Virtual Consultant: System Requirement Specifications

Aditya Chandrasekhar (U1923951A) : Project Manager / Lead Developer
Aratrika Pal (U1922069F) : Backend Developer
Shruthi Srinivas (U1923611G) : Backend Developer / Release Engineer
Chong Zhe Ming (U1920757K) : Frontend Developer
Khush Kothari (U1922279J) : Frontend Developer
Kushal Sai Gunturi (U1923232F) : QA Manager
Yi Jia Xin, Joceline (U1920057J) : QA Engineer
Lim Yun Han Darren: (U1921275J) : Frontend Developer

Team Name: 404Found
School of Computer Science and Engineering, Nanyang Technological University

Submitted to—
Yu Han
School of Computer Science and Engineering, Nanyang Technological University

2. Problem Statement

Teleconsultation has transformed the landscape of the world in the way things are done on a global scale. It provides opportunities to make healthcare more convenient, better coordinated, and closer to home. Patients can have consultations with doctors, nurses and allied health professionals in the comfort of their homes or at a location where privacy is assured. These virtual appointments enable patients to receive ongoing care where in-person clinic consultations are not necessary or possible.

The COVID-19 pandemic has dramatically changed how outpatient care is delivered in health care practices [1]. To decrease the risk of transmitting the virus to either patients or health care workers within their practice, providers are deferring elective and preventive visits, such as annual physicals. Our product has the potential to reduce the unnecessary visits by patients to hospitals by virtually connecting patients with available doctors.

People who have limited access to healthcare will also make use of our application to seek help from the doctors, allowing the professionals to access the medical condition of the patient and advise them accordingly.

Presently, there exists no such application in Singapore that allows both patients and doctors to have direct interactions with an intermediary organisation that acts like a middleman which may incur additional processing time to link up patients and doctors.

Similar applications existed in the form of TeleConsult by NUH [2] and Teleconsult by RafflesMedicalGroup, where users can download the application for virtual consultations to seek medical assistance. However, there are limitations such as:

- Virtual Consultation is limited to residents of Singapore
- Only the doctors attached to these hospitals are available on their respective apps, which may cause longer waiting times for patient's to get a consultation
- Lack of a feed for doctors to create posts and share healthcare tips with patients

We believe that our idea could be developed further and better integrated to bring greater convenience to the users by allowing them to seek virtual medical assistance and gain healthcare insights via our application.

3. Overview

3.1 Background

With the rise of global pandemic Covid 19 cases, people are seeking to reduce the number of physical interactions and move to virtual interactions whenever possible, in the interest of health and safety. Hence, virtual platforms have become the preferred option. The growth of healthcare services is also a beneficiary for this phenomenon. This convenience for users, in turn, brings about an easy and convenient way for people to seek medical assistance at the comfort of their own home.

Moreover, according to a new report from the World Bank and WHO [3], at least half of the world's population cannot obtain access to essential health services. Our application aims to solve this urgent problem. With our application, those who do not have access to healthcare will be able to seek medical consultations with professional doctors.

Virtual Consultant is envisioned with this in mind, to integrate and provide a convenient online platform where patients can seek consultation from doctors without physical interactions. Virtual Consultant aims to provide a safe space where doctors can be segregated from the patients such that they can reduce their exposure to the symptoms of their patients especially in the Covid-19 world we live in today.

Our product also provides a simplified version of a social media platform with the theme of medical health and wellbeing. It allows doctors to create posts to share experiences, provide information, and improve the patients' knowledge of diseases, symptoms and recovery. Patients can follow the doctors and engage with their posts through likes, with the final goal of improving their medical literacy.

3.2 Overall Description

Virtual Consultant is a progressive web application that runs on both desktop and mobile, where doctors and patients can connect through chat virtually. We have incorporated other useful features in the website, including the ability for doctors to view patient-history for context of illness, a feed for patients to get health-related posts from doctors they choose to follow, and a discussion forum for the posts. Virtual Consultant will provide healthcare-access to many users who have an internet connection, but non-optimal healthcare.

To focus on our objective of accessible healthcare, we will focus on building an extensible, accessible, maintainable and user friendly app that targets diverse groups of people around the world. The MERN stack, with a progressive web application is best for this task, permitting a rapid cross-platform deployment to reach out to as many users as possible. We place an emphasis on a good user-experience and are excited to build a powerful application with the ability to positively-impact lives.

4. Investigation and Analysis Methodology

4.1 System Investigation

In this project, we follow industry practices for the application for the best possible user-experience. When the user (patient or doctor) opens the application, he/she is required to log in with their account credentials. A secure verification process is used to ensure the authentication process is reliable. After login, the UI presented is different based on the user being a patient or doctor.

The patient may click on the “Consult” button to start a chat with the doctor. Patients are asked to key in details of their medical history during registration. Doctors can then view profiles of patients to better understand their symptoms and make an accurate diagnosis. Patients can also edit the details of their medical history to keep their profiles updated. The records are stored on MongoDB, a NoSQL database provider. The chats are encrypted on AWS to maintain privacy.

Doctors have a “View Patients” button where they can connect with patients. They are also allowed to post health related content through a “Make Post” button to create awareness about various diseases such as Covid-19, the symptoms to look out for, precautions to take to avoid being affected by it, home remedies to follow and so on. Users can follow these doctors and view their posts on their feed to stay updated on diseases such as Covid-19 etc. It is similar to a social media platform but is strictly limited to medical and health related matters. The like feature on the posts allow users to engage with the doctors and become aware about the prevention of different diseases.

4.2 Analysis Methodology

4.2.1 Feasibility Studies and Requirements Elicitation

We will be employing extensive research, and conducting interviews and surveys to ensure that Virtual Consultant is able to target its core-demographic’s needs. The Quality Assurance (QA) team will be looking at methods to certify doctors from around the globe. Ensuring that doctors have the necessary qualifications is paramount for our application, since unsuitable advice may threaten the patient’s life. We intend to approach organisations such as the WHO to conduct interviews, since they have a global presence across the world and have extensive understanding of procedures.

The Requirements elicitation will include contacting both patients and doctors. Patients will be interviewed about the sort of problems they often face, but do not visit the

doctors for. Their proficiency in medical-jargon will also be analysed in these interviews to set the appropriate simplicity for the application. The doctors will have their schedules observed, to ensure we tailor the app in such a way that it is simple to use, and they can fit the advice into their busy schedules.

Surveys will be passed out to the general populace in countries with high internet access but insufficient healthcare (e.g. rural India) to gauge the general medical habits of the patients, and how their quality of life can be improved.

4.2.2 System Analysis and Requirements Specification

Scope

1. Perform reliable authorization of account, provide different privileges based on login.
2. Develop a privacy-oriented, secure communication link between the back-end, database and front-end.
3. Provide communication channels between patients and doctors.
4. Provide a feed-platform for doctors to post and patients to view general medical-advice.

Limitations

1. The application will have a high-load of simultaneous users, which might make it more likely to crash.
2. No user can access the application (even the non-critical features that do not need authorization) without registration.
3. If the communication between the back-end and the database breaks down, the application will be inaccessible because all sensitive medical-records and feed-info is stored in the database.

4.2.3 Object Oriented Design with UML

Our application utilizes the industry-standard MVC (Model View Controller) design paradigm, which ensures OOP practices. Our UML Use-case diagram below shows the separation of different behaviors.

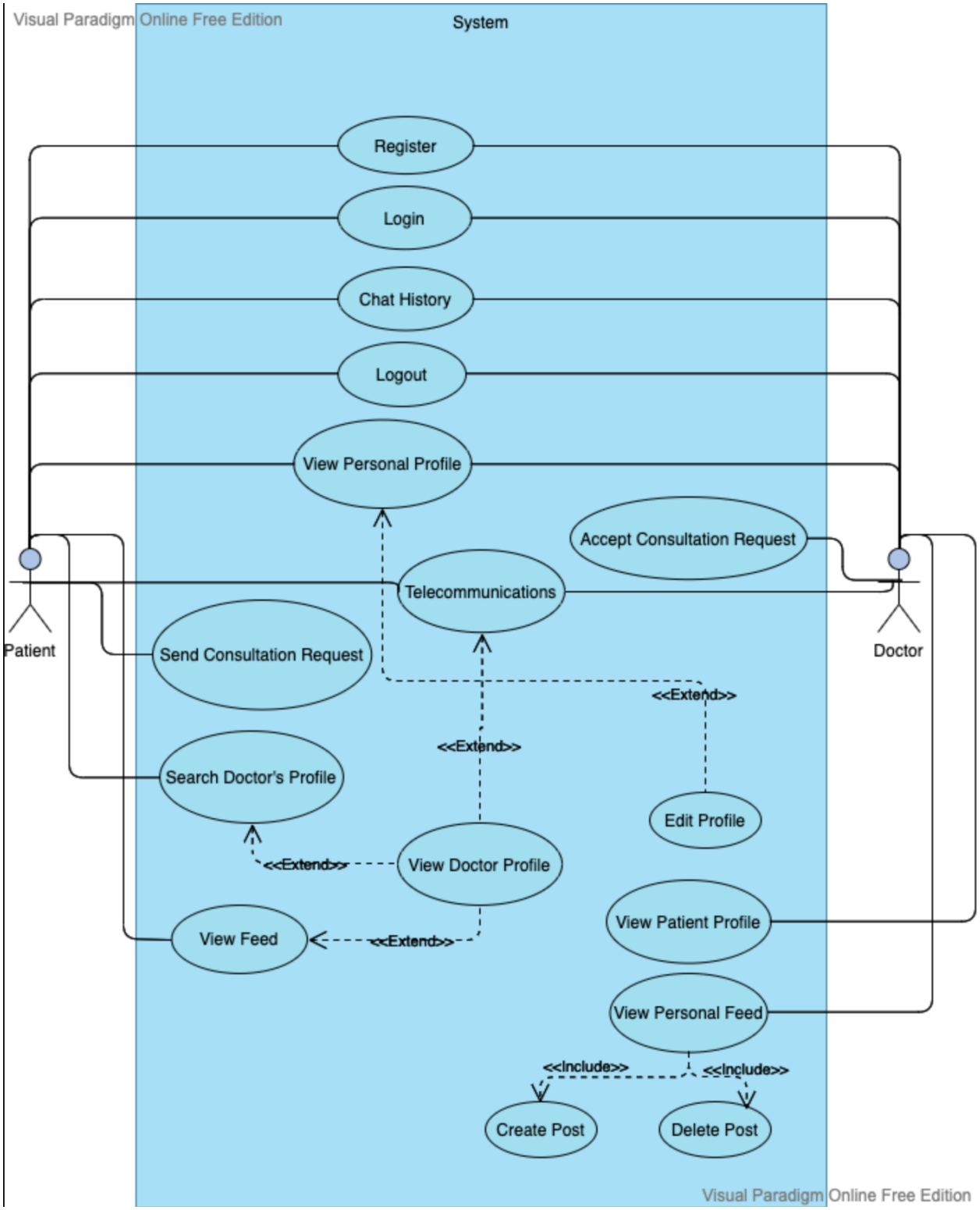
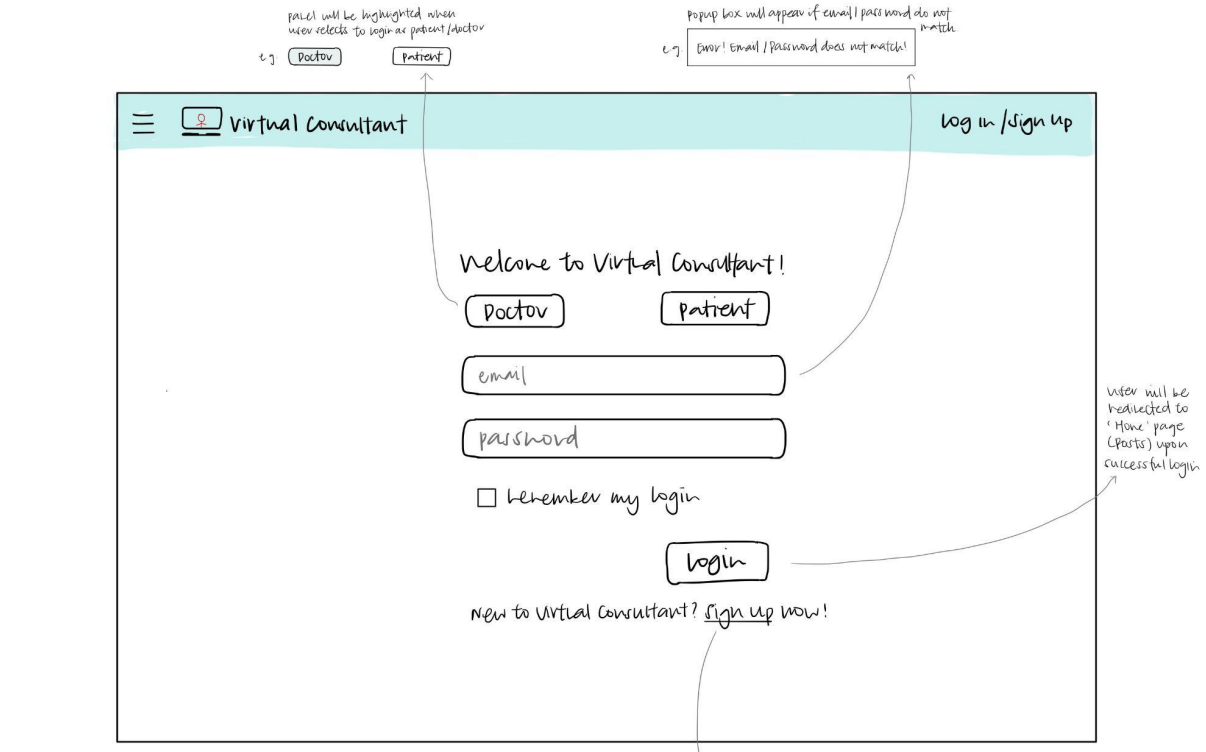
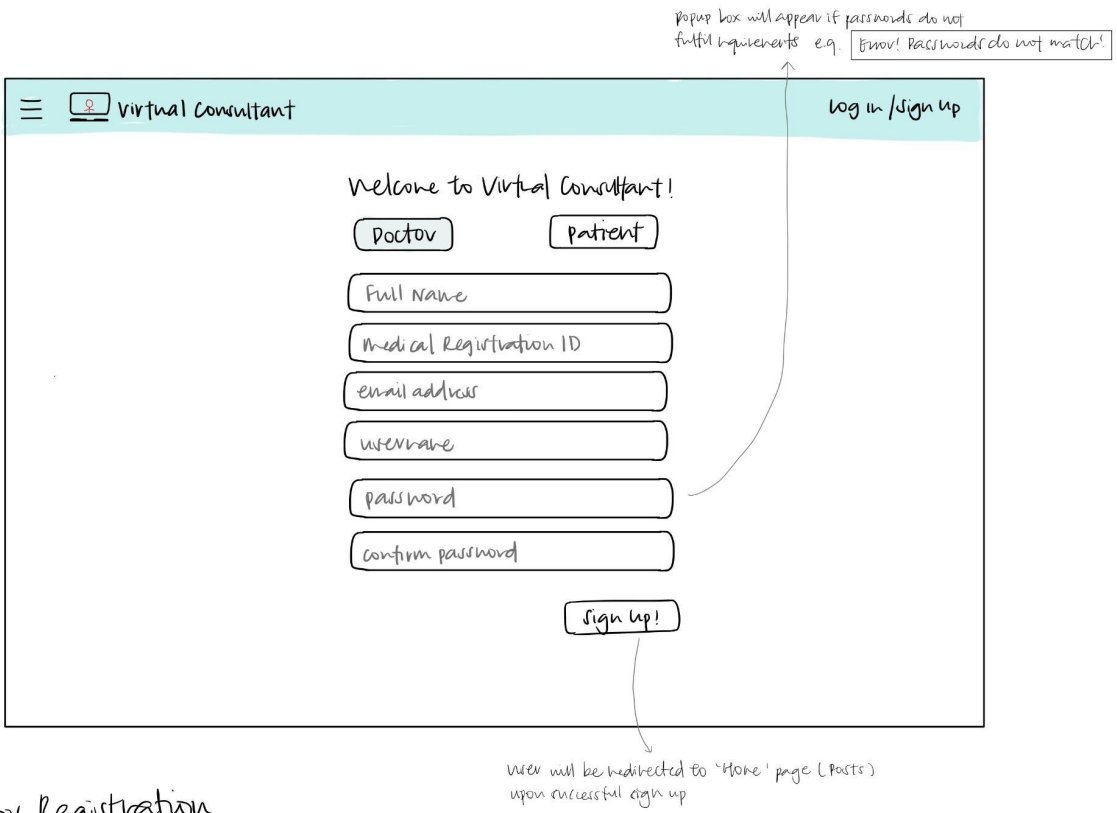


Figure 1: Use-Case Diagram

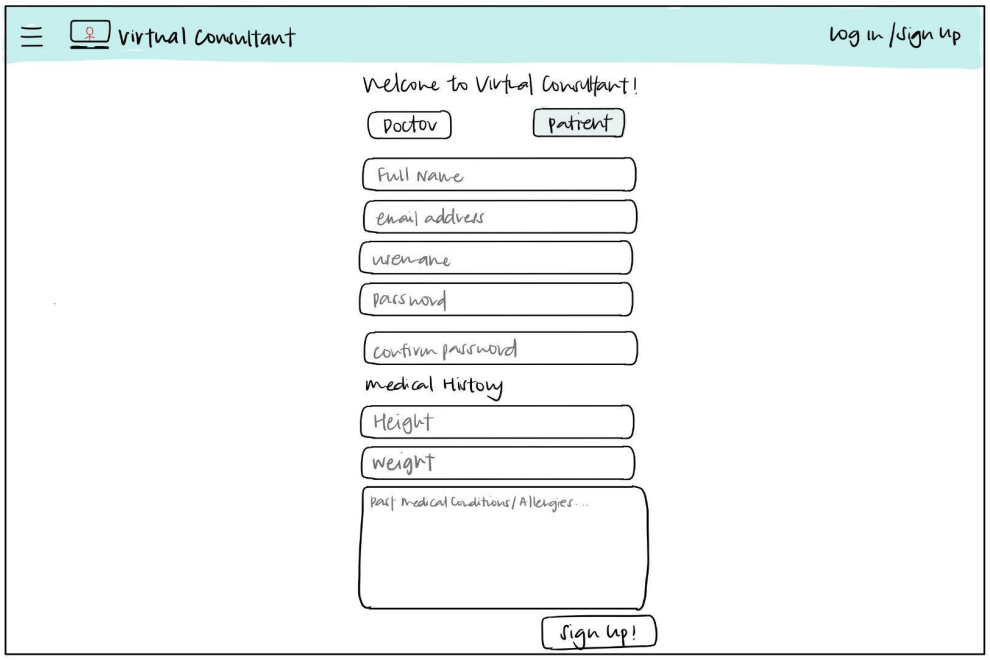
4.2.4 Prototyping



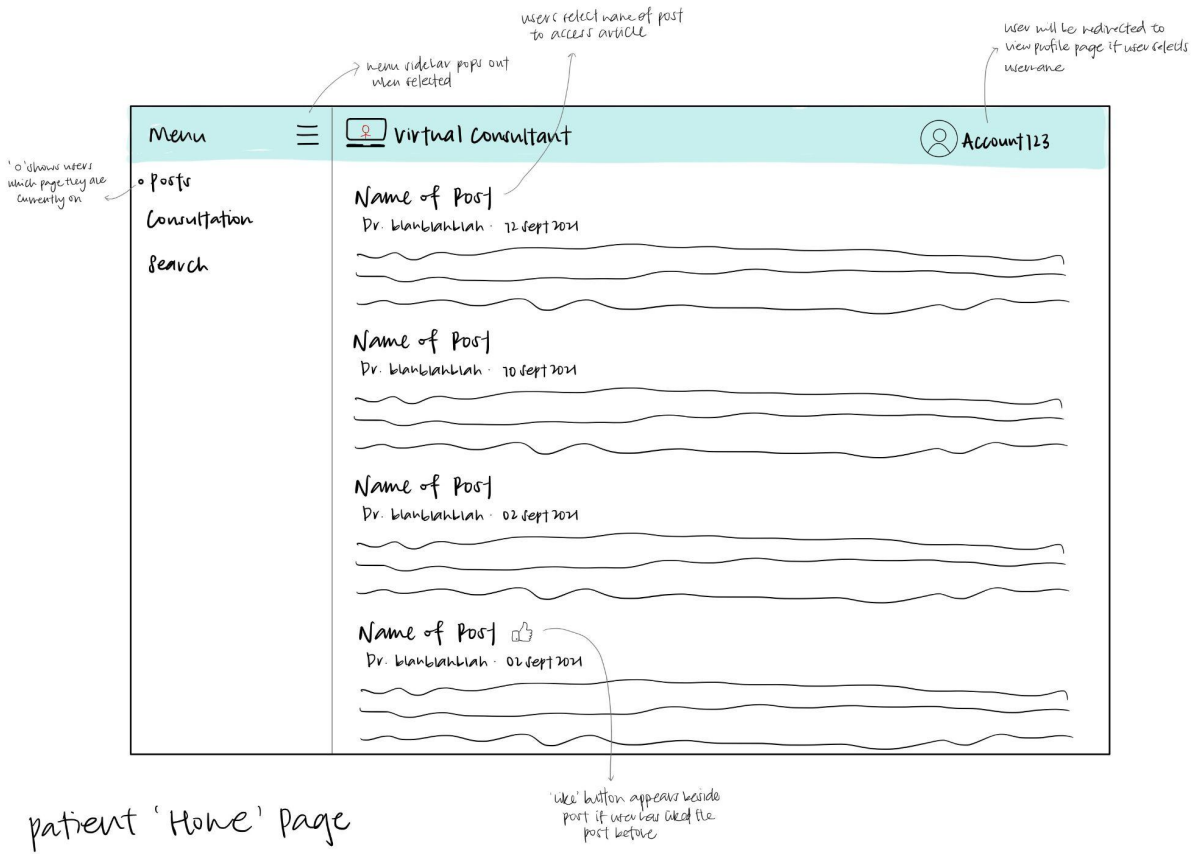
Login Page



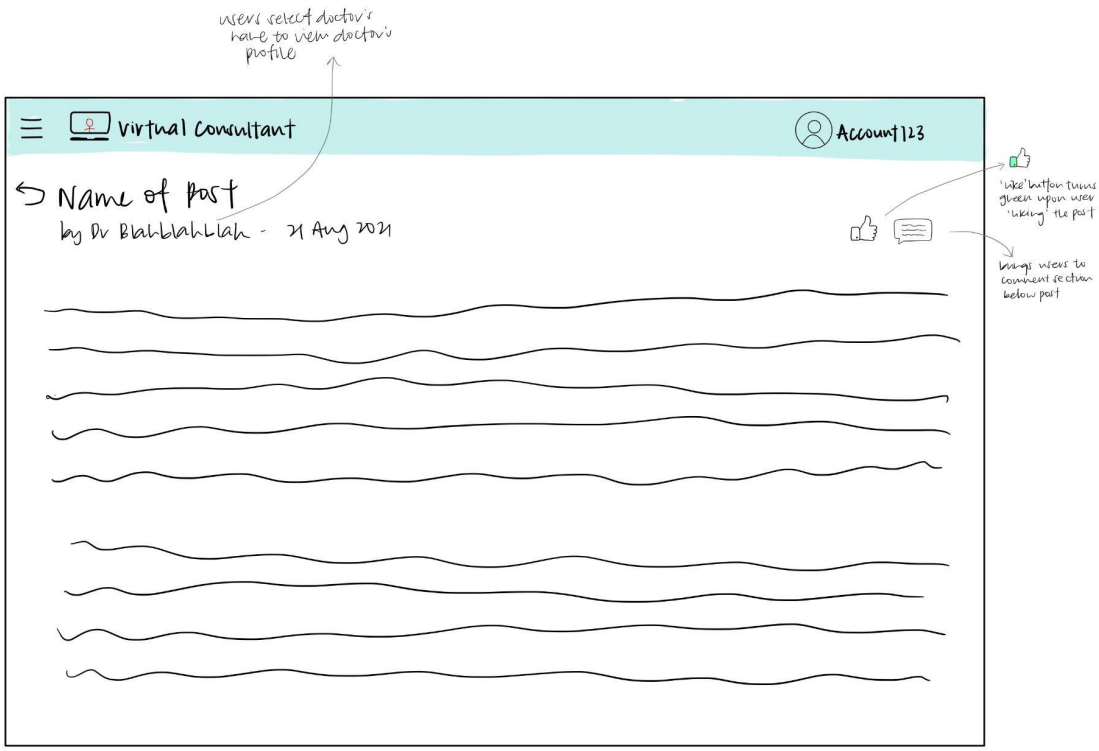
Doctor Registration



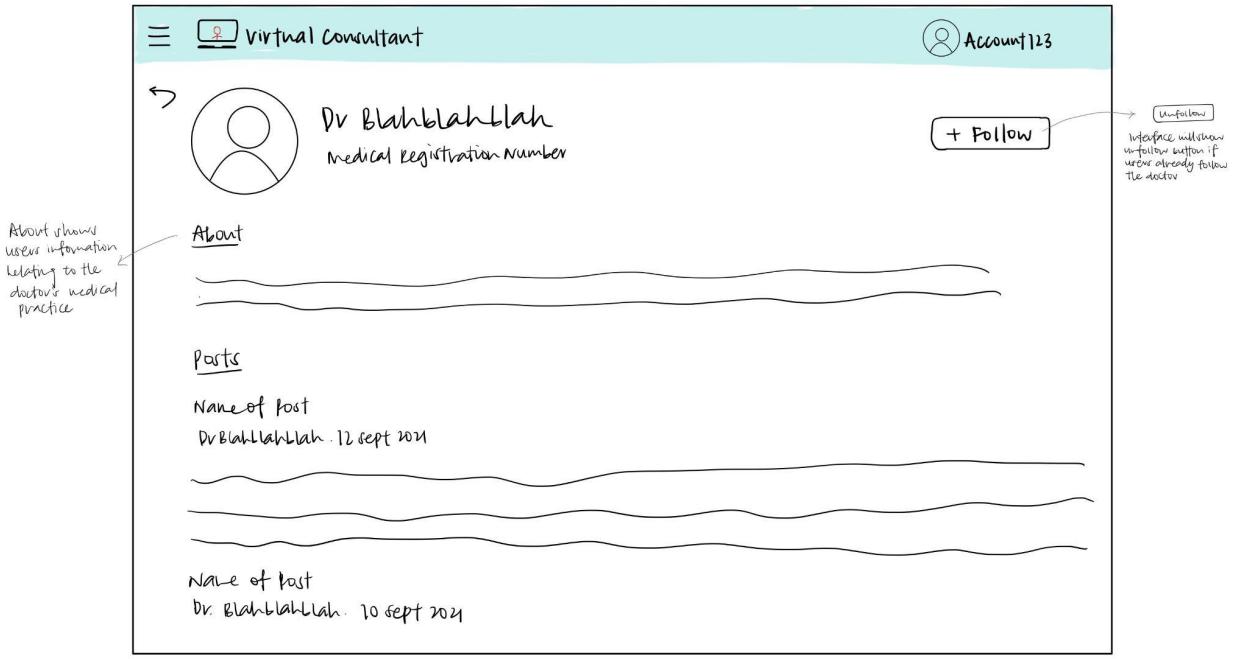
Patient Registration



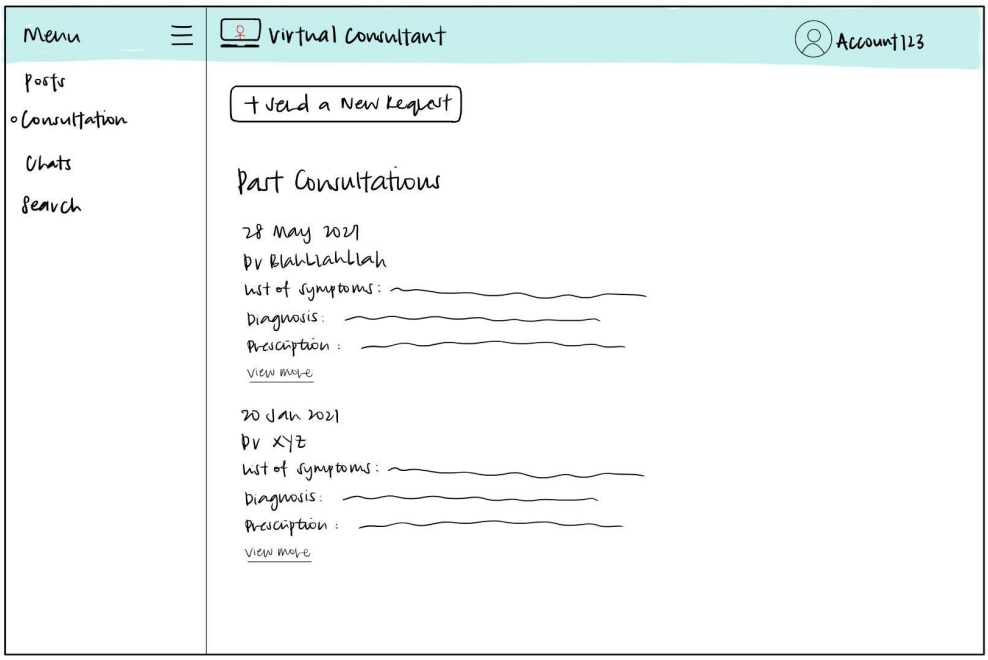
patient 'Home' Page



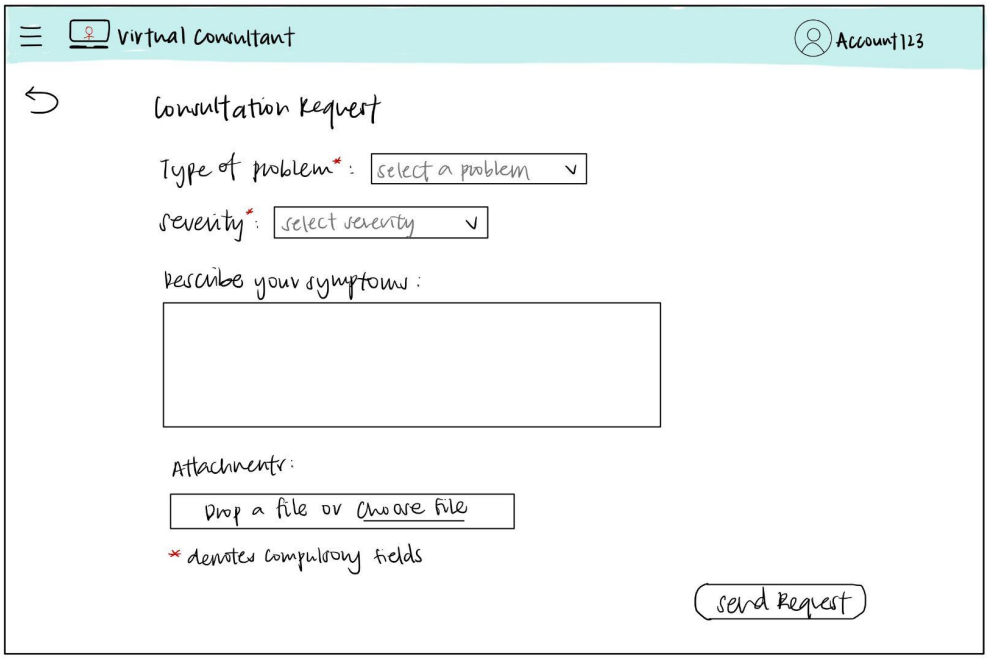
patient view post page



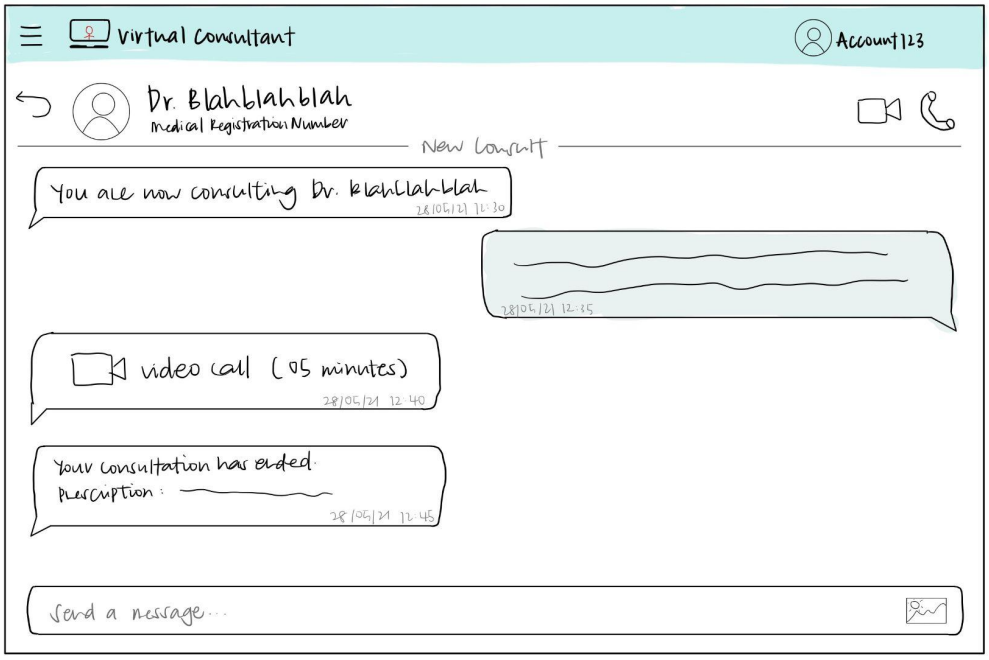
patient view Doctor Page



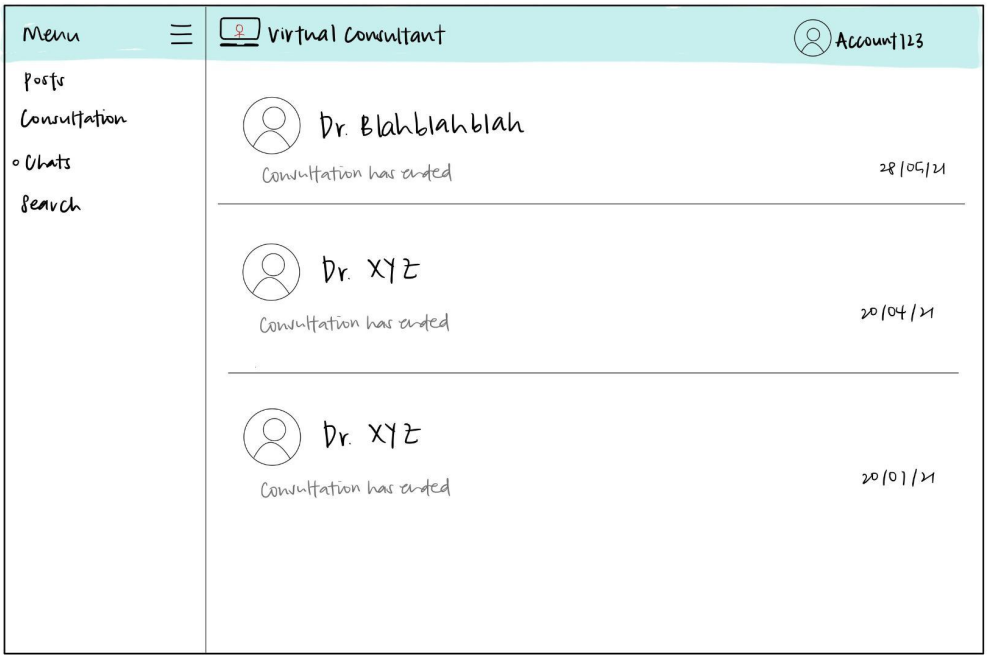
patient Consultation Page



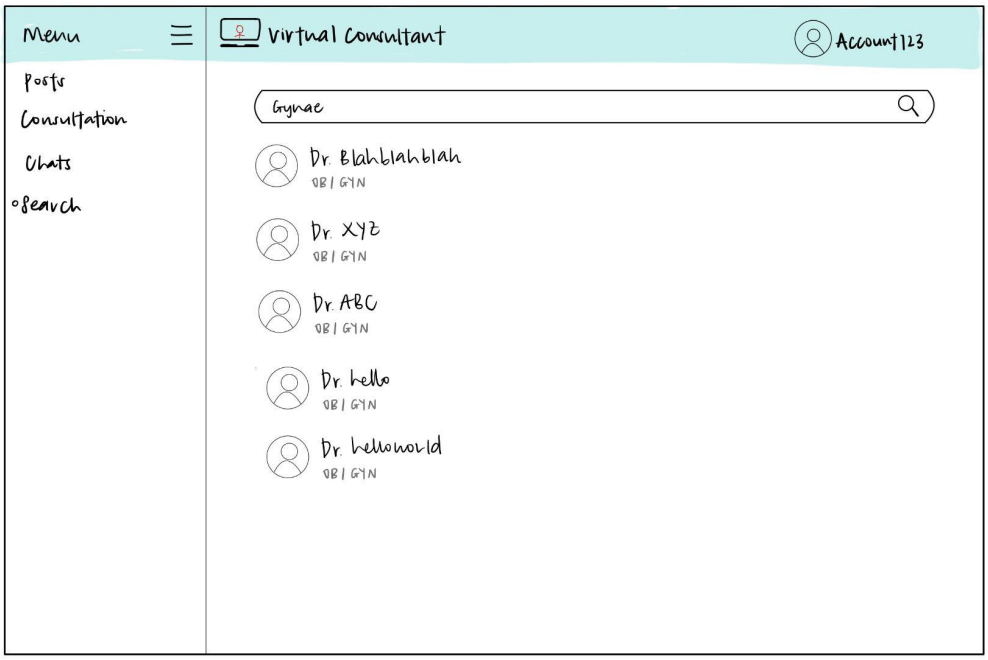
patient send new request page



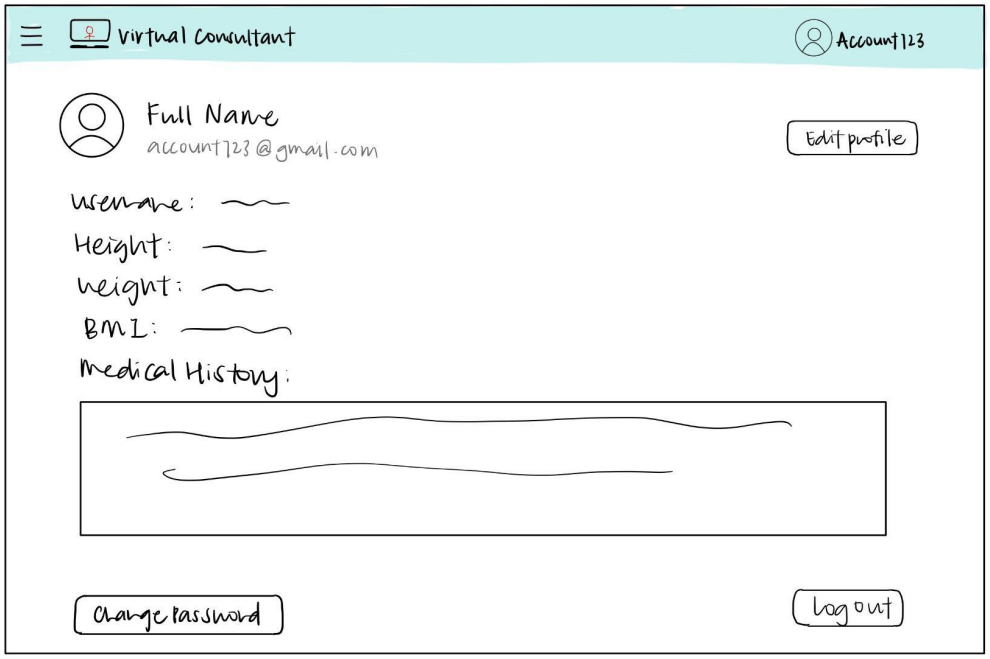
Patient Chat (Consult) Page



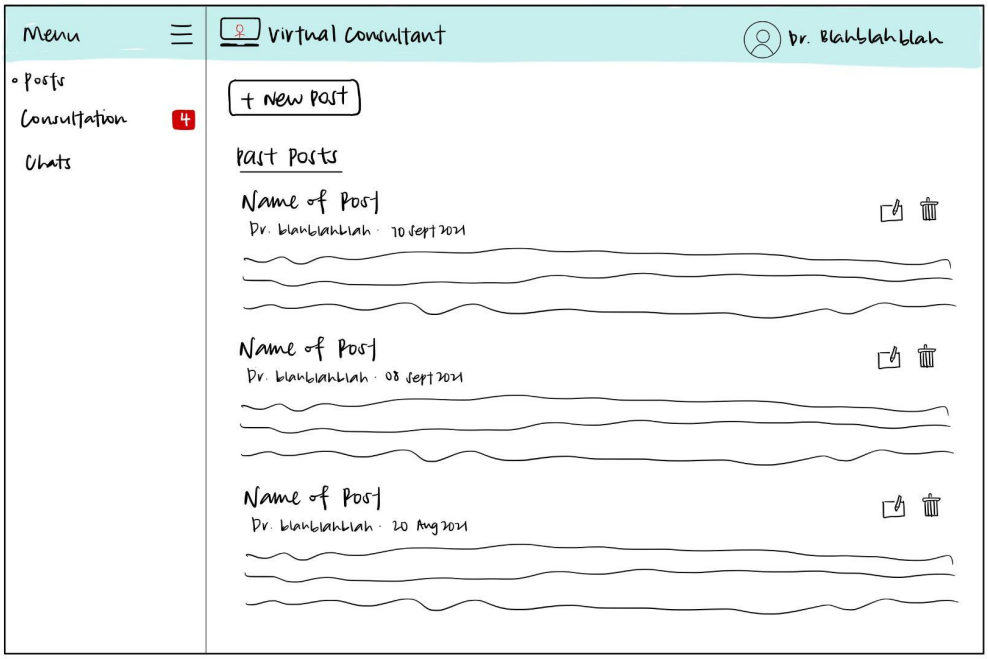
Patient Chat History Page



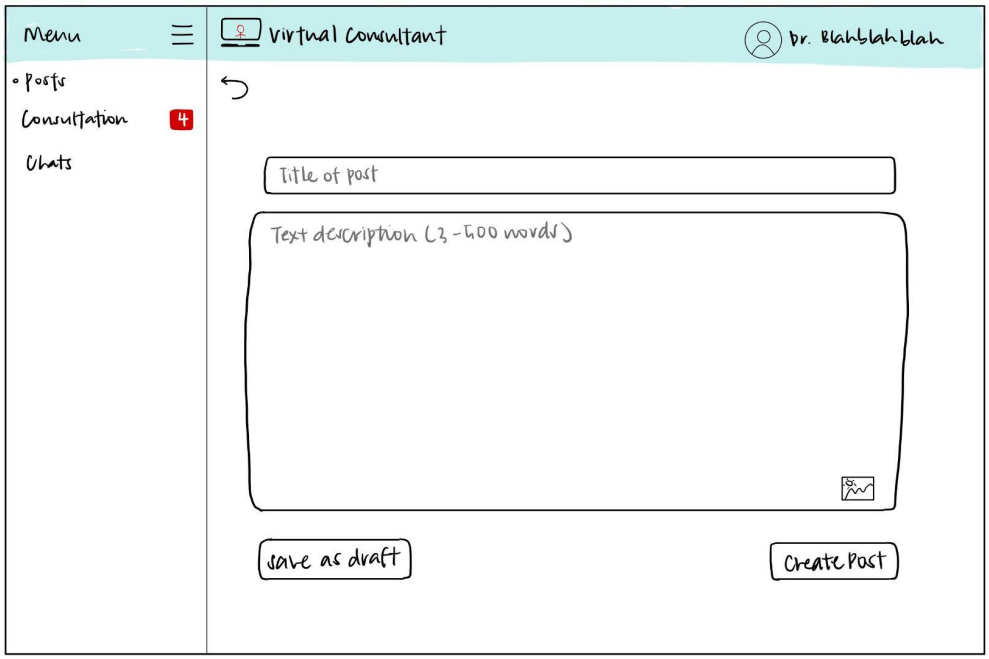
Patient Search Page



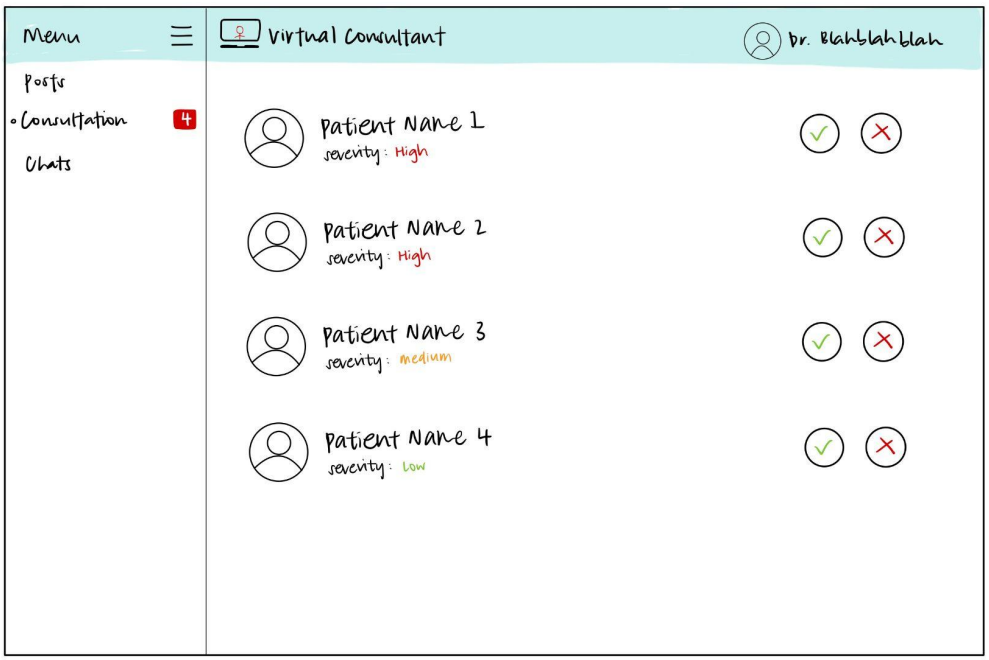
Patient Profile Page



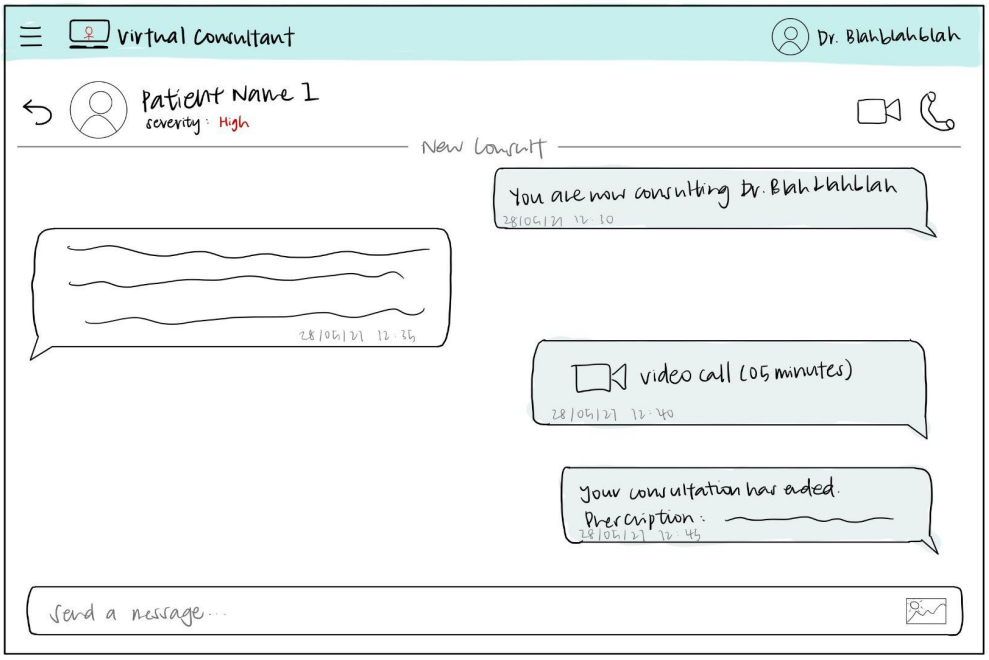
Doctor 'Home' (Posts) page



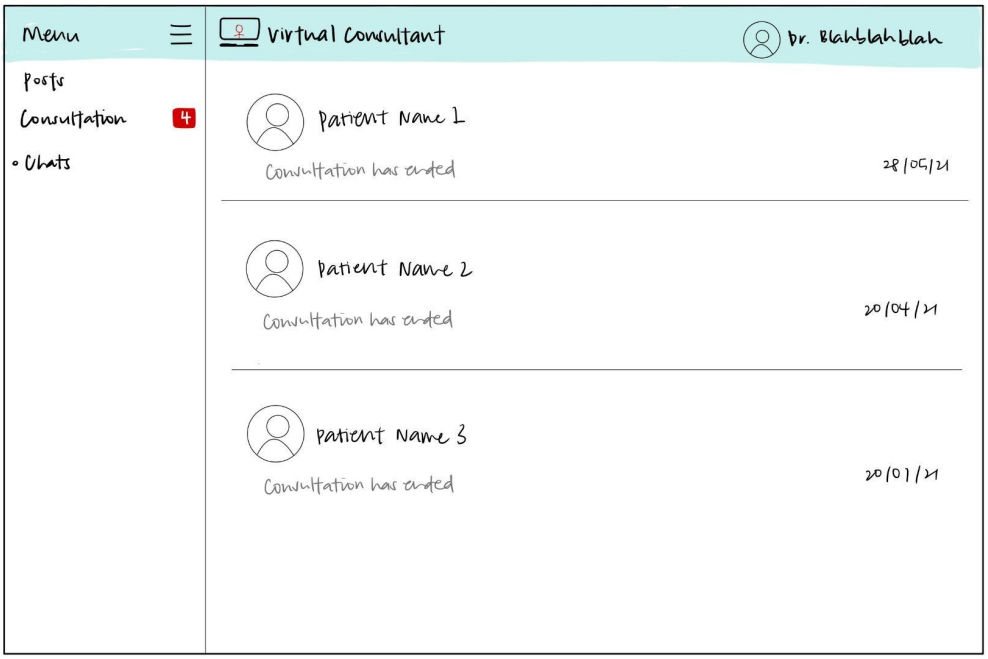
Doctor Create Post page



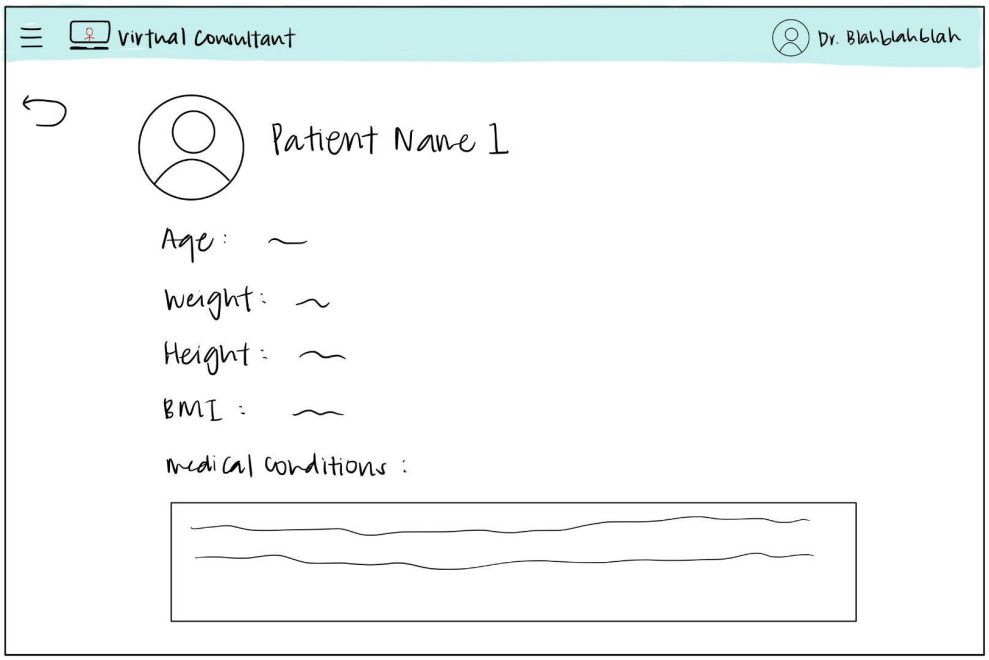
Doctor Consultation Request Page



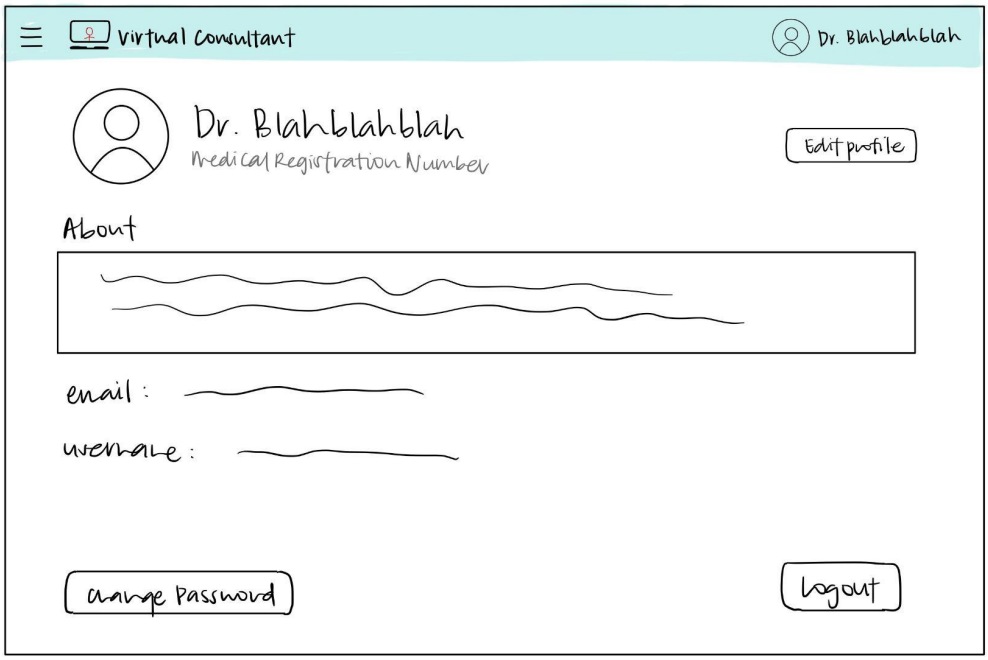
Doctor Chat Page



Doctor Chat History Page



Doctor 'View Patient Profile' Page



Doctor Profile Page

5. Constraints

5.1 Scalability

Virtual consultant is utilising Mongo Database and Node.js which are scalable. As a NoSQL database, MongoDB is scalable as its data is not coupled relationally. Data is stored as JSON-like documents which are self-contained. This allows those documents to be easily distributed across multiple nodes through horizontal scaling.

Node. js has a built in module that enables the creation of child processes that run simultaneously and share the same server port. Each child has its own event loop, memory, and V8 instance. The child processes use Inter-process communication to communicate with the parent Node.

5.2 Data and Function Mapping

All data used in the application such as user authentication, are stored in Mongo Database, a fully cloud managed NoSQL database. All information can be added remotely from the cloud system. Furthermore, once a user is logged in, all permissible data can be readily accessed from any device. A source code change is required if there is a change in the data model such as addition/deletion/updating of new data properties or the way data is displayed in the front-end.

Virtual Consultant does not run on a mainframe system. Instead, the backend is deployed on the cloud, with a cluster of low-cost servers, operated by a third-party vendor, Mongo Database, spread out globally across different data centres for data redundancy. Any new function added to the cloud system is deployed across all users remotely, whereas any change in a function pertaining to the business logic or front-end requires a change and recompilation of the source-code of the main application.

5.3 Proprietary Hardware and Software

Virtual Consultant is a progressive web application that runs on both desktop and mobile, where doctors and patients can connect through chat virtually. Since it’s not an application built for Android or iOS operating systems, it is highly accessible by simply using any web surfing application (Google Chrome, Safari, Firefox, Microsoft Edge).

The device is required to have internet connectivity, either Wi-Fi and/or mobile data (4G broadband cellular connectivity or later) and it does require a front camera.

5.4 Batch Updates vs Real-Time Updates

Virtual Consultant requires registration of all users for authentication purposes. All registered user data is synchronized via Mongo database system on a close to real-time basis. Other data displayed and updated in the application in real-time as well with the help of event listeners. Regardless of any manual backups, the system is configured to perform a daily backup of the database, assigned to a specific hour each day. The data transferred to our Mongo Cloud Storage bucket will be time stamped in ISO 8601 standard with the specific naming convention via a scheduled job.

5.5 Project Schedule

There is a three-month timeframe to implement a production system for Virtual Consultant. From August 2021 to November 2021.

Firstly, project requirements are solicited along with a clear definition of the product scope. Secondly, to further iron out the requirements, a use-case model and description is generated in the design phase. Thirdly, all development activities are started only after the completion of the first two phases by the development team. Finally, the release team engages in publishing a release plan, followed by the QA team to perform necessary testing.

6. Operational Requirements

6.1 System User support

Users must have a ‘Support’ section in the application. The support section must include FAQ (Frequently Asked Questions) to address common user concerns. The FAQ section must be duly updated in case of any new recurring question or issues. Moreover, a community support forum should also be created for the users to be easily able to find answers to specific and recent issues, which might not be present in the FAQ. The questions in this forum should also be then answered by the official support team at the earliest.

6.2 Application Services and Technical Support

Users can report any bugs they come across the application or recommend enhancements through feedback. The users can use the above community forum, or the report bugs feature in the support section to report new bugs.
In order to ensure high availability and uptime, Network Administrator and Database Administrator support is required. Database redundancy and network best practices must be put in place to allow adequate system performance and efficiency. Further, some users can be selected as beta testers for the new features, which will allow them to find unidentified bugs or issues before major releases.

6.3 Security Features

Since the application contains a social media feature, system security must be implemented and maintained. In the case of the video call, it must be end-to-end encrypted. With end-to-end encryption, there will be no unauthorised third party who can access the video call.

A passcode protection must also be implemented to enable the user to unlock their account with a passcode. The passcode will have a password strength to ensure that the password will not be easily decrypted.

6.4 Administration Features

There must be authorized system administrator(s) who have the right to block any users from using the application, should the need arise. Blocked users will require separate assistance to be able to use the web application again.

Furthermore, measures to prevent scraping of the application data must be put in place to prevent bots or crawlers from misusing the system. The administrators will also monitor the presence of any fake accounts of doctors. All doctors must be verified with their Medical Registration ID provided by the Singapore Medical Council (SMC) before being allowed to be a Doctor on Virtual Consultant.

6.5 System Hardware and Data

A computer operations centre must be set up. It must service and maintain the system hardware and set up a failover for unforeseen or scheduled primary system down time. It must also carry out scheduled data back up to avoid data loss and release system patches should the need be. The hardware must be serviced every quarter and any quality checks must be performed to ensure quality standards of the equipment.

Disaster recovery protocols and drills must be in place in case of any such system wide or regional failures. The cloud provider must have adequate service level agreements to ensure proper uptime of the application. Moreover, complete procedures and guidelines must be followed for data storage in compliance with regional guidelines and data regulations.

6.6 Audit Trail

System audit trail must be an inherent part of Virtual Consultant. All transactions in the system will capture the action that was taken, the time and by whom. The audit trail includes both the user access and modification as well as administration access. Proper Identity Access (IAM) Policies must be in place to ensure complete track of the access and modification history to check any unintended or unauthorised access of the application. Regular monthly audit log checks must be taken, and the audit trail must be verified on a fortnightly basis.

7. Functional Requirements

7.1. System Functionality to be performed (Doctor and Patient)

7.1.1. Register

- 7.1.1.1. The system must provide a separate register entry for Doctors and Patients
  - 7.1.1.1.1. The system must allow the Doctor to register to the Doctor domain
  - 7.1.1.1.2. The system must allow the Patient to register to the Patient domain
- 7.1.1.2. The system must allow the Doctor to fill in his/her name, email address, username, Medical Registration ID, password, confirm password
  - 7.1.1.2.1 The system must validate that all the text fields are filled up
  - 7.1.1.2.2. The system must validate that email address provided is valid and has not already been registered
  - 7.1.1.2.3 The system must validate if the password contains at least 6 characters
- 7.1.1.3 The system must allow the Patient to fill in his/her name, email address, username, weight, height, medical history, password, confirm password
  - 7.1.1.3.1 The system must validate that all the text fields are filled up
  - 7.1.1.3.2. The system must validate that email address provided is valid and has not already been registered

7.1.1.3.3 The system must validate if the password contains at least 6 characters

7.1.1.3.4. The system must validate if the password matches with the confirm password

7.1.1.4 The system must redirect the user to the home tab upon successful account creation

7.1.2. Login

7.1.2.1. The system must provide a separate login entry for Doctors and Patients

7.1.2.1.1. The system must allow the Doctor to login to the Doctor domain

7.1.2.1.2. The system must allow the Patient to login to the Patient domain

7.1.2.2. The system must be able to validate the email and password text fields

7.1.2.2.1. The system must be able to validate that the email and passwords are filled

7.1.2.2.2. The system must display corresponding error message when any of the fields are not filled

7.1.2.3. The system must validate the user's credentials with the database before letting them use their respective domains

7.1.2.3.1. The system must display an error message when the email is not registered

7.1.2.3.2. The system must display an error message when the password does not match with the value in the database

7.1.2.4. The system must redirect the user to the home tab after successful login

7.1.3. Chat History

7.1.3.1. The system must allow the users to view their chat history

7.1.3.2. The system must display a list of all the conversations that the user has had

7.1.3.3. The user must select any one of the conversations at one instance

7.1.3.4. The system must display the entire history of the conversation including text, audio call duration and video call duration with respective time stamps

7.1.3.5. The user must be able to view the profile of the person who's conversation they're viewing

7.1.4. Telecommunication

7.1.4.1. Once a Doctor accepts a Consultation request, the system must display a chat interface for both the Doctor and the Patient to communicate with each other

7.1.4.2. The users must be able to send and receive messages to each other

7.1.4.3. The users must be able to audio conference with each other
7.1.4.4. The users must be able to video conference with each other
7.1.4.5. The users must be able to send attachments to each other like images, videos, audios

7.1.5. View Personal Profile
7.1.5.1. The user must be able to view their own profile
7.1.5.2. The system must display all the users information
7.1.5.3. The user must be able to edit his/her personal profile
7.1.5.3.1. The user must be able to change his/her username
7.1.5.3.2. The user must be able to change his/her password
7.1.5.3.3. The user must be able to change their email ID
7.1.5.3.4 Patients must be able to change their medical history information
7.1.5.4. The system must save the updates made by the user and modifies the value in the database

7.1.6. Logout
7.1.6.1. The system must allow the user to logout of his/her account
7.1.6.2. The system must ask the user to confirm whether he/she wants to logout
7.1.6.3. When the user clicks yes, the system must redirect the user back to the login page
7.1.6.4. When the user clicks no, the user must remain on the same page

7.2. System Functionality to be performed (Patient)

7.2.1. View feed
7.2.1.1. The patient must be able to view a feed on his/her homepage upon logging into the application.
7.2.1.2. The system must display posts made by doctors followed by the patient, on the feed, ordered by recency.
7.2.1.3. The patient must be able to like the displayed posts.
7.2.1.4. The system must redirect the patient to the doctor’s personal profile if the patient clicks on the doctor’s username shown in their post.

7.2.2. Send Consultation Request
7.2.2.1. The patient must select the Type of Problem from a dropdown menu.
7.2.2.1.1. The Type of Problem can be ‘Cardiology’, ‘Gynaecology’ and so on.
7.2.2.2. The patient must select the Severity level from a dropdown menu.

7.2.2.2.1. The Severity level can be ‘High’, ‘Medium’ or ‘Low’.

7.2.2.3. The patient must add a text message if he/she deems it necessary.

7.2.2.4. The patient must add any attachment if he/she deems it necessary.

7.2.2.4.1. The attachment must be in jpg/png/pdf/mp4 format.

7.2.2.4.2. The patient can add a maximum of 5 attachments.

7.2.2.5. The system must send the consultation request to all the doctors who specialize in the Type of Problem selected by the patient.

7.2.3. Search Doctor’s profile

7.2.3.1. The patient must navigate to the search tab on the application.

7.2.3.2. The patient must enter the doctor’s name who they want to view.

7.2.3.3. The system must perform a fuzzy search based on the entered doctor’s name.

7.2.3.4. The system must display the closest matching doctor profiles based on the search results.

7.2.3.4.1. If no matching results are found, the system must inform the user that no such doctors were found.

7.2.3.5. The system must redirect to the doctor’s profile if the patient clicks on any displayed search result.

7.2.4. View Doctor’s profile

7.2.4.1. The system must redirect the patient to the doctor’s personal profile page.

7.2.4.2. The system must display the personal information of the Doctor.

7.2.4.2.1. The system must display the type of problem that the doctor specialises in.

7.2.4.3. The system must display all the posts made by the doctor, ordered by recency.

7.2.4.4. The system must allow the patient to view a post made by the doctor.

7.2.4.5. The system must allow the patient to like a post made by the doctor.

7.2.4.6. The system must allow the patient to follow or unfollow the doctor.

7.3. System Functionality to be performed (Doctor)

7.3.1. View feed(create post and delete post as subpoint)

7.3.1.1. The doctor must be able to view a feed on his/her homepage upon logging into the application.

7.3.1.2. The system must display all the posts made by the doctor, ordered by recency.

7.3.1.3. The doctor must be able to create a post.

7.3.1.3.1. The doctor must click on a button to create a new post.

7.3.1.3.2. The doctor must add a text description of the post.

7.3.1.3.2.1. The text description must be within 3-500 words.

7.3.1.3.3. The doctor must be able to add an attachment to the post if needed.

7.3.1.3.3.1. The attachment must be in jpg/png/pdf/mp4 format.

7.3.1.3.4. The system must create the new post and add it to the database.

7.3.1.3.5.The system must update the feed by including the newly created post.

7.3.1.4. The doctor must be able to delete a post.

7.3.1.4.1. The doctor must choose the post to be deleted.

7.3.1.4.2. The system must delete the post from the database.

7.3.1.4.3. The system must update the feed by removing the deleted post.

7.3.2. Accept consultation request

7.3.2.1. The doctor must receive a notification from the system about the patient's consultation request.

7.3.2.2. The doctor must accept or decline the request.

7.3.2.3. If the doctor accepts the request, the system must set up a chat room enabling chat, audio and video communication between the patient and the doctor.

7.3.2.4. If the doctor accepts the request, the system must display that the patient has received help and not allow any other doctors to accept it further.

7.3.2.5. If the doctor declines the request, the system must wait for one of the other doctors to accept the request.

7.3.2.6. If no doctor accepts the request within a certain timeout limit, the system must display a message to the patient stating that no doctors are available.

7.3.2.6.1. If the severity level is 'High', the system must set the timeout limit as 10 minutes.

7.3.2.6.2. If the severity level is 'Medium', the system must set the timeout limit as 20 minutes.

7.3.2.6.3. If the severity level is 'Low', the system must set the timeout limit as 30 minutes.

7.3.3. View Patient's Profile

7.3.3.1. The system must redirect the doctor to the patient's personal profile page.

7.3.3.2. The system must display the personal information of the patient.

7.3.3.3. The system must display the medical information of the patient.

7.3.3.3.1. The system must display the age of the patient.
7.3.3.3.2. The system must display the weight of the patient.
7.3.3.3.3. The system must display the height of the patient.
7.3.3.3.4. The system must display the Body Mass Index (BMI) of the patient.
7.3.3.3.5. The system must display the medical history of the patient.

8. Input Requirements

8.1. Medical Registration ID provided by Singapore Medical Council (SMC)

The Singapore Medical Council (SMC) is a statutory board under the Ministry of Health Singapore, which maintains the Register of Medical Practitioners in Singapore. SMC provides a unique Medical Registration ID to each doctor registered under them. While registering an account on Virtual Consultant, doctors must know their Medical Registration ID which was provided by SMC. This is to ensure that only registered and verified medical professionals are registered as doctors on our app. As currently the application caters to Singapore, anyone who is not registered as a doctor with the SMC will not be able to have a doctor account on Virtual Consultant. Our application does web scraping to gather a list of all the Medical Registration IDs provided by SMC and during doctor account registration, it validates that the Medical Registration ID provided as input is found in the list that was formed by scraping.

8.2. Patient Medical Information during Registration

When patients register an account on Virtual Consultant, they must provide the following medical information about themselves: age, height, weight and past medical history. The application validates the age, height and weight and also uses the height and weight information to calculate the patient's Body Mass Index (BMI). The past medical history information is a string value where all the patient's existing or historical medical conditions need to be mentioned separated by whitespace.
This information is stored on the application and is visible to doctors when they see the patient's profile. Often, medication prescribed can vary depending on the age, height or weight of the person, and patients might not be allowed to take a medicine because of their past medical history. Hence having the patient's medical information is extremely important for doctors to be able to prescribe the appropriate medication for any medical condition.

8.3. Patient Condition Information required for Consultation Request

When the patient sends out a Consultation Request through Virtual Consultant, they must add the following information about their condition: type of problem, severity level, an optional text message, and optional further attachments. The type of problem can be selected from a drop down menu and will have options such as ‘Cardiology’, ‘Paediatrics’, ‘Gynaecology’ etc. Based on the type of problem, our application will notify the doctors who specialise in that area and send the consultation request to them only. Patients can also select the severity level from a drop down menu, among three options: ‘High’, ‘Medium’ and ‘Low’. Optionally, they can add a text message to explain their medical condition to the doctor so that it can be sent in the request. Optionally, they can also add attachments in jpg, png or pdf format to send in the request. These attachments could be any images of their condition, previous reports or medical checkup certificates.

9. Process Requirements

The following are among the inherent requirements that Virtual Consultant must be able to handle.

9.1 Performance

The application must be able to handle concurrent use of the system on a 24x7 basis. It must also implement the functional requirements while sending, receiving, and displaying user messages to assist the overall user experience.

9.2 Data Integrity

Commit transactions that are completed and/or rollback unfinished or time-out transactions. For example, error messages displayed for transactions that failed due to time-out/ system failure etc. Successful results for successful transactions.

9.3 Data Validation

Data errors from the user’s end or backend database must be handled. There will be data validation and error-handling functionalities implemented in the online registration system.

9.4 Data Repository

The application will use the database to store the repository of users' personal information. The database will also be used for other administrative purposes as and when the need arises.

10. Output Requirements

10.1 Feed containing posts of doctors the user follows

The main screen of the application (once the user logs in) displays content from the doctors the user follows. The posts in each user's feed are ordered according to the time at which it was posted. The user can like the posts in their feed. Below the feed is a bar with 4 buttons- Home, Send Requests, Search Bar and Profile. The 'Home' button leads to the main screen of the app with the user's feed.

10.2 Consultation Requests

From the Home page with the user's feed, the user can click a button 'Send Requests' that leads to a page with a form to be filled out. Once the user fills it out, consultation requests are sent out to available doctors. Once the user gets assigned to a doctor, he/she is taken to a page where they can choose to communicate with the doctor via text, audio or video.

10.3 Doctors Profiles

From the Home page, the user can look up names of doctors in the 'Search Bar' to view their profiles. Once in the doctors profile, he/she can follow the doctors page and view the content posted by them.

10.4 Users Profile

From the Home page, the user can view their own profiles by clicking 'Profile' where they can view and edit their biodata. There is also a 'Logout' button for the user to logout of the application.

10.5 Create and Delete Posts (for doctors)

From the Home Page, the doctor can create posts by clicking the 'Create Post' button. The doctor can also delete their older posts by clicking the 'Delete Post' button next to the post.

11. Hardware Requirements

11.1 Network

Network connectivity is required at all times in order to keep the users feed updated and to send requests to doctors over the internet. Thus, the client's device must be connected to the internet before the application is launched.

11.2 Client Devices

The client can use any device to use the application as it is accessed via the internet browser and will adapt to the device it is being viewed on. To be able to access the internet, network connectivity of the device (Wi-fi and/or cellular data) should be functioning.

11.3 Production Support Systems

MongoDB server computers are required to store information on user accounts in the MongoDB database.

12. Software Requirements

12.1 Client Operating Systems

12.1.1 Google Chrome
Google Chrome is a cross-platform web browser developed by Google. It was built with free software components from Apple WebKit and Mozilla Firefox and supports Linux, macOS, iOS, Android and Windows systems.

12.1.2 Safari
Safari is a graphical web browser developed by Apple. It was built with open-source software such as WebKit. It is currently supported on macOS, iOS, and iPadOS

12.2 Client Application

The front-end of the application is built using React, a UI development framework. React is a free open-source library maintained by Facebook. React is mainly concerned with state management and rendering that state to the DOM.
React uses JavaScript or TypeScript as the programming language which compiles it into executable code which is rendered on the web.

12.3 Network System

All data is sent to our MongoDB backend system over the HTTPS protocol. Since some of the data sent over the web is sensitive, an encryption protocol such as WPA2/AES is used to prevent data leakage.

WPA2 is an encryption method used to secure the vast majority of Wi-Fi networks. A WPA2 network provides unique encryption keys for each wireless client that connects to it.

There is no network diagnostic tool built into the application that warns the users of potential network pitfalls.

12.3.1 TCP/IP

The Transmission Control Protocol is a communications standard that enables application programs and computing devices to exchange messages over a network. It is designed to send packets of information over networks. TCP organizes data so that it can be transmitted between a server and client application. It guarantees the integrity of data being communicated over a network.
Additionally, the Internet Protocol is the method for sending data from one device to another across the internet. Every device has an IP address that uniquely identifies it and enables it to communicate with and exchange data with other devices connected to the internet.

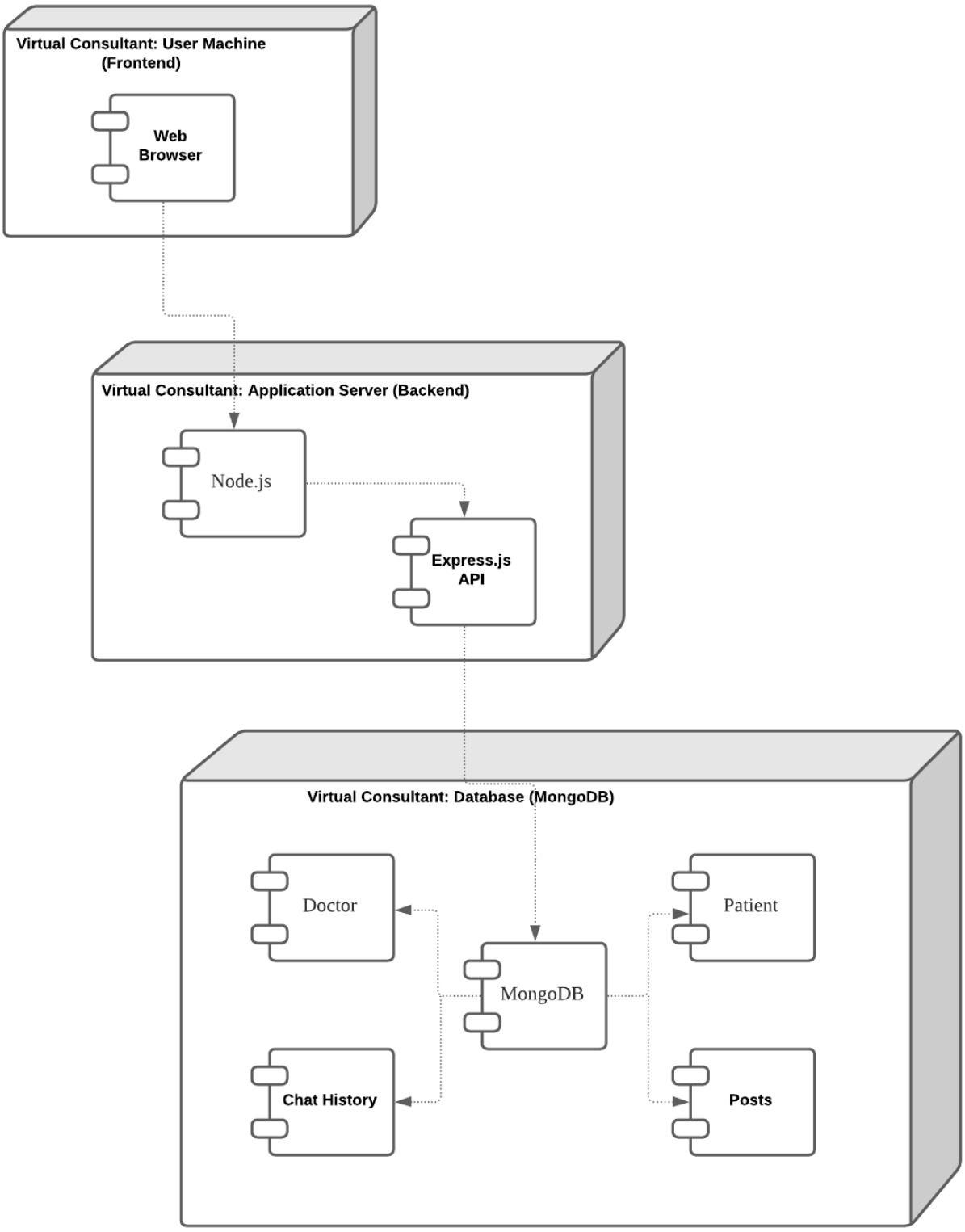
12.3.2 HTTPS

Hypertext transfer protocol secure is the secure version of HTTP, which is the protocol primarily used to send data between a web browser and a website. HTTPS is encrypted in order to increase security of data transfer. This is particularly important when users transmit sensitive data such as passwords within our application. HTTPS thus prevents websites from having their information broadcast in a way that is easily viewed by anyone snooping on the network.

12.4 Licenses

Application Development Tools	Github	MIT
	Figma	Organization License
	Visual Studio Code	Microsoft
Backend and Database system	Mongo Atlas	Server side Public License

13. Deployment Requirements



14. References

[1] *What impact has covid-19 had on outpatient visits?* Commonwealth Fund. https://www.commonwealthfund.org/publications/2020/apr/impact-covid-19-outpatient-visits

[2] *TeleConsult - NUH: National University Hospital*. For Patients & Visitors. https://www.nuh.com.sg/patients-visitors/Pages/Teleconsult.aspx

[3] World Health Organization. *World bank And WHO: Half the world lacks access to essential health services, 100 million still pushed into extreme poverty because of health expenses*. World Health Organization. Retrieved from: https://www.who.int/news/item/13-12-2017-world-bank-and-who-half-the-world-lacks-access-to-essential-health-services-100-million-still-pushed-into-extreme-poverty-because-of-health-expenses