

A Mobile Application for Healthcare Self-Monitoring

Chua Tian Foo¹, Suhaimi Abd Ishak^{1*}

¹Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author Designation

DOI: <https://doi.org/10.30880/aitcs.2023.04.01.017>

Received 19 June 2022; Accepted 26 September 2022; Available online 30 June 2023

Abstract: A mobile application for healthcare self-monitoring is an application built and developed to help the general people communicate more efficiently with medical professionals. This mobile application was created to enable the general public to receive customized services or consultations from medical professionals. Obtaining the service is time-consuming and inefficient, and there is no dedicated platform for communication between users and medical professionals. The Agile software development technique was used to create the project. The Firebase database and Android Studio tools are used to create the application. After the implementation phase, the functional testing and User Acceptance Testing were built. Overall, this mobile application might provide a specific platform for users and medical experts to communicate and share accurate healthcare information.

Keywords: Healthcare, Self-Monitoring, Mobile-Based, Web-Based, Agile

1. Introduction

The World Health Organization (2000) described the fundamental aim as “all activities whose primary purpose is to promote, restore, and maintain health.” [1] which efforts to affect health determinants as well as more direct health-improving behaviors are included in its definition of a healthcare system. The World Health Organization's (WHO) Global Observatory for eHealth defines mobile health (mHealth) as “medical and public health practice supported by mobile devices, such as mobile phones, patient monitoring devices, personal digital assistants (PDAs), and other wireless devices.” [2]. Mobile health (mHealth) technologies have the potential to supplement or replace medical advice because to their widespread availability and use. However, most evaluation attempts are currently unable to effectively analyze all elements of mHealth solutions, which is concerning. This project is to create a mobile application to provide a platform for the public to get a tailored service based on their health condition.

The usage of information technology in the healthcare field is growing at an exponential rate all around the world. Thus, to learn more about healthcare, the public seeks advice from experts such as physicians, doctors, and clinicians. Some of them, however, do not have the opportunity to confer with

*Corresponding author: suhaimiabd@uthm.edu.my

these experts. They do not have a dedicated platform and are forced to rely on social media to communicate. This condition makes it harder for the general people to obtain more medical professional guidance. Besides, general user has to travel a distance to specific locations just to seek advice or need guidance from medical professionals which the process more complicated and inconvenience esp. Many consumers lack expertise and comprehension of the health-care system, as well as views of the severity of illness, technology literacy, and health literacy.

Other than that, some of certain locations provide medical professional services that providing healthcare guidelines or suggestions, still getting the service is difficult and inefficient for the public. It is possible that it is because the registration, appointment, and consultation processes take too long which bring inconvenient for individuals who work in offices, since they require time to personally attend to the medical professionals. This proposes healthcare mobile application able to provide a platform for them to communicate or receive medical professional guidance with medical professionals remotely without travelling to specific locations.

Based on the problem statement, several objectives can be identified. The objectives of this project are:

1. To design a mobile application for healthcare using object-oriented approach based on user requirements.
2. To develop the mobile application for healthcare using Android Studio Integrated Development Environment (IDE).
3. To test all the modules and functions of the healthcare mobile application.

On the other hand, Mobile Application for Healthcare Self-Monitoring included three user types: user, medical professional, and administrator. The user is the main user for this application as they will connect or communicate with medical professionals to have a tailored service. Besides, this Mobile Application included four functional modules: a module of user management, a module of communication between users and medical professionals, a module of self-monitoring health reports, and an administrator panel.

2. Related Work

A health system, also known as health care system or healthcare system, is the organization of people, institutions, and resources that deliver health care services to meet the health needs of target populations [3]. There are as many health systems as there are countries all around the world, each with its own history and organisational structure. Implicitly, nations must construct and develop health systems in accordance with their requirements and resources, despite the fact that primary healthcare and public health initiatives are common aspects in nearly all health systems [4].

Self-monitoring is a key aspect of patients' self-management, especially in diseases like diabetes, asthma and heart failure [5]. Self-monitoring may be classified into two sorts based on the function it serves: acquisitive and protective self-monitoring. The goal of acquisitive self-monitoring is to gain others' attention and approval. It entails analysing people' emotions and changing conduct in order to assist the individual fit in or gain attention, prestige, or power. The benefits of self-monitoring appear to be promising and research suggests that it can improve self-management, symptom control, and disease regulation, as well as decrease complications, improve patients' coping and attitudes toward their condition, create realistic goals, and improve quality of life.

Five applications are selected for comparison for the benchmark of the proposed application. The systems are Doctor Anywhere [6], MaNaDr [7], Speedoc [8], WhiteCoat [9], and Betterhelp [10]. A table will be represented later to show their difference and similarity with our proposed application.

Doctor Anywhere is a healthcare mobile application companion to assist people in taking control of their health condition and living a healthier life. The main dashboard of this healthcare mobile application allows the user to consult a locally licensed doctor anytime and anywhere. It provides

Doctor Anywhere virtual clinic which user able to get medication through simple video consultation with a locally licensed doctor.

On the other hand, MaNaDr is a healthcare mobile application that supports IOS and Android systems. Siaw Tung Yeng, the founders of MaNaDr, created the healthcare mobile application to combine the possibilities of traditional medical care and the digital world. This healthcare mobile application was design and developed by an ambitious team that believes in the transformative power of healthcare and provides a telemedicine service.

Speedoc is a mobile platform that aims to provide better services such as request for a doctor video consultation, nursing visit, medicine refill, and non-emergency ambulance, all in one place. By building a complete end-to-end system, Speedoc connect doctors to patients, making it affordable, accessible, and convenient for patients to receive medical care.

WhiteCoat is a major telehealth service that allows users to video consult a qualified general practitioner from anywhere in minutes using the healthcare mobile application. This healthcare mobile application's primary dashboard allows users to consult a locally licenced doctor at any time and from any location. From treating common ailments to chronic illnesses, quality medical care is now at your fingertips.

Lastly, Betterhelp is a healthcare mobile application that supports IOS and Android systems. Alon Matas, Danny Bragonier, are the founders of Betterhelp, created the healthcare mobile application to offer the convenient way to get professional help from a licensed therapist. Over 20,000 qualified, experienced, and accredited therapists are available through this healthcare mobile application.

The proposed application is an Android-based healthcare self-monitoring application that will be built to provide a platform for the targeted user who desires to remotely check their health. The software is designed for people who need to keep a closer eye on their health. The proposed application may let the user get connect with medical professional in a simple step. Table 1 shows the comparison between existing applications and proposed applications.

Table 1: Comparison between the Existing Applications and the Proposed Application

Features	Doctor Anywhere	MaNaDr	Speedoc	WhiteCoat	Betterhelp	Proposed Application
Android-based	✓	✓	✓	✓	✓	✓
IOS-based	✓	✓	✓	✓	✓	X
Web-based	✓	✓	✓	✓	✓	✓
Attractive User Interface	✓	X	X	X	✓	✓
Clear Navigation	✓	✓	✓	✓	✓	✓
Simple Interface	✓	✓	✓	✓	✓	✓
Everything Findable	✓	✓	✓	✓	✓	✓
Invisible Design	✓	✓	✓	✓	✓	✓
Many Banner Advertisements and Pop-Ups	X	✓	X	X	✓	X
Connection with Professionals	✓	✓	✓	✓	✓	✓
Medical Reports	✓	✓	✓	✓	✓	✓
Question-Based Personalized	X	X	X	✓	✓	✓
Remote Self-Monitoring	X	X	✓	✓	X	✓

* ✓ - Available, * X - Not Available

3. Methodology/Framework

An agile model is used in developing this project. The reason agile is employed is that Agile methods and procedures frequently aid in a regulated style of the management process that encourages frequent inspection and conversion. Agile provides an iterative approach for developing software ideas [11]. Iterative means to clarify the programme many times or through repetition and provide an output with the perfect objective established at the start. The development cycle will be conducted for the first time, allowing for additional software enhancement. The cycle will then be repeated to improve the project's performance and result.

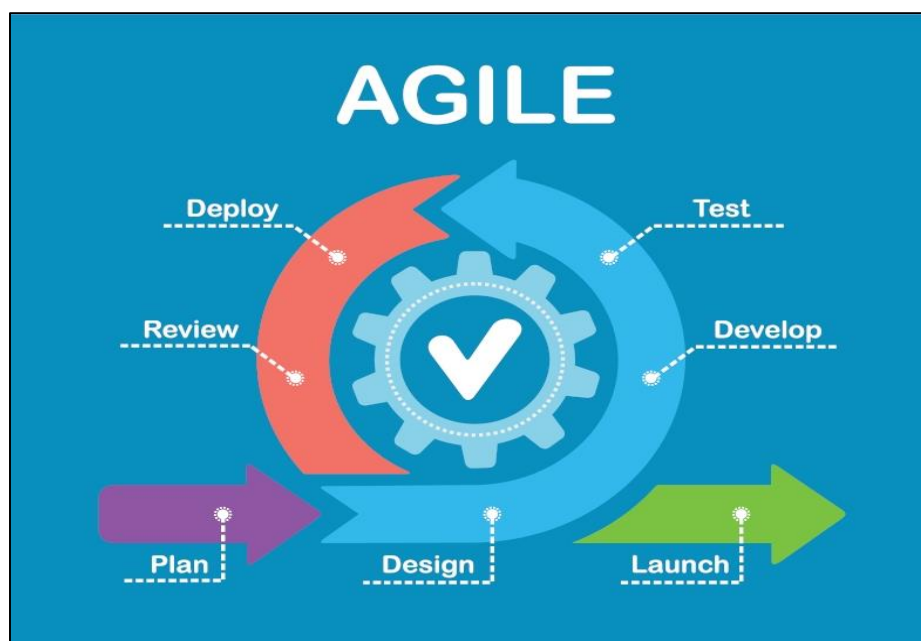


Figure 1: Agile Model [12]

3.1 Planning Phase

During the planning phase, the problem statements are identified, and the objectives are set to overcome the problems. The project's scope, including target users and functional modules, is set with the expected output discussed. A Gantt Chart is prepared to allow the project can be done systematically, prioritize, and do more in less time.

3.2 User Requirements

During this user requirement phase, first documentation containing a list of project requirements is developed. In this phase, the project proposal was released to explain the project's end accomplishment. The purpose of this phase is to define the project's user needs as well as the software and hardware requirements. A survey approach is used to determine the user's exact requirements. In this manner, the user may explicitly describe what they expect from the programme, and the developer can generate a result that meets the user's requirements. To collect information from responders, a questionnaire containing a series of questions and various prompts was created.

The functional and non-functional requirements are gathered to ensure the quality of the application. The functional requirement describes the system's behavior to the end-user, as displayed in Table 2. The non-functional requirements determine the quality attributes of the system, as exhibited in Table 3.

Table 3: Functional requirement of the proposed application

Function	Functionalities
Register	This function allows users and professionals to register a new user to the application with a valid email address, username, and password.
Login	This function allows the user to input a valid username and password.
Logout	This function allows users to log out from the application after using it.
Profile update	This function allows user to update their profile.
Self-Monitoring Health Report	User may upload their self-reports or health status image and medical professionals able to download and view.
Chat and Messaging	This function allows the user to chat and message with the medical professionals in real time without any delay.
Consultation	This function allows the user to connect with the professionals.
Manage user account	This function allows the administrator to update medical professional account status for validation purposes.
Manage Report Database	This function allows users, medical professionals, and administrators to retrieve or update data for the report database.

Table 4: Non-functional requirement of the proposed application

Requirement	Description
Operational	To allow for real-time database updates, the application will have an always-on data connection.
Performance	The programme will have an always-on data connection to enable for real-time database changes.
Security	Only a valid login and password are required to access the system.
Cultural and Political	Personal information regarding users cannot be sent outside of Malaysia.

3.3 Design Phase

System Architecture Diagram, Use Case Diagram, Sequential Diagram, Activity Diagram, Entity-Relationship Diagram (ERD) were illustrated in the design phase. All the diagrams are used to outline the system function, Activity Diagram for user as is displayed in Appendix A, activity diagram for medical professional is displayed in Appendix B and activity diagram for administrator is displayed in Appendix C.

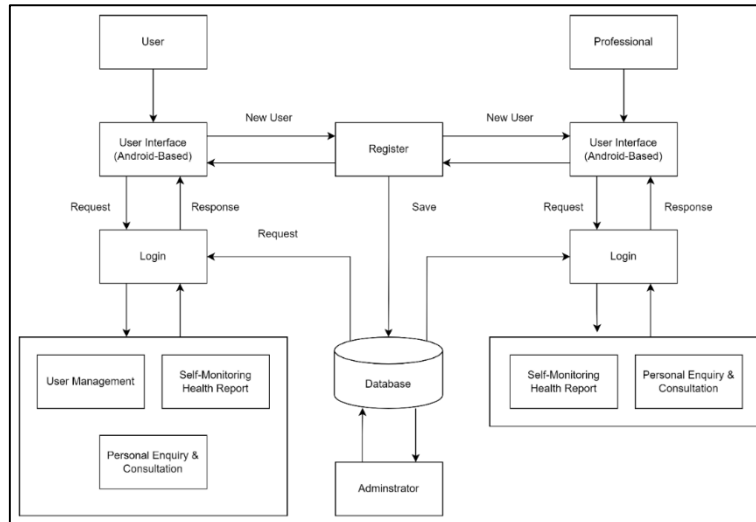


Figure 2: System Architecture Diagram

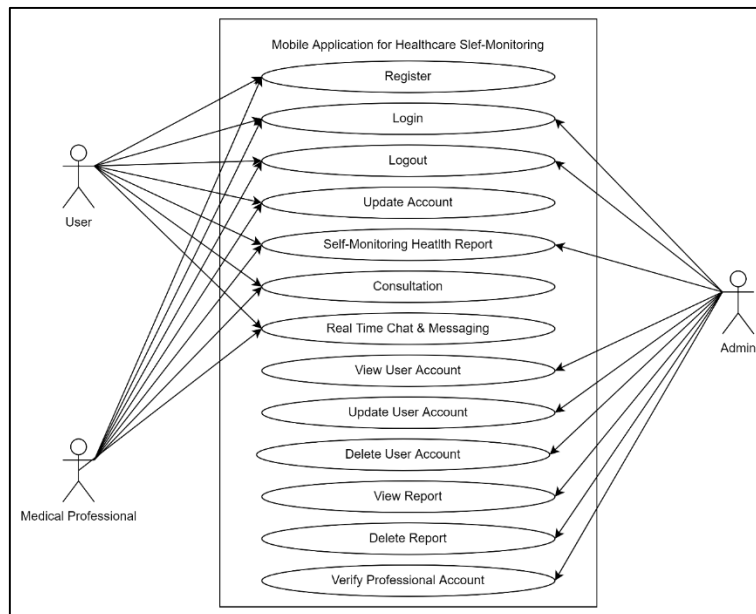


Figure 3: Use Case Diagram

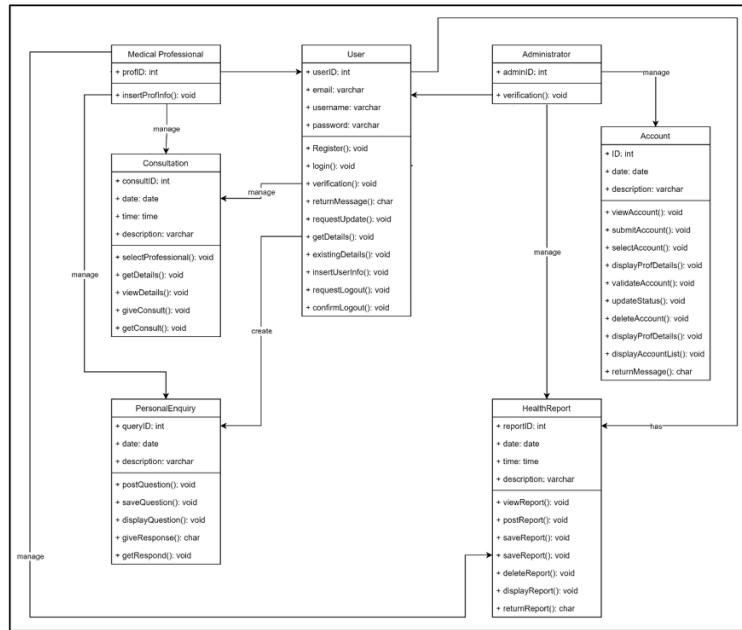


Figure 4: Class Diagram

3.4 Implementation Phase

After getting the user acceptance of the proposed system established, the implementation phase begins. Mobile Healthcare Self-Monitoring is developed using Android Studio IDE, which used Dart programming language for the both back-end user side and the front-end system. Mobile Healthcare Self-Monitoring Administrator Panel is developed in Flutter framework and Dart language. Firebase is used to set up the proposed system with security and recovery procedures fully defined.

3.5 Testing Phase

After getting the user acceptance of the proposed system established, the implementation phase begins. Mobile Healthcare Self-Monitoring is developed using Android Studio IDE, which used Dart programming language for the both back-end user side and the front-end system. Mobile Healthcare Self-Monitoring Administrator Panel is developed in Flutter framework and Dart language. Firebase is used to set up the proposed system with security and recovery procedures fully defined. The summary workflow for the development of the proposed system is shown in Table 2.

Table 2: Workflow for the Development of the Proposed System

Phases	Activity	Deliverables
Planning	1. Identify problem statements, objectives, the scope of the project, expected result, and project significance.	1. Project proposal.
	2. Set up a work plan.	2. Gantt Chart.
	3. Study online resources and articles that are related to the title.	3. Literature review.
	4. Study the features and functionality of existing applications.	4. Make comparison between existing applications and proposed applications.
	5. Questionnaire distributed.	5. Gather user requirements.
	6. Identify hardware and software requirements.	6. Hardware and software requirements.
	7. Identify functional and non-functional requirements.	7. Functional and non-functional requirements.
	8. Identify the relationship among all classes.	8. Use case diagram, sequence diagram, activity diagram, and class diagram.
Design	1. Design the wireframe.	1. Wireframes.
	2. Design the database.	2. Database specification.
Implementation	1. Develop the system module.	1. Proposed system.
	2. Integrate the system.	2. Errors found and fixed.
	3. Connect with the database.	
	4. A simple system prototype is developed initially for testing.	
Testing and Deploy	1. Conduct system testing	1. Fix and improve the bugs.
	2. Identify the area of improvement	2. Fix and ready to release the new system.
		3. Deploy the new system.

4. Results and Discussion

For both the front-end and back-end user sides of the healthcare mobile application, the Flutter framework and the Dart programming language were used. Flutter framework and Dart programming language were used to create the Healthcare Administrator Panel. To link the system to Firebase Realtime Database and Firebase Storage, Firebase SDKs have been implemented.

4.1 System Implementation

Figure 4 depicts the application registration interface. This activity will register an account for a new user, and there will be validation for the input field. Then, the system will check the username, phone number, and email in Firebase Database whether it has been registered and start the registration process if the data is unique.

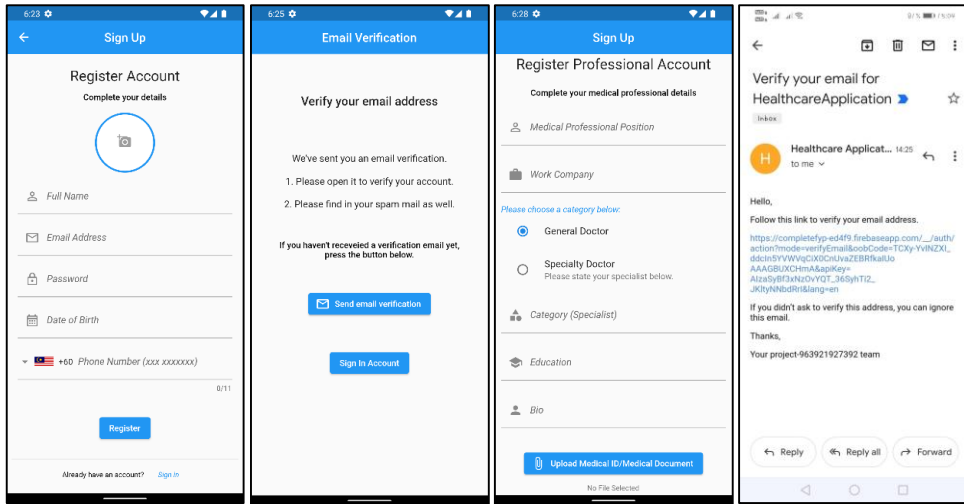


Figure 5: Register process for user and professional

Figures 5 shows the login interface for the application. The user or the professional can select the account type and login into the application by using their username and password. The application will check the status of professionals and redirect to the different interfaces based on the account status, which are pending, and verified.

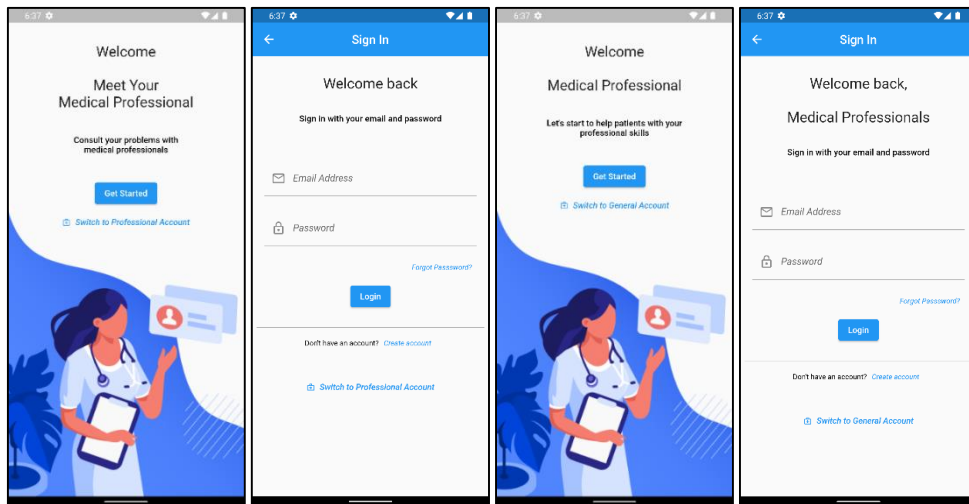


Figure 6: Login process for user and professional

Figure 6 shows the user profile interface for the user. Users are allowed to update their full name, and birthday as well as an alert dialogue box to confirm the user’s action before the logout process.

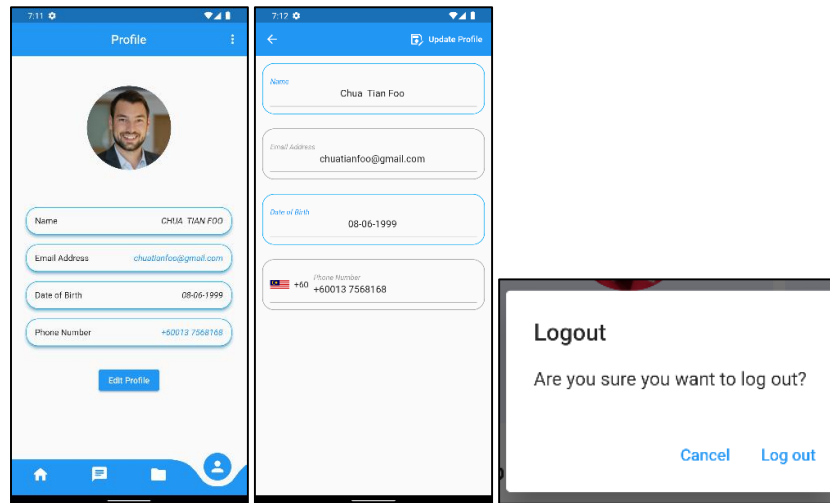


Figure 7: User profile interface for user and Logout confirmation

Figure 7 shows the user profile interface for the medical professional with pending status and verified status. Figure 8 shows the medical professional details in homepage and ‘Chat Now’ button in the details page. Chatroom will be created and listed in chat and messaging page.

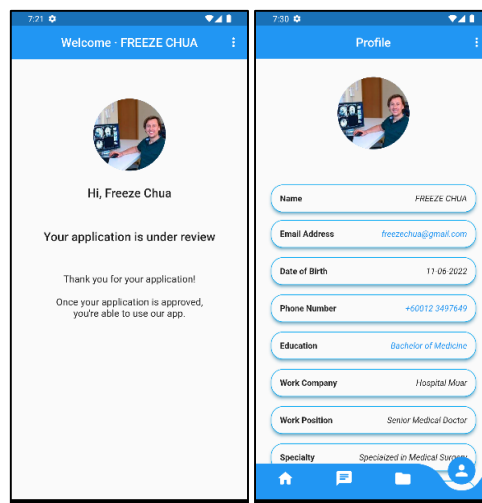


Figure 8: User profile interface for “pending” (left) and “verified” (right) medical professional

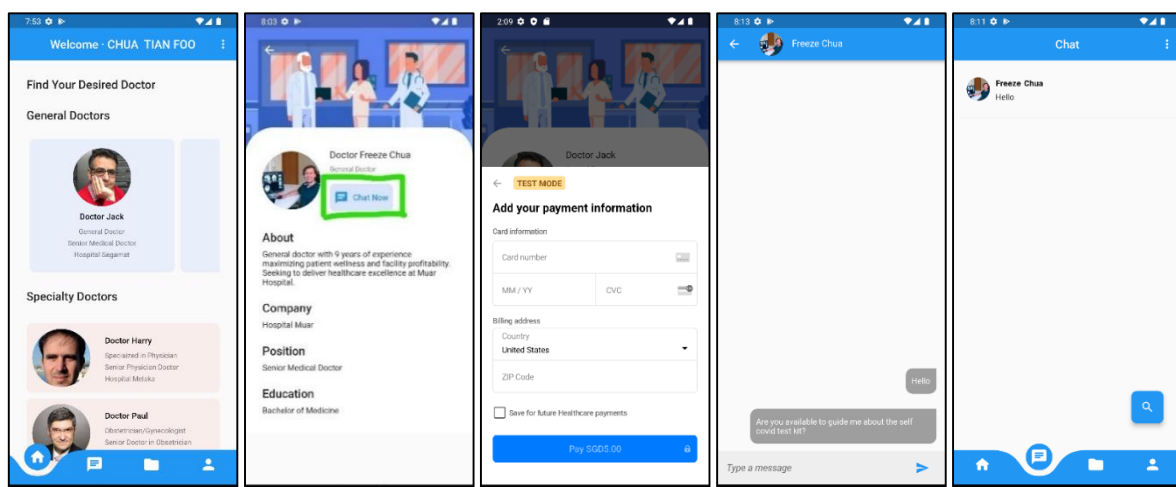


Figure 9: Medical professional details page, chat list and Realtime chatroom

Figure 9 shows search function which user able to search the medical professional by their name. Only verified medical professional are listed in the searched function.

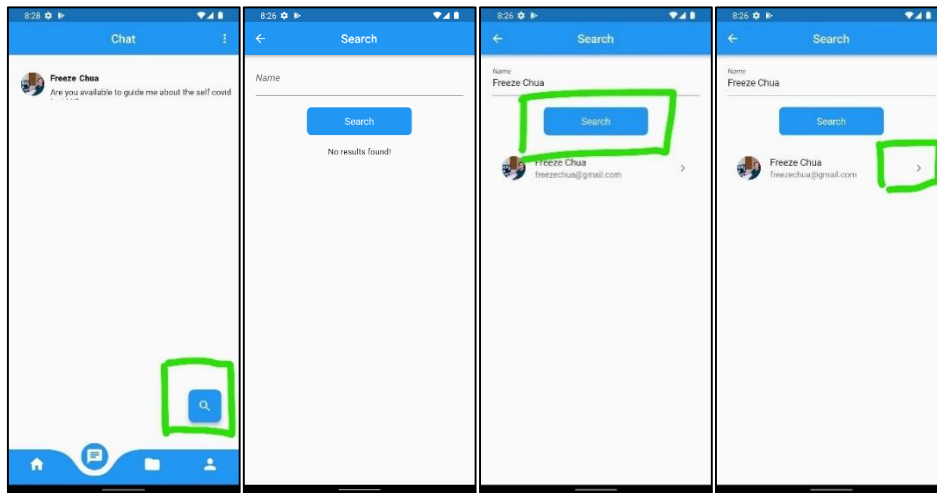


Figure 10: 'Search' button redirects to search screen and chatroom screen

Users can create and upload their self-report which will then list out in the report as well as update and delete their uploaded self-report as well as view the file as shown in Figure 10. The medical professional can track the total self-report that have been uploaded by users as shown in Figure 11. Medical professional download self-report shown in Figure 12.

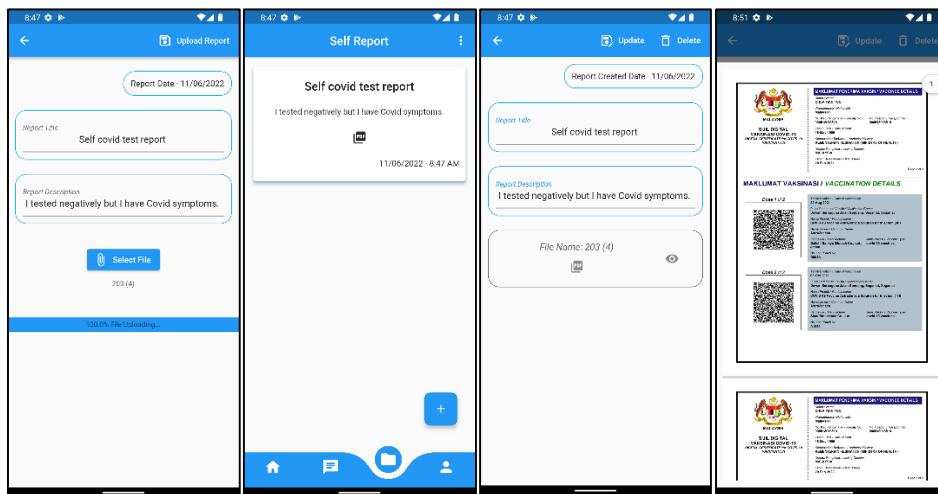


Figure 11: Add/Update/Delete/View self-report

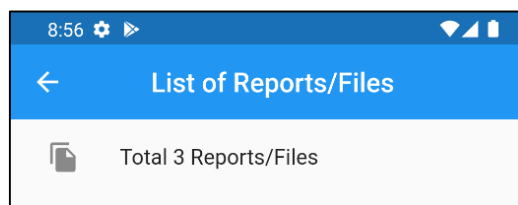


Figure 12: Summary of self-report list for medical professional

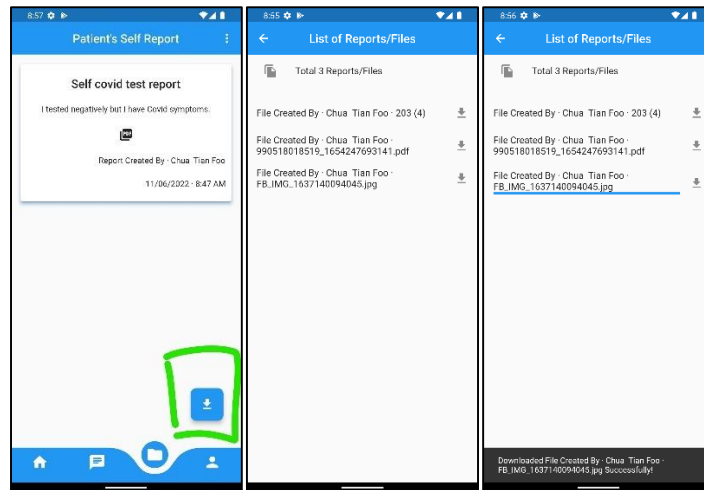


Figure 13: View and download self-report by medical professional

Healthcare administrator panel is a web application developed using Flutter framework and Dart language. Figure 13 shows the login interface for the administrator. In the dashboard function, the administrator can check the overview of the application that includes the total of users, total of questions and total of consultations as shown in Figure 14.

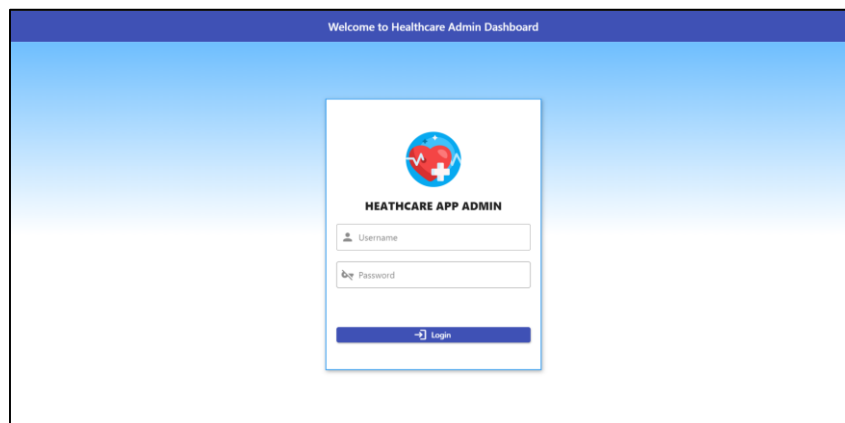


Figure 14: Login interface for Healthcare administrator

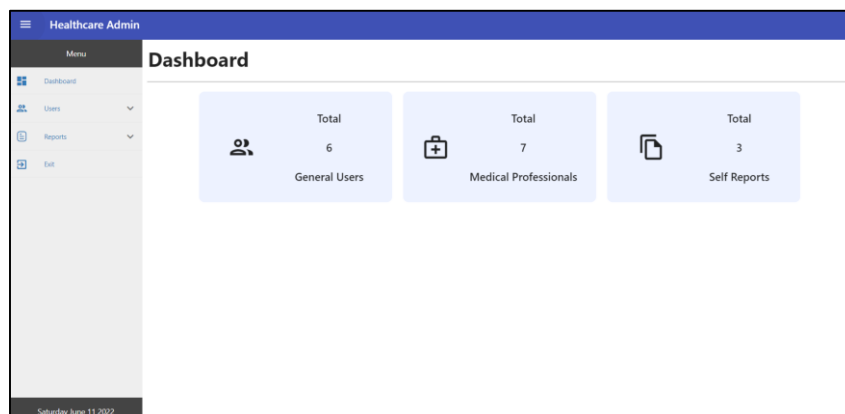


Figure 15: Dashboard interface for administrator

The administrator can view medical professionals and user list available in the mobile application in this interface, as shown in Figure 15 and Figure 16. The administrator can also update the professional's status after checking all the documents submitted by the professional. Figure 17 shows the report list uploaded by users and can be view and delete by administrator if needed.

PROFILE PICTURE	NAME	CONTACT	CATEGORY	WORK INFO	PROFESSIONAL DOCUMENT	STATUS	ACTION
	Jack	Email Address: jack@gmail.com Phone Number: +60012 3456789	Doctor Category: generalDoctor	Work Company: Hospital Segamat Work Position: Senior Medical Doctor	Documents	Status: Verified	Delete
	Harry	Email Address: freezechua@gmail.com Phone Number: +60013 6579511	Doctor Category: specialtyDoctor Specialized in Physician	Work Company: Hospital Melaka Work Position: Senior Physician Doctor	Documents	Status: Verified	Delete
	Freeze	Email Address: freeze@gmail.com Phone Number:	Doctor Category: generalDoctor	Work Company: Hospital Segamat Work Position:	Documents	Status: Verified	Delete

Figure 16: Professional list interface for administrator

PROFILE PICTURE	NAME	EMAIL	PHONE NUMBER	DOB	ACTION
	Loh Chun Kang	Email Address: ai190241@siswa.uthm.edu.my	Phone Number: +60111 115911	DOB: 15-08-1999	Delete
	Chua Tian Foo	Email Address: chuatianfoo@gmail.com	Phone Number: +60013 7568968	DOB: 08-06-1999	Delete
	Ong Pei Rui	Email Address: ai190238@siswa.uthm.edu.my	Phone Number: +600111986495	DOB: 05-08-1999	Delete
	Teh Yu Fung	Email Address: ai190243@siswa.uthm.edu.my	Phone Number: +60017 8844773	DOB: 18-08-1999	Delete

Figure 17: User list interface for administrator

REPORT FILE	SENDER	TITLE	DESCRIPTION	CREATED	ACTION
203 (4)	Chua Tian Foo	Self covid test report	I tested negatively but I have Covid symptoms.	11/06/2022 4:47 PM	Delete View

Figure 18: Self-report list interface for administrator

4.2 System Testing

System testing is a sort of software testing that is done on an entire integrated system to see whether it meets the requirements. Functional testing will be used in this project to evaluate the features and functionality of the proposed application.

4.2.1 Test Plan

A test plan is a structured document that specifies the software testing approach, objectives, timeframe, estimates, deliverables, and resources required. The Test Plan aids the developer in determining the amount of time and effort necessary to ensure the application's quality. The test plan's

goal is to ensure that each module functions properly and achieves the desired results. The test strategies for all of the functionalities given in the Healthcare application are shown in Tables 5.1 to 5.6.

Table 5.1: Testing results of a user management module

No.	Functions	Test Cases	Expected Results	Actual Results
1.	Registration	Incomplete data input	Display alert message if the input fields empty.	Pass
		Unique email address	An error message will display if the email address already exists in the database.	Pass
		Email verification	Need to verify email address using unique email generated by system.	Pass
		Complete registration form	Successfully register user with information and redirected to home page.	Pass
2.	Login	Incomplete data input	An alert message will display and ask the user to complete the login form.	Pass
		Complete input with invalid username or password	Alert message will display, and login request is rejected.	Pass
		Complete form	Login successfully and redirected to the homepage.	Pass
3.	Status of professional	Check the status of professional	Redirect to the different interface based on the professional's account status.	Pass
4.	User profile	Display the user profile data as registered	The profile page will show the username, full name, email and password.	Pass
		Update user profile with valid input	The user can update their profile picture, gender, birthday, email and phone number. The profile interface will then show the changes.	Pass
		Update professional profile with valid input	The professional can update their educational background for verification purposes.	Pass
		Update professional account status	When unverified professional submits their application, the account status will be changed to pending.	Pass
5.	Forget password	Update password with validate email address	A reset password email will be send to valid email address which is the email address is not valid then it unable to reset password.	Pass

Table 5.2: Testing results of reporting self-report summary function

No.	Functions	Test Cases	Expected Results	Actual Results
1.	Payment	Stripe gateway	Redirect to the Stripe SDK and proceed with the test mode process.	Pass
		Status of payment	Status payment will be shown.	Pass

Table 5.3: Testing results of communication between the user and medical professional

No.	Functions	Test Cases	Expected Results	Actual Results
1.	Consultation	View medical professional details	The medical professionals' details will be displayed if the user clicks on their profile picture.	Pass
		Realtime chat and messaging	The chat and messaging is in real-time without any delay.	Pass
2.	Chat and messaging	Send message	The message will be sent successfully from sender to receiver.	Pass
		Receive message	The message will be received successfully by the receiver from the sender.	Pass
3.	Search	Search valid user	Searched user will be listed out if the user is valid and available.	Pass
		Search invalid user	Searched user will not be listed out and will displayed an message of 'No user record found'.	Pass

Table 5.4: Testing results of reporting self-report summary function

No.	Functions	Test Cases	Expected Results	Actual Results
1.	Report	Calculate the total of self-reports uploaded by user	The total self-report uploaded will be count.	Pass

Table 5.5: Testing results of the administrator panel module

No.	Functions	Test Cases	Expected Results	Actual Results
1.	Dashboard	The total of the user exists in the database	The dashboard will display the total of user and professionals exists in the database.	Pass
2.	Medical professional list	View medical professional details	The medical professionals' details will be displayed in a table.	Pass
		Delete the medical professional account	Confirm dialogue will be displayed. If the yes button is clicked, the medical professional will be deleted from the database.	Pass
		View medical professional document	New window is opened to view the document file uploaded by medical professional for verification purposes.	Pass
3.	User list	View user details	The users' details will be displayed in a table.	Pass
		Delete the user account	Confirm dialogue will be displayed. If the yes button is clicked, the user will be deleted from the database.	Pass
4.	Self-report list	View self-report details	The self-report details will be displayed in a table.	Pass
		Delete the self-report	Confirm dialogue will be displayed. If the yes button is clicked, the self-report will be deleted from the database.	Pass

4.2.2 User Acceptance Test

User Acceptance Testing (UAT) is a type of testing in which the end-user or customer validates and accepts the software system before it is deployed. UAT is the ultimate level of testing after functional, integration, and system testing. Only 15 people participated in this testing due to time constraints: five professionals, five experienced users, and five newcomers. Following the data collection from the user, the output is assessed and presented in a graph. As seen in Appendix D, ranking one is extremely dissatisfied, whereas ranking five is extremely satisfied. Table 5.5 summarizes the findings and input from 15 people who evaluated the user interface. The majority of users were pleased with the application's simplicity of use, navigation, content arrangement, interface design, and text. No one was dissatisfied or extremely dissatisfied with the suggested application's interface design. Tables 5.6 present the results and outcomes of the application feature assessments. All of the data from Tables 5.5, 5.6 may be shown in the graphs in Figures 18 and Figure 19.

The test was divided into two parts: the user interface and the system's features, both of which were rated on a scale of 1 to 5. While 1 denotes very unsatisfactory, 5 denotes very satisfactory.

Table 5.5: Result of user interface evaluation

No.	Features	Rating					Total
		1	2	3	4	5	
1	Easy to use and understand	0	0	2	5	8	15
2	Navigation	0	0	0	9	6	15
3	Layout of the content	0	0	0	5	10	15
4	Interface design (e.g. background, colour)	0	0	0	11	4	15
5	Text (e.g. font family, font size)	0	0	4	5	6	15

Table 5.6: Result of application’s features evaluation

No.	Features	Rating					Total
		1	2	3	4	5	
1	Registration function	0	0	4	5	6	15
2	Login function	0	0	3	6	6	15
3	User information management function	0	0	1	5	9	15
4	Realtime chat and messaging function	0	0	0	8	7	15
5.	Self-report function	0	0	3	6	6	15

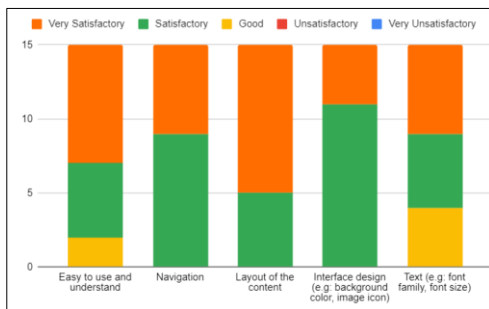


Figure 19: Result of user interface evaluation

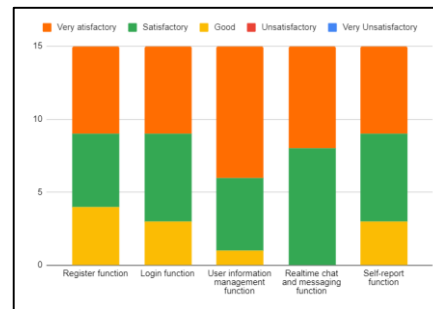


Figure 20: Result of application features evaluation

5. Conclusion

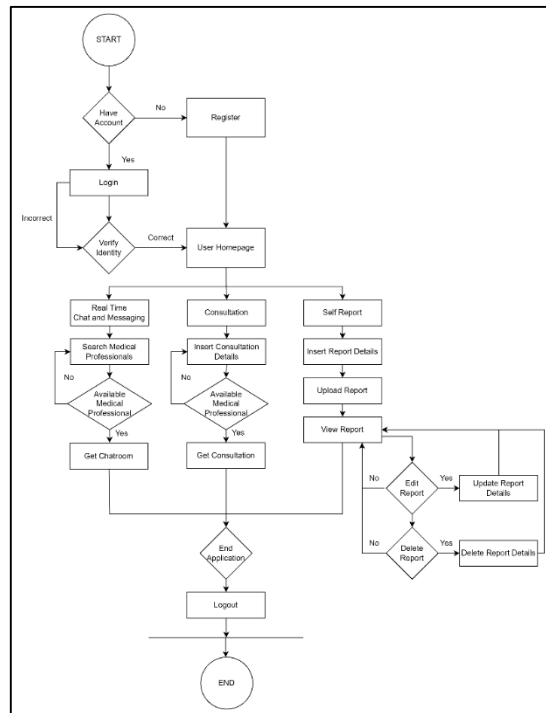
Although the majority of the proposed application's functions have been successfully built and the objectives have been met, there are some restrictions. Deep scrolling is used in the application to find the pros they want in the list. There is no search tool for users to use to look for it. Furthermore, just the most basic information on the professional is available, with no review or rating feature from previous users. The application's benefits and disadvantages may be analyzed utilizing the system's advantages, and new options for improving the weaknesses can be explored. To begin, a search bar should be included to the site so that users can quickly find the specialists they are looking for.

Then, in the future, the review and rating mechanism should be introduced. The user must fill it out after the personal inquiry or consultation session to assist the Healthcare application administrator in understanding the criticism or whether the professionals achieve their expectations for improvement. This feature will also assist newcomers when they first use the application. Furthermore, push alerts and reminders are also regarded a feature for both users and medical professionals, as well as the scheduled appointments.

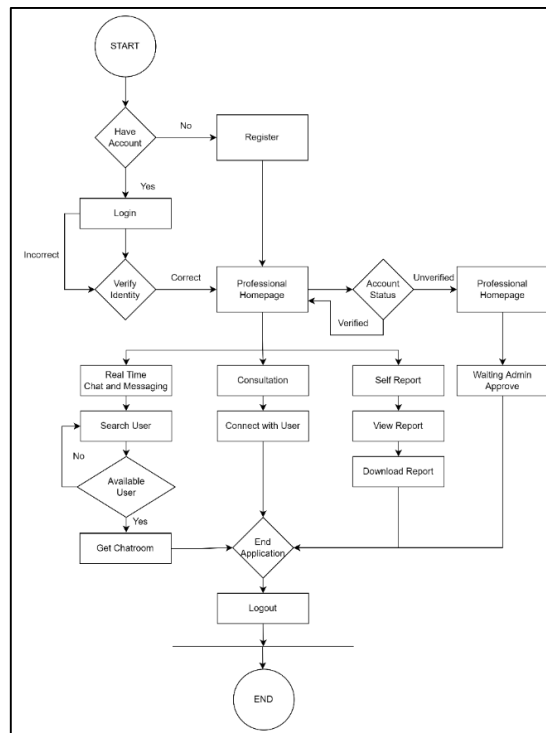
Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support and encouragement throughout the process of conducting this study.

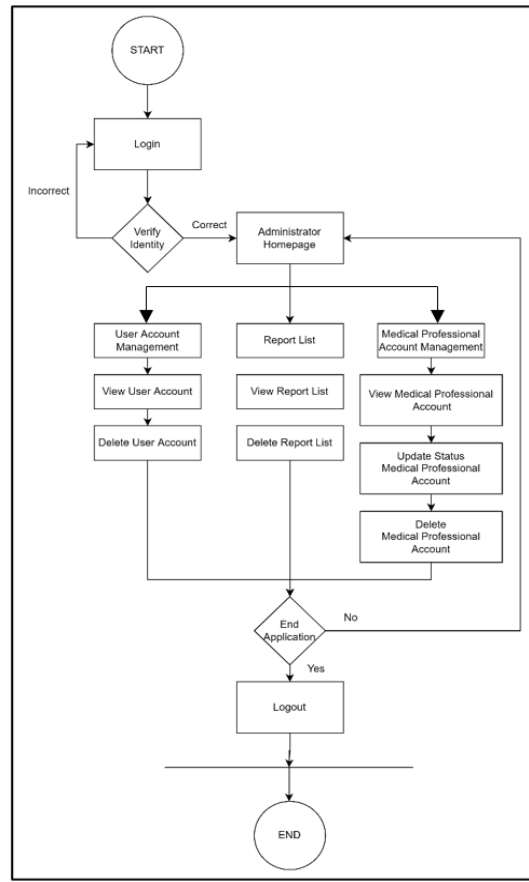
Appendix A (Activity Diagram for User)



Appendix B (Activity Diagram for Medical Professional)



Appendix C (Activity Diagram for Administrator)



References

- [1] World Health Organization. 2020 [Online]. Retrieved from World Health Organization: <https://www.euro.who.int/en/health-topics/Health-systems>
- [2] World Health Organization. 2020 [Online]. Retrieved from World Health Organization: <https://searchhealthit.techtarget.com/definition/mHealth>
- [3] World Health Organization. 2020 [Online]. Retrieved from World Health Organization: <https://www.euro.who.int/en/health-topics/Health-systems>
- [4] White F. "Primary health care and public health: foundations of universal health systems". *Med Princ Pract.* 24 (2): 103–116. doi:10.1159/000370197. PMC 5588212. PMID 25591411, 2015.
- [5] Barlow J, Wright C, Sheasby J, Turner A, Hainsworth J. Self-management approaches for people with chronic conditions: a review. *Patient Educ Couns.* 2002;48(2):177–87. [https://doi.org/10.1016/S0738-3991\(02\)00032-0](https://doi.org/10.1016/S0738-3991(02)00032-0), 2002.
- [6] Lim, W. M., Lai, W. M., "Doctor Anywhere", 2021 [Online]. Available: <https://www.doctoranywhere.my/>
- [7] Siaw, T. Y., "MaNaDr", 2021 [Online]. Available: <https://manadr.com/>
- [8] Verma, S., "Speedoc", 2021 [Online]. Available: <https://speedoc.com/my>
- [9] Koh, B., "WhiteCoat", 2021 [Online]. Available: <https://whitecoat.com.sg/>
- [10] Matas, A., Bragonier, D., "Betterhealth", 2021 Available: <https://www.betterhelp.com/>
- [11] Soundararajan, S., Arthur, J. D., and Balci, O. A Methodology for Assessing Agile Software Development Methods. *Agile Conference.* Dallas: TX. pp. 51-54, 2012.
- [12] Lark. Selecting the Right Business Methodology: AGILE. 2021 [Online]. <https://www.larksuite.com/blog/agile-methodology>