appearance.

We hence decided to use sans serif typefaces to create the website since they are most often connected with the mood and feelings of modernity, friendliness, directness, cleanliness, and minimalism. (ref Figure 58).

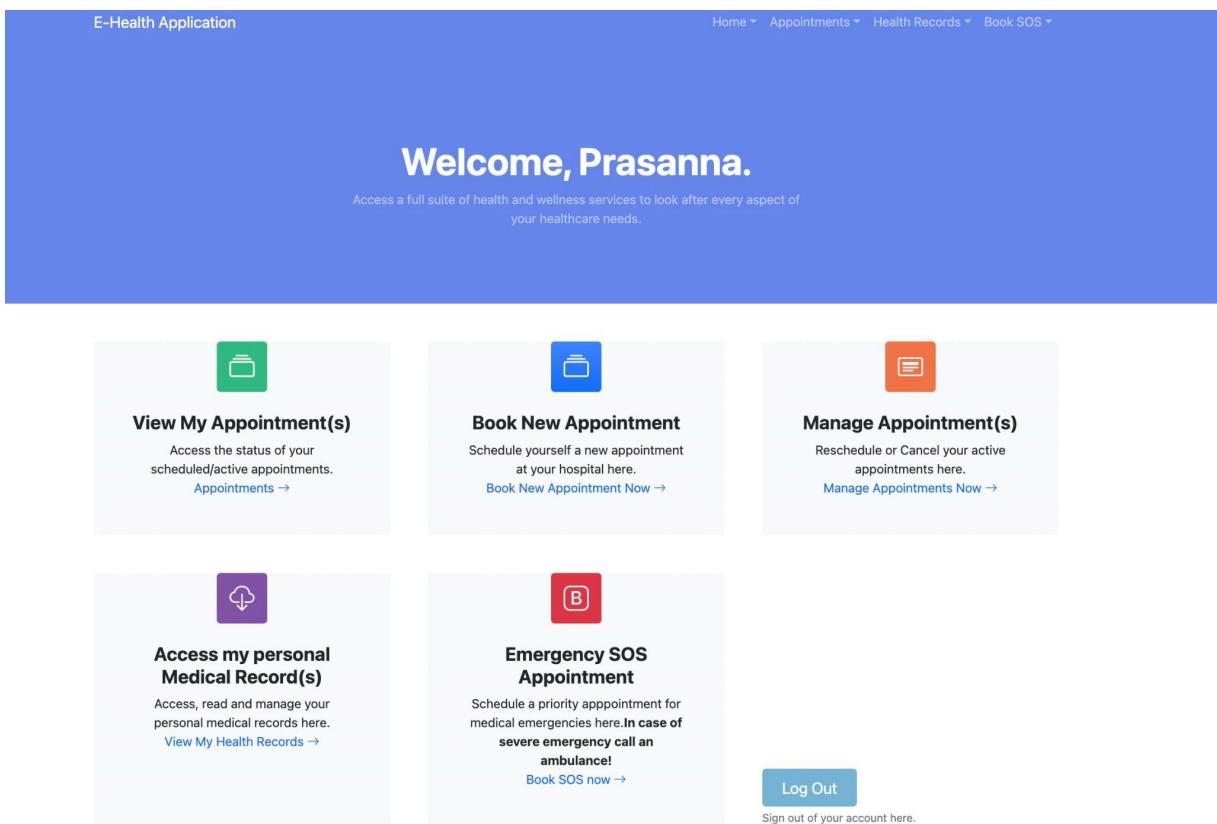


Figure 58. feasible font type, and sizing (example)

**4) Avoided over-embellishing the design:** Avoiding the overcrowding of the website was one of the most important guidelines we wanted to make sure was adhered to. Our main recommendation was to make the features and content straightforward so that the target audience wouldn't feel overpowered by them. We all thought that when it came to elders, their comfort ought to come first. To do this, we kept our user interface (UI) incredibly simple for their convenience. Apart from the flip calendar on the page where patients may book and reschedule appointments, which allows them to select a day and view the available time slots for that day, we did not incorporate any further animations. Additionally, we made sure that each feature was accessible and usable from any other placement of buttons (especially the SOS page).

# System Development

---

## Conceptualisation of Application

During the infancy stages of the development process, much of the application's key functionalities were derived from user stories. By breaking down user stories into appropriate tasks, we are then able to assign them their order of priority to be completed. Taken from our midterms proposal, the table below depicts the process of breaking down user stories into workable tasks:

User Story	Product Backlog	Priority	Assigned Developer	Assigned Sprint
As a user, I want to be able to register an account with my medical details easily so that my medical information can be remembered and provided for me to make appointments.	Medical details form during registration (frontend)	High	Oliver	Sprint 1
	Medical details form during registration (backend)	High	Oliver	Sprint 2
	Medical details form during registration (styling)	Medium	Prasanna	Sprint 2
	New user registration and login (frontend)	High	Mohan	Sprint 1
	New user registration and login (backend)	High	Mohan	Sprint 2
	New user registration and login (styling)	Medium	Prasanna	Sprint 2
As a user, I want to be able to see when I can make appointments and what appointments I currently have, so that I can track my scheduled appointments and know when I am able to make one.	Appointments page (frontend)	High	Prasanna	Sprint 1
	Appointments page (backend)	High	Wei Xiang	Sprint 2
	Appointments page (styling)	Medium	Prasanna	Sprint 2
	Book appointments page (frontend)	High	Wei Xiang	Sprint 1
	Book appointments page (backend)	High	Wei Xiang	Sprint 2
	Book appointments page (styling)	Medium	Prasanna	Sprint 2
	Reschedule appointment + confirmation (frontend)	High	Wei Xiang	Sprint 1

	Reschedule appointment + confirmation (backend)	High	Wei Xiang	Sprint 2
	Reschedule appointment + confirmation (styling)	Medium	Prasanna	Sprint 2
As a user, I want to be able to cancel my appointments in the event I am unable to go for it.	Cancel appointment + confirmation (frontend)	High	Ye Myat	Sprint 1
	Cancel appointment + confirmation (backend)	High	Ye Myat	Sprint 2
	Cancel appointment + confirmation (styling)	Medium	Prasanna	Sprint 2
As a user, I want to be able to have medical advice be rendered available to me immediately in the event I really need it.	Emergency SOS appointment (frontend)	High	Oliver	Sprint 1
	Emergency SOS appointment (styling)	Medium	Prasanna	Sprint 2
As a medical staff, I want to be able to access patient's appointment details and medical information so that I can plan the schedules for the day.	Database schema and tables creation	High	Darren	Sprint 2
	Doctor's login	Low	Wei Xiang	Sprint 3
	Doctor's access on patient appointments page	Low	Wei Xiang	Sprint 3

Table 3. User Story-Task Prioritastion Table

Having the user stories broken down into multiple subtasks gives the team a better view of the technical requirements needed to bring the software to fruition. These tasks were then chosen and tagged to their respective sprint planning boards found in the images above. The team made use of Jira, which is a powerful project management and planning tool. Using the scrum sprint planning boards provided by Jira, we were able to keep track of our sprints and progress.

## Git and the GitHub repository

GitHub is a version control tool for software that provides various functionalities such as backing up files, reverting commits, branching, merging and many more. The platform allows the team to work on individual segments of the project and conveniently upload and clone the updated codebase. Codes that every developer has completed will be uploaded to the GitHub repository either directly via the user interface on GitHub or Git command line. The following image depicts some of the commits done on GitHub by the team:

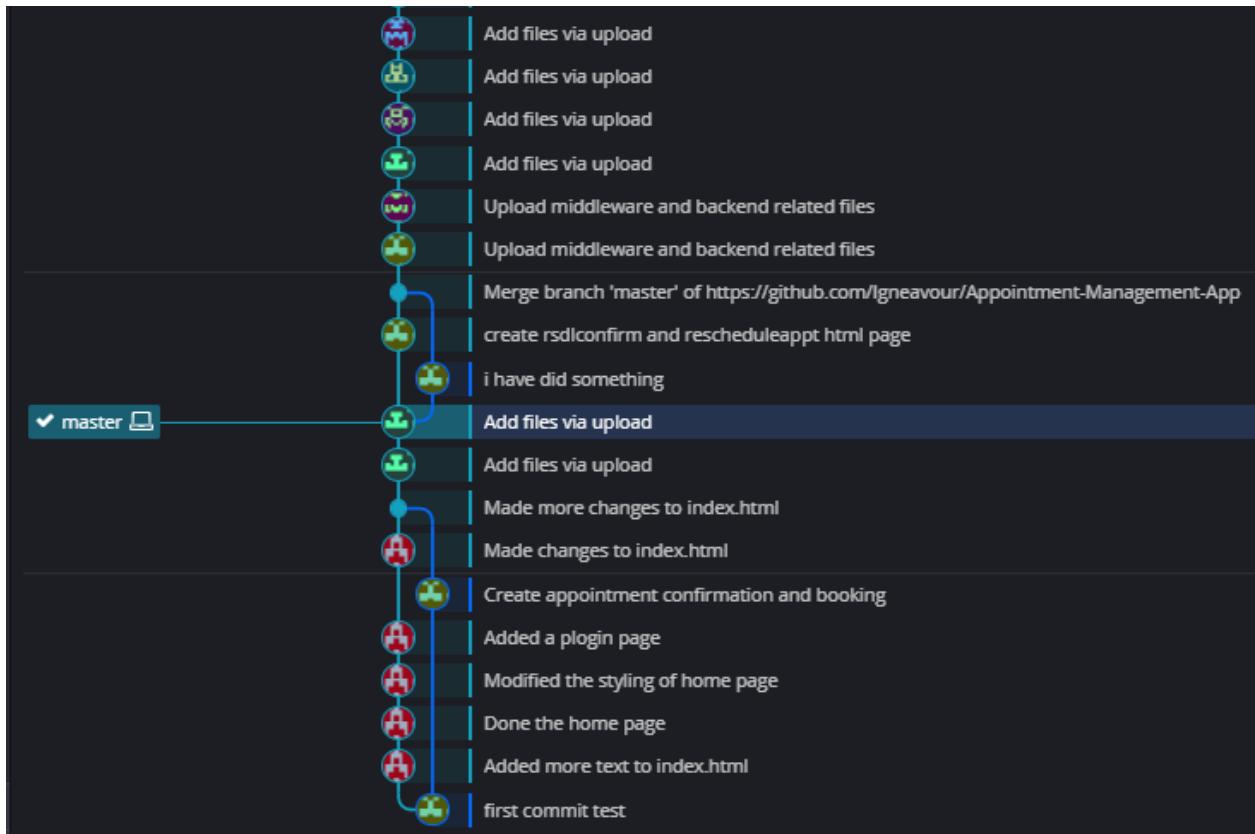


Figure 59. Frontend commits taken from GitKraken

As seen from the image, the commits were made to GitHub by the team. However, a frank review of the commits is that it does leave much to be desired. There were attempts by a few team members to write a proper commit message. However, the remaining commits were mostly either written in the non-standard format or default commits written upon directly uploading files to GitHub which also raises another issue: Most changes were made directly on the master branch with little to no branching. This may be fine for a small-scale project such as this, but issues may arise for projects that are on a larger scale. There was also a mild blunder that occurred to the commits above which was the accidental deletion of several commits due to a forced push to the repository. It happened roughly between the “create rsdlconfirm and...” and “upload middleware and...” commits. The commits that were affected were then re-uploaded to the repository later as seen from the multiple file uploads subsequently. Although the commits for the project were not satisfactory, it was a good learning experience for the team and we believe that this would prepare us better for future projects.

# Software Structure and Components

The image on the right is an overview of the file structure of our web application. It includes the front-end HTML files, the middleware routes for linking the front-end inputs to the backend, the back-end files for establishing a connection to the database and the styling for the design of the webpages. We have decided that all codes pertaining to the middleware that fetches or retrieves data would be compiled in the main.js file and can be found in the routes folder. The views directory mainly houses front-end HTML files as well as script files and styling files essential for the running of the web application. Finally, the app.js file contains all the configurations and code that imports the necessary modules to run the web application.

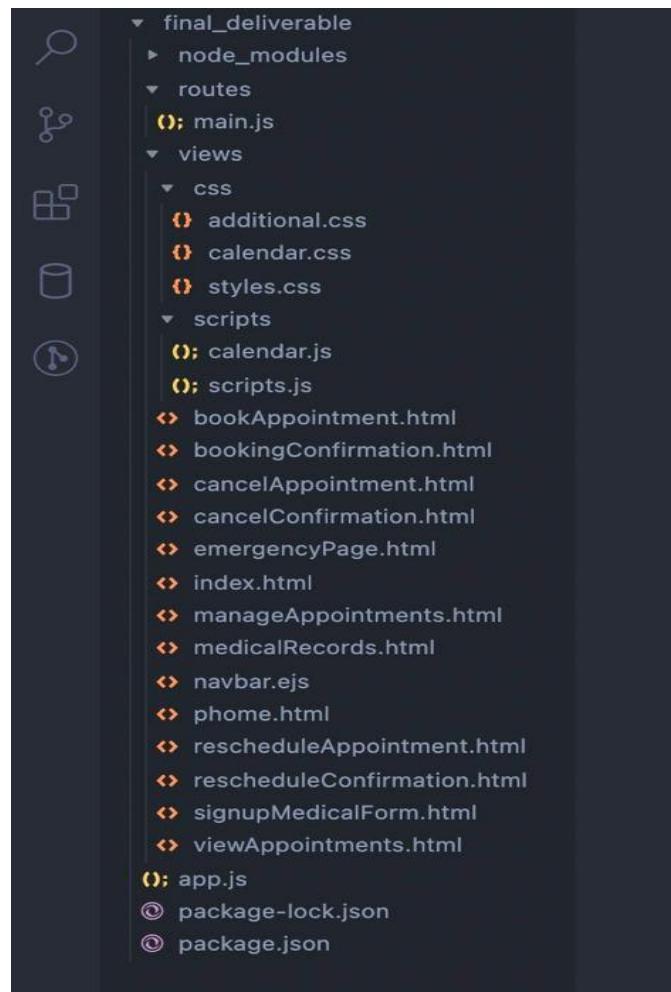


Figure 60. Application's structural overview

## Front-end and styling

As seen from the image on the right, the HTML files along with certain scripts and styling necessary for certain components on the front-end can be found in the views directory.

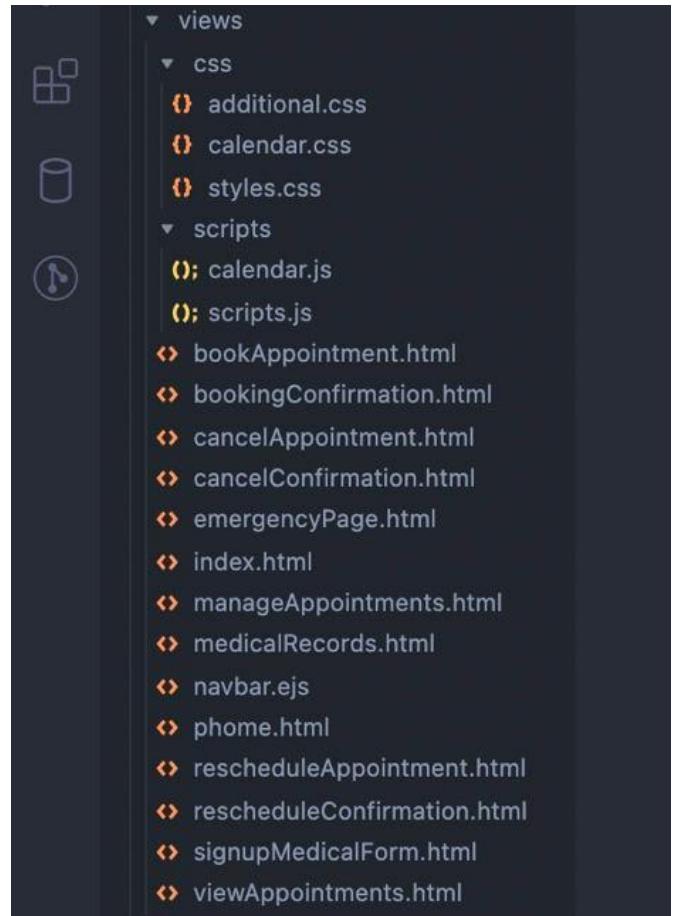


Figure 61. Front-end components in views directory

## Middleware



Figure 62. Middleware components

The main.js file hosts all routes and forms for the web where the input of the users is submitted and sent to the database via the POST method while relevant information of the patient/user is retrieved from the database and relayed to the front-end via the GET method.

## Backend

```
1 const express = require("express");
2 const bodyParser = require("body-parser");
3 const mysql = require("mysql");
4 const cookieParser = require("cookie-parser");
5
6 const app = express();
7 const port = 3000;
8
9 app.use(bodyParser.urlencoded({
10   extended: true
11 }));
12
13 app.use(cookieParser());
14
15 // set the user credentials to use the database server
16 const db = mysql.createConnection({
17   host: "localhost",
18   user: "root",
19   password: "YourRootPassword",
20   database: "health_app"
21 });
22
23 // connect to database
24 db.connect((err) => {
25   if (err)
26   {
27     console.log(err);
28   }
29   else
30   {
31     console.log("Connected to database."); // success message
32   }
33 });
34 global.db = db;
35
36 require("./routes/main")(app);
37
38 app.use(express.static(__dirname + '/views'));
39 app.use(express.static(__dirname + '/css'));
40
41 app.set("view engine", "ejs");
42 app.engine("html", require("ejs").renderFile);
43 app.listen(port, () => console.log(`Example app listening at port ${port}!`));
44
```

Figure 63. App.js backend code

For app.js, it is located outside all the other directories as a standalone file. Here, the necessary modules are imported. Connections to the database are also made in this file. In the event the connection could not be established, the error is caught.

# Black Box Testing

After the team was done with their first iteration of the software, we conducted a few series of black box testing to ascertain if all the functions are working properly as intended.

## Home Page (guest state)

S.N	Scenario	Input	Expected Output	Test Conclusion
1	User would like to create an account	a) Click "Sign up here!" b) Fill in the required fields and click "Sign me up!" button c) Fill in the required fields for medical form and click "Sign me up!"	a) Bring the user to the bottom of the homepage where the signup form is at b) Redirects users to medical form page c) Brings the user to the logged-in state for homepage	Pass
2	User would like to create an account but did not fill in email properly	a) Click "Sign up here!" b) Fill in the required fields but email properly and click "Sign me up!" button	a) Bring the user to the bottom of the homepage where the signup form is at b) User cannot proceed and is prompted that email is invalid	Pass

Table 4. Black box testing for Home Page (guest state)

## Home Page (logged-in state)

S.N	Scenario	Input	Expected Output	Test Conclusion
1	User would like to log out	a) Scroll to the bottom and click on "Log Out" button	a) Redirects user back to the guest homepage	Pass
2	User would like to go to view their appointments	a) Scroll down and click on "Appointments →" OR "Appointments" under the navbar and click "View Existing"	a) Redirects user to View Appointments page	Pass
3	User would like to book a new appointment	a) Scroll down and click on "Book New Appt Now →" OR "Appointments" under the navbar and click "Book New"	a) Redirects user to Book Appointments page	Pass
4	User would like to reschedule/cancel their appointment	a) Scroll down and click on "Manage Now →" OR "Appointments" under the navbar and click "Manage Existing"	a) Redirects user to Manage Appointments page	Pass
5	View Existing Medical Records	a) User should be able to retrieve records from any selected hospital from which they might have records at. b) Fill in form to request rich media reports from selected hospital	a) Display records from hospital(inpatient/outpatient dates, case, etc) if it exists in the database. b) Be able to submit request to hospital c) View default medical records (prognosis, vaccinations etc, in accordance to submitted medical form)	Pass
6	User would like to schedule an emergency appointment	a) Click on "Book SOS now →" b) Click on "Book Immediately →"	a) Redirects user to Emergency Page b) An alert will pop up informing them that the team has been notified and they will get back to the user as soon as possible	Pass

Table 5. Black box testing for Home Page (logged-in state)

### **Book Appointments Page**

S.N	Scenario	Input	Expected Output	Test Conclusion
1	Booking an Appointment	a) Fill in the required fields and click "Book my Appointment"	a) Redirects to 'Booking Confirmation page' where booked info from the DBS is displayed	Pass

Table 6. Black box testing for Book Appointments Page

### **View Appointments Page**

S.N	Scenario	Input	Expected Output	Test Conclusion
1	User would like to add new appointments	a) Click on "Add New Booking" OR Click on "Appointment" under the navbar and click "Book New" b) Fill in required fields and submit	a) Redirects to book appointments page OR Redirects to book appointments page b) Redirects to Booking Confirmation Page and show appointments	Pass
2	User would like to reschedule appointments	a) Click on "Reschedule Booking" b) Fill in required fields and submit	a) Redirects to Reschedule Confirmation Page b) Redirects to Reschedule Confirmation Page and only shows rescheduled appointment	Pass
3	User would like to delete appointments	a) Click on "Delete Booking" b) Click "Delete" button	a) Redirects to Cancel Appointment Page b) Redirects to Cancel Confirmation Page and only shows cancelled appointment	Pass

Table 7. Black box testing for View Appointments Page

### Manage Appointments Page

S.N	Scenario	Input	Expected Output	Test Conclusion
1	User would like to reschedule appointments	a) Click on "Reschedule Booking" b) Fill in required fields and submit	a) Redirects to Reschedule Confirmation Page b) Redirects to Reschedule Confirmation Page and only shows rescheduled appointment	Pass
2	User would like to delete appointments	a) Click on "Delete Booking" b) Click "Delete" button	a) Redirects to Cancel Appointment Page b) Redirects to Cancel Confirmation Page and only shows cancelled appointment	Pass

*Table 8. Black box testing for Manage Appointments Page*

### Medical Records Page

S.N	Scenario	Input	Expected Output	Test Conclusion
1	View Existing Medical Records	c) Users should be able to retrieve records from any selected hospital from which they might have records at. d) Fill in form to request rich media reports from selected hospital	d) Display records from hospital(inpatient/outpatient dates, case, etc) if it exists in the database. e) Be able to submit request to hospital f) View default medical records (prognosis, vaccinations etc, in accordance to submitted medical form)	Pass

*Table 9. Black box testing for Medical Records Page*

### **Emergency Appointments Page**

S.N	Scenario	Input	Expected Output	Test Conclusion
1	Book an SOS emergency	a) Confirm agreement to conditions (default, yes) b) Book confirmation	a) Alert users that a representative will be contacting them immediately with further details & confirmation.	Pass

*Table 10. Black box testing for Emergency Appointments Page*

## Code Rundown

In terms of code, we thought the first step was to build a thorough database to house all the data we would need. For the same, we choose the most sought-after open-source relational database management system in the market, MySQL. We used MySQL workbenches to manage and maintain the same. The database schema we used is attached below.

DBS 1: Health App					
TB 1: Existing Users					
Patient ID	Name	Email	Password		
TB 2: Appointment					
Patient ID	Name	PH	Date	Time	Type
TB 3: Personal Data					
Patient ID	Name	PH	Address	Emerg. Contact	Nationality
TB 4: Medical History					
Patient ID	Name	Prognosis	Hospital	Inpatient	Outpatient
				Medication	Vaccine
				Vaccine Date	

Figure 64. Database Schema

The individual web pages, each of which has the following code architecture and connects to the database in the various ways detailed below, were then the focus of our work.

### 1) Log in Page (index.html),

(find attached code snippets below)

- **Front End:** The front end design is mainly built off of bootstrap plugins, a gridding system and CSS. The design is mainly sectioned to address the two scenarios of user types. If the user is new, they would be directed to sign up and register their details with the organisation. If they are existing users, they can simply just log in. Their entered credentials will be checked by the middleware and if the user credential exists and matches they will simply be redirected to the home page. Else they would be issued an alert given the respective situations.

- **Middleware:** Each field submitted by the user (regardless of whether they are signing up or registering) is extracted from the HTML inputs (retrieved using the basic HTML 'name' attribute) and sent to the database using the body-parser in the npm JS package manager.

- **Backend:**

- 1) **New/unregistered users:** When a new user fills out the form, we perform an insert SQL statement to add the appropriate values to the database using a straightforward 'INSERT INTO' MySQL query. Along with the query statement itself, we send the query function all of these values in the form of an array. They are now taken to the medical declarations page, where they must enter more information so that it may be saved in the database.

```

146 // register a new user
147 app.post('/registered', (req, res, next) => {
148   var name = req.body.fname + " " + req.body.lname;
149   var email = req.body.email;
150   var password = req.body.password;
151   var userCredentials = [name, email, password];
152   // query to insert user details into the database
153   var query = "INSERT INTO existing_users (name, email, password) VALUES(?, ?, ?);";
154
155   db.query(query, userCredentials, (err) => {
156     if (err) {
157       console.log(err.message);

```

Figure 65. Insert new user information into the database

- 2) **Existing users just logging in:** On the whole, it can simply be said that the entered email address is first checked as to its existence in our database. If it does not then an alert is sent immediately. If it does exist, their password is then checked if it matches the password of the user of the same email address entered. Both of these are done using the MySQL 'SELECT' query statement. If the statement returns a value then we know that the user record exists in our database and we can authenticate the user. The alerts are sent out using the built-in HTML alert options.

```

173    // authenticate a user who is Logging in
174    app.post("/submitLogin", (req, res) => {
175      var userName = req.body.name;
176      var password = req.body.password;
177      var credRecord = [userName, password];
178      // query to authenticate a user who is Logging in by checking the credentials
179      var query = "SELECT name from existing_users WHERE name = ? AND password = ?;";
180
181      db.query(query, credRecord, (err, result) => {
182        if(err)
183        {

```

Figure 66. Select Query that checks if user login credentials are valid

- **Cookies:** After registration, we would initially know nothing about the user. In order to address this issue, we save the user's data in a cookie on the browser. We now know who our users are and can find their information in our database. Therefore, in the event that a user refreshes the page or their browser crashes, they wouldn't need to log in again because our cookies kept the user and their account signed in. We have also implemented a feature where in a user wishes to log out, the cookies would be deleted from their system.

```

161    // create a secure cookie to remember the user who signed in
162    res.cookie('name', name, {
163      maxAge: 5000,
164      secure: false,
165      httpOnly: true,
166      sameSite: 'lax'
167    });
168    res.render("signupMedicalForm.html");
169  });
170  });
171});

```

```

// delete user's cookie from browser when they log out
app.get("/logOut", (req, res) => {
  res.clearCookie('name', {domain: 'localhost', path: "/"});
  res.redirect("/");
});


```

Figure 67. Cookie Log In/ Out Functionality

## 2) Medical Declaration (for new users, signupMedicalForm.html)

(find attached code snippets below)

- **Front End:** Simple Bootstrap-powered HTML and CSS code is used to render a form that asks users to declare their medical information. The right HTML "name" attributes are set up to receive the selected or entered value and send it to the middleware.
- **Middleware:** The body-parser in the npm JS package manager is used to extract each field that the user has provided from the HTML inputs (basic 'name' attribute) and send it to the database.

- **Backend:** We use a simple 'INSERT INTO' MySQL query to execute an insert SQL statement to add the necessary values to the database. As illustrated in the supporting excerpts below, we submit the query function of all of these values as an array together with the actual query statement.

```
// medical form post to database
app.post("/submittedMedicalForm", (req, res, next) => {
  // Insert into personal_data db
  var name = req.cookies.name;
  var age = req.body.age;
  var nationality = req.body.nationality;
  var phone = req.body.phone;
  var emergent = req.body.emergencyNum;
  var gender = req.body.gender;
  var birth = req.body.birthDate;
  var address = req.body.address;
  var userInfo = [name, phone, address, emergent, nationality, age, birth, gender];
  var query = "INSERT INTO personal_data (name, phone_number, address, emergency_contact, nationality, age, date_of_birth, gender) VALUES (?, ?, ?, ?, ?, ?, ?);";

  // Insert into medical_info db
  var familyHistory = req.body.familyHistory;
  var medication = req.body.medications;
  var allergies = req.body.allergies;
  var vaccination = req.body.vaccination;
  var moreMedicalInfo = req.body.moreInfo;
  var medicalInfo = [name, familyHistory, medication, allergies, vaccination, moreMedicalInfo];
  var mquery = "INSERT INTO medical_info (name, family_conditions, medication, medical_allergies, covid_vaccinedoses, pastmedical_healthconditions) VALUES (?, ?, ?, ?, ?, ?);";

  // if first query is successful, then execute the next query
  db.query(query, userInfo, (err) => {
    if (err) {
      console.log(err.message);
    } else {
      db.query(mquery, medicalInfo, (err) => {
        if (err) {
          console.log(err.message);
        } else {
          // render the home page to the user
          res.render("phome.html", {userName: name});
        }
      });
    }
  });
});
```

Figure 68. Insert new user medical information from the medical declaration form into the database

### 3) Home Page (phome.html)

(find attached code snippets below)

- **Front End:** We were able to create and release the features of our application in the form of grids on this page that make content more ordered and reachable/accessible for our target customers by utilising the grid system and card system supplied by the inbuilt bootstrap functionalities as shown below. Every single feature (book/manage/delete appointments, make SOS appointments, or examine medical records) is available from this main page and would be redirected upon clicking the proper links (using the nav bar or the cards themselves).

- **Middleware and Backend:** With the exception of displaying the user name in the title card, this page does not actually interact with the backend in any way. (using simple EJS that picks up or retrieves the registered name). However, this aids in showing that all of the user information has been collected, so going forward, every given area of the website will only provide information about the selected or displayed user.

```
<h1 class="display-5 fw-bolder text-white mb-2">Welcome, <%= userName %>.
    |   |   <!-- Get name from databases -->
</h1>
```

Figure 69. validate to user that they have successfully logged in by displaying their user name

#### 4) Book Appointments Page (phome.html)

(find attached code snippets below)

- **Front end:** There are 3 basic sections to the page. One is the navigation where all other features are available, another is the comprehensive calendar section and the last was a form. The built-in calendar allows users to select and date and view all available time slots available for an appointment that day. This calendar system was built using very basic HTML and CSS with an added JS event handler which picks up all the available time slots for that particular code. The form section on the right would allow users to fill in the form with other appropriate details (with "name" attributes set up to receive the selected or entered value and send it to the middleware).
- **Middleware:** The body-parser in the npm JS package manager is used to extract each field that the user has provided from the HTML inputs (basic 'name' attribute) and send it to the database.

```
18 // render the book appointments page
19 app.get("/bookAppointment", (req, res) => {
20     // take the user name from the browser cookies and extract the first and Last name
21     var name = req.cookies.name;
22     var array = name.split(' ');
23     var fname = array[0];
24     var lname = array[1];
25     // pass the first and last name as ejs parameters to the bookAppointment page
26     res.render("bookAppointment.html", {fname: fname, lname: lname});
27 });
28
```

Figure 70. Book appointment page rendering

- **Backend:** With the help of the cookies we save in the browser, we supply the standard user information, such as their first and last names by default in the form itself. (the EJS templating engine is used to pass these values to the front end). We access the name attribute of the input fields to extract the user-entered values after the user has scheduled an appointment, and then we run the insert SQL command to add the appointment information to our database.

```

203 // allow the user to book a new appointment
204 app.post("/bookedAppointment", (req, res) => {
205   var name = req.body.fname + " " + req.body.lname;
206   var number = req.body.number;
207   var dateAndTime = req.body.c_date + " " + req.body.time;
208   var reason = req.body.reason;
209
210   var appointmentRecord = [name, number, dateAndTime, reason, "Confirmed"];
211   // query to insert user appointment details into the database
212   var insertQuery = "INSERT INTO appointments (name, phone_number, date_time, type, status) VALUES (?, ?, ?, ?, ?);"
213
214   db.query(insertQuery, appointmentRecord, (err) => {
215     if (err)
216   {

```

*Figure 71. Insert new Appointment details from input into the database*

## **5) Manage Appointments Page (reschedule, delete & appropriate confirmation Pages) (manageAppointments.html, rescheduleAppointment.html, rescheduleconfirmation.html, and deleteconfirmation.html)**

(find attached code snippets below)

- It is significant to note that all of the active appointments are shown on our manage appointments page (manageAppointments.html). This was accomplished utilising a straightforward table format and a 'SELECT' MySQL query, which retrieves and displays all appointments saved in the database (with the help of EJS). It is from this table that the user can either delete or reschedule their appointments at their convenience.

```

// render the manage appointments page
app.get("/manageAppointment", (req, res) => {
  var name = req.cookies.name;
  var nameRecord = [name];
  // query to get user appointment details from the database
  var query = "SELECT appointment_id, type, date_time, status FROM appointments WHERE name = ?;";

  db.query(query, nameRecord, (err, result) => {
    if (err)
    {
      console.log(err.message);
    }
    else
    {
      var details = result;
      res.render("manageAppointments.html", {details: result})
    }
  })
});

```

Figure 72. ID selector for which any selected appointment can be rescheduled and deleted.

### - Reschedule Appointment

- **Front end:** The front end is similar to that of the book appointments page. There are 3 basic sections on the page. One is the navigation where all other features are available, another is the comprehensive calendar section and the last was a form. The built-in calendar allows users to reselect the date and rechoose amongst the available time slots available for the appointment. This calendar system was built using very basic HTML and CSS with an added JS event handler which picks up all the available time slots for that particular code. The form section on the right would allow users to fill in the form with other appropriate details (with "name" attributes set up to receive the selected or entered value and send it to the middleware). The rest of the form's fields, excluding the date and time sections, are read-only and cannot be modified by the user.
- **Middleware & Backend:** We pass the id of the selected appointment from the frontend to the middleware through the name attribute of the selected button in order to access the particular appointment of the user in the database. We will then update the database with the latest changes. If they choose to reschedule an appointment, they will be sent to a page where they can fill in the new appointment date and time and reschedule it. We run the update SQL statement to update the chosen records that are passed as an input array to be run in the database query.
- **Confirmation:** The newly chosen date and time have been updated in the database. After that, the users are brought to the confirmation page, where the content is

shown as table content (EJS, Bootstrap and CSS). The most recent data is pulled from the database and shown here.

```
// render the reschedule appointments page
app.get("/rescheduleAppointment", function (req, res) {
  var appointmentId = req.query.buttonId;
  var idRecord = [appointmentId];
  // query to get all the user appointment details from the database
  var query = "SELECT * FROM appointments WHERE appointment_id = ?;";

  db.query(query, idRecord, function (error, data) {
    if (error) {
      console.log(error.message);
    } else {
      res.render("rescheduleAppointment.html", { data: data[0] });
    }
  });
});
```

Figure 73. Render updatebooking form for selected appointment the user wishes to reschedule the same with respective details.

```
// reschedule the user's appointment
app.post("/rescheduleAppointment", function (req, res) {
  // query to update user appointment details details in the database
  let sqlquery =
    "UPDATE appointments SET phone_number = ?, date_time = ? WHERE appointment_id = ?";
  let id = req.body.appointment_id;
  let date_time = req.body.c_date + " " + req.body.time;

  let newrecord = [
    req.body.number,
    date_time,
    id
  ];
  // if first query is successful, then execute the second query
  db.query(sqlquery, newrecord, (err) => {
    if (err)
    {
      console.log(err.message);
    }
    else
    {
      // query to search user appointment details from the database
      let query = "SELECT * FROM appointments WHERE appointment_id = ?;";
      let idRecord = [id];
      db.query(query, idRecord, function (error, data) {
        if (error)
        {
          console.log(error.message);
        } else
        {
          res.render("rescheduleConfirmation.html", { appointment: data[0] });
        }
      });
    }
  });
});
```

Figure 74. For selected appointment, with ID '?', update query to update appointment information in the database.

### - Delete Appointment

- **Front End (Confirmation):** Following the user's selection of the appointment to be deleted, a confirmation prompt will ask the user if they are sure they want to proceed with the deletion. Upon approval, users are led to the delete confirmation page, where they are shown a table (EJS, HTML, Bootstrap and CSS) with the details of the deleted appointment.
- **Middleware & Backend:** The appointment details are pulled up based on the ID of the appointment the user selected before going to remove, and the record is deleted from the database using a straightforward 'DELETE FROM' MySQL statement. It is also important to note that only that specific database record is removed. The rest remains unaltered.

```
// render the cancel appointment page
app.get("/cancelAppointment", (req, res) => {
  var idRecord = [req.query.buttonId];
  // query to get user appointment details from the database
  var query = "SELECT * FROM appointments WHERE appointment_id = ?";

  db.query(query, idRecord, (err, result) => {
    if (err)
    {
      | console.log(err.message);
    }
    else
    {
      | res.render("cancelAppointment.html", {details: result[0]});
    }
  });
});
```

Figure 75. Render cancel appointment form for selected appointment the user wishes to cancel the same with respective details

```

// allow the user to cancel an appointment
app.post("/cancelAppointment", (req, res) => {
  var idRecord = [req.body.buttonId];
  // query to search user appointment details into the database
  var searchQuery = "SELECT * FROM appointments WHERE appointment_id = ?;"

  db.query(searchQuery, idRecord, (err, result) => {
    if (err)
    {
      console.log(err.message);
    }
    else
    {
      // pass the cancelled appointment details to the user
      res.render("cancelConfirmation.html", {details: result[0]});
    }
  })
  // query to delete user's appointment from the database
  var deleteQuery = "DELETE FROM appointments WHERE appointment_id = ?;"

  db.query(deleteQuery, idRecord, (err) => {
    if (err)
    {
      console.log(err.message);
    }
  })
});

```

Figure 76. For selected appointment, with ID '?', delete query to remove appointment from database.

## 6) Medical Records (medicalRecords.html)

(find attached code snippets below)

- **Front End:** Additionally, the Medical records page is divided into three sections. We have a patient information section on the right side of the page, where essential medical records, including personal information, prognosis, and vaccination, are provided. Retrieving records directly from hospitals is made easier by the two sections on the left. The first one assists us in retrieving patient records from hospitals, including MISC data like inpatient date, case, etc. The second component is a form that patients can use to seek thorough reports from the hospital, such as CT scans or MRIs, for the right reasons.
- **Middleware & Backend:** The 'SELECT FROM' MySQL query, which renders all the pertinent patient-related medical records, is assisted in rendering data by the middleware EJS and is then displayed in the bootstrap table.

```
// render the user's medical record page
app.get("/medicalRecord", (req, res) => {
  var name = req.cookies.name;
  var nameRecord = [name];
  // query to get user personal details from the database
  var personalQuery = "SELECT name, age, gender, date_of_birth, address FROM personal_data WHERE name = ?;";
  // query to get user medical details from the database
  var medicalQuery = "SELECT family_conditions FROM medical_info WHERE name = ?;";

  // if the first query is successful, then run the second query and pass all the data
  // from the database into the medical records page
  db.query(personalQuery, nameRecord, (err, result) => {
    if (err)
    {
      | console.log(err.message);
    }
    else
    {
      | db.query(medicalQuery, nameRecord, (err, medicalData) => {
        |   if (err)
        |   {
        |     | console.log(err.message);
        |   }
        |   else
        |   {
        |     | res.render("medicalRecords.html", {details: result[0], data: medicalData[0]});
        |   }
      })
    }
  });
});
```

Figure 77. Render Medical Records Page

## **7) Emergency SOS (emergencySOS.html)**

- **Working Synopsis:** On a card and a button in the front end area are all the conditions and limitations that apply to the SOS feature. Only after accepting the aforementioned terms and restrictions could the user make an SOS appointment. Additionally, a high alert notification is delivered to the hospital's end and in turn, a hospital representative contacts the user right away and offers them assistance with their emergency appointment.

# Analysis

---

It is undeniable that there are several telehealth applications available right now. One of the most well-known ones (in the Asian setting) is Doctor Anywhere. Consumers may get convenient healthcare services at their fingertips thanks to Doctor Anywhere. The healthcare professional anywhere app allows users to access a variety of services including consultation and claims management to prescription and wellness items, saving them time travelling to clinics or hospitals and waiting to be seen by a doctor. Even though our program was unable to implement firewall authentication or BigQuery, unlike the doctor anywhere app, we were able to comprehensively solve the app's major flaws, including user flexibility and navigational ease. The same has been evidenced in the images below.

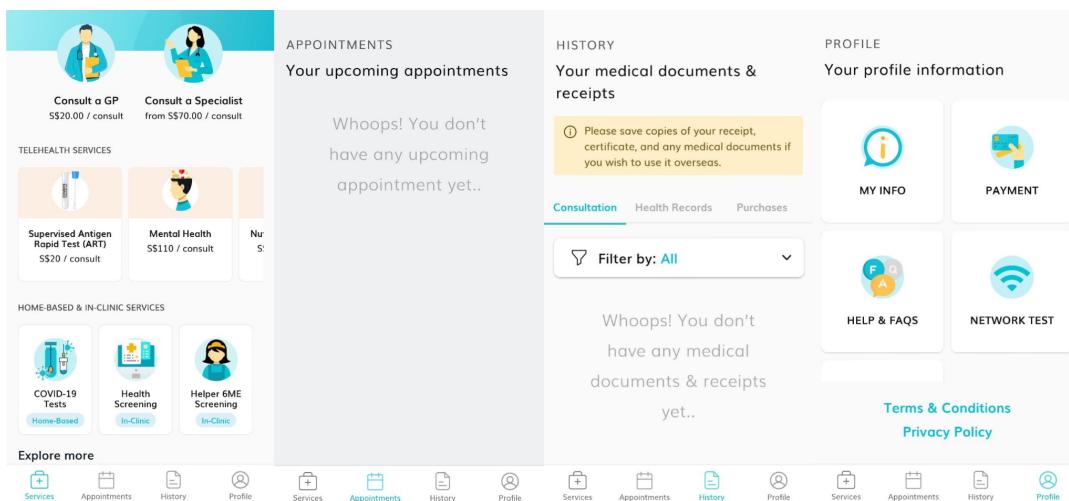


Figure 78. Outlook of Complex Navigation of the Doctor-Anywhere App

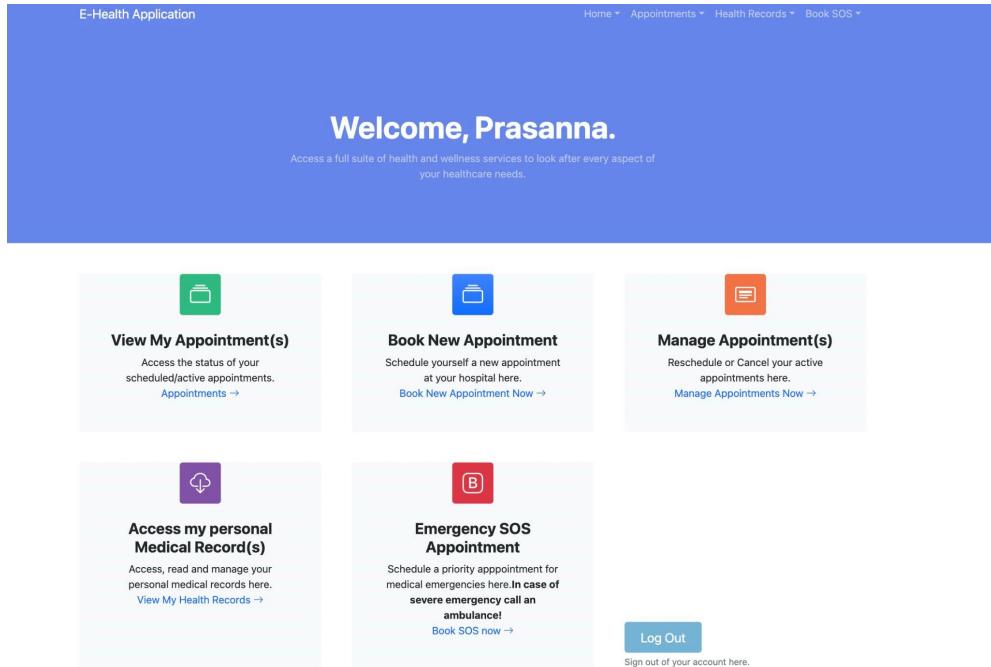


Figure 79. Simple Navigation of our E-Health Application

Overall, there were two key components to the project: (managing, tracking, and committing changes), and (testing, developing, and debugging). With the aid of several open source software programs, including the previously mentioned GitHub and Jira, the first half of the project's needs were successfully met. We were able to allocate manpower, track modifications, and manage work leveraging the same software, which vastly improved team productivity. It also meant we had to track & update/commit so frequently and on a regular basis. It was something new we learned and undoubtedly took us some time to adjust. Our team's ability to maintain consistency and keep going was entirely due to our sincerity, and motivation.

The project's second half required substantially more time and complexity. It was constructed using Bootstrap, JS, and CSS as its three major building blocks. Very simple CSS served as the foundation for the primary structure. Following this, the website's architecture was developed using the well-known front-end framework known as Bootstrap. It took our team some time to fully comprehend the framework's various components, including the grid system and other crucial elements like SAAS layout and typography. The framework was initially outside the reach of our study. After rigorous research, we were able to combine the same in accordance with our three key project feature goals: client needs, client comfort, and project objectives.

## Evaluation & Potential Project Scope

---

We are convinced that we have gathered the right resources as a team. Our first goals for the project have undergone a fluid metamorphosis, becoming something that surpasses what was originally envisaged. Our intensive building process provided us with the opportunity to learn how to manage human, time, and software resources more effectively. In terms of our product features, it is apparent that the app has gone through numerous rounds of testing and evaluation. One of the finest choices we looked for was to outfit our project with open-source technologies like JIRA and GitHub. We were able to manage and keep tabs on the jobs we had to finish, as well as those that were in progress and completed. Additionally, it made managing human resources simpler and helped us determine who to assign where. We can confidently state that each team member became more well-versed with the agile approach and principles.

While this is the case, it is only right to acknowledge that there is so much potential or scope for improvement of the app. There may be additional features that would make our app more attractive to market players.

Location-based services, video conferencing with medical experts, or even storing rich media reports with the help of better proprietary resources such as a larger database may give it that very same competitive edge. The expert access aspect is especially important. It is possible to provide users with better access to expert medical care through the app. Think about a bone specialist or a heart specialist. Since such experts are highly sought after, the public doesn't easily have access to them since they must undergo numerous processes to even be considered for an appointment.

Clearly, there are disparities in access to expert medical care. Physical consultations would be extremely time-consuming and exhausting for both patients and doctors when there is a high patient-to-doctor ratio. This issue can, however, be easily addressed using our app. It is especially relevant to pioneers. Similar results could be achieved by implementing a comprehensive video consulting system. Chatbots can also be developed using AI and ML to supplement expert support.

Medical literacy is another crucial aspect of integrating rich media records. Medical literacy or the degree to which the user can comprehend detailed medical records is essential when retrieving detailed records of the same. A lot of this can also be explained by AI and machine learning. Improved readability of discharge instructions offers health organisations an opportunity to support patients. AI-powered bots will enable them to take

control of their health outcomes as they are primary navigators of the health records system.

Having comprehensive features embedded in the application makes it possible for patients to get medical advice despite regional limitations. Additionally, it would potentially allow patients to get critical care throughout scenarios with physical limitations such as the pandemic. This allows patients to get access to medical care without placing themselves or medical personnel at risk of infection. Patients can receive prescriptions that are delivered right to their doorsteps in addition to consultations with doctors and specialists.

Telehealth not only makes healthcare more available, but it also provides high-quality, reasonably-priced primary healthcare to everyone, everywhere, and relieves pressure on healthcare systems by cutting down on patient wait times. Future developments suggest that the telemedicine software market will spread rapidly and become increasingly integrated with publicly funded healthcare.

# **Conclusion/Summary**

---

## **Overview of Project Outcomes**

Reflecting upon our project aims and goals that were proposed back in our first proposal, we set out to create an appointment booking system where elderlies take control of their appointment schedules. We also proposed an SOS feature where elderlies are able to make an emergency appointment in the event they need it. In addition to these functionalities, we have also considered a simple Graphic User Interface (GUI) for less cluttering and warmer colours to reduce the strains on the eyes, especially for the elderlies. Lastly, we have also considered extra features such as the option to change the app's language and font size to cater to the multi-racial society in our country. However, due to time constraints and the priority of the other features that we had set out to accomplish, this was pushed through a few sprints of backlog. In the near future, the team could consider working on implementing these features. Based on the user interviews done in the prototyping and iteration section, the team believes that the business goals "Lower the chance of the healthcare system being overwhelmed in situations where a high volume of elderlies require healthcare assistance at the same time" and usability goals "Helping elderlies overcome the stigma of using technology and feel comfortable with using the app to attend to their healthcare needs" proposed in the software proposal has been met to some extent, if not entirely.

## **Team Achievements**

Through this whole software development journey, the team has overcome many obstacles and challenges. These include the lack of technical knowledge in certain technologies, pandemics and therefore, social constraints holding our physical scrum sessions back, unmatched schedules of every team member and so on. In conclusion, given this opportunity to collaborate on this arduous yet fulfilling project, we went through countless research, interviews and user surveys, and learned various industrial software development methodologies and technological frameworks in order to produce a working model of the project. This in itself, could be considered an achievement for the project and the team and the team believes that these experiences are invaluable and will be able to aid us in our software development journey in the future.

# References

---

*Distinguished Examiner,*

- 1) **Here is a link to our video tutorial illustrating the features of our app.**  
**Please follow the link here: <https://youtu.be/wivZEzLJeZY>.**
- 2) **Find the link to our GitHub repo here:**  
**<https://github.com/Igneavour/Appointment-Management-App>**

1 *Color Designing for seniors.* (n.d.).

<https://eldertech.org/color-in-designing-technology-for-seniors/>.

2 *Elderly Stats Singapore.* (n.d.).

<https://www.statista.com/statistics/1113131/singapore-elderly-single-households-number/>.

3. *Bootstrap Documentation and Reference.* (n.d.).<https://getbootstrap.com/>

4. *eHealth Applications to Support Independent Living of Older Persons: Scoping Review of Costs and Benefits Identified in Economic Evaluations.* (n.d.).

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7988395/>

5. *E-health app guidelines developed to improve elderly care.* (n.d.).

<https://cordis.europa.eu/article/id/118486-ehealth-app-guidelines-developed-to-improve-elderly-care>

6. *Taking Mobile Tech to Older Adults.* (n.d.).

<https://www.singaporetech.edu.sg/digitalnewsroom/taking-mobile-tech-to-older-adults/>

7. *Aged Care App Development.* (n.d.).

<https://tateeda.com/blog/how-to-develop-an-elder-care-mobile-app-the-ultimate-guide>

8. *Encouraging health app use with seniors.* (n.d.).

<https://healthmanagement.org/c/hospital/issuearticle/encouraging-health-app-use-with-seniors>

*9. My SQL workbench and reference documentation. (n.d.-b).*

*<https://www.mysql.com/products/workbench/>*