

**FACIAL AND SKINCARE SERVICE SYSTEM**

**BY**

**LOW CHIA HUI**

**A REPORT**

**SUBMITTED TO**

**Universiti Tunku Abdul Rahman**

**in partial fulfillment of the requirements**

**for the degree of**

**BACHELOR OF COMPUTER SCIENCE (HONOURS)**

**Faculty of Information and Communication Technology**

**(Kampar Campus)**

**JAN 2023**

## REPORT STATUS DECLARATION FORM

**Title:** FACIAL AND SKINCARE SERVICE SYSTEM  
\_\_\_\_\_  
\_\_\_\_\_

**Academic Session:** JAN 2023

I LOW CHIA HUI  
**(CAPITAL LETTER)**

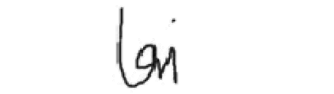
declare that I allow this Final Year Project Report to be kept in  
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.



(Author's signature)

Verified by,



(Supervisor's signature)

**Address:**

385, LORONG 3, JALAN  
BATU KAWA, TAMAN PHOENIX  
93250 KUCHING, SARAWAK

Lai Siew Cheng

Supervisor's name

**Date:** 28/04/2023

**Date:** 28/4/2023



## DECLARATION OF ORIGINALITY

I declare that this report entitled “**FACIAL AND SKINCARE SERVICE SYSTEM**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature :  \_\_\_\_\_

Name : LOW CHIA HUI

Date : 28/04/2023

## **ACKNOWLEDGEMENTS**

I would like to thank my supervisor, Ts Lai Siew Cheng and my moderator, Ts Yeck Yin Ping, who gave me a golden opportunity to engage in embedded machine learning features in a mobile application. Apart from that, they have given me a great deal of direction to finish this project as their assistance has been incredibly beneficial to me. I am grateful to express my sincere appreciation to my parents and family for their love, encouragement, and support that they have extended to me throughout my journey.

## **ABSTRACT**

Skin care management is paramount to ensure the clinic's management is smooth and manageable, as well as the first impression from the customer. Additionally, good management will ease the clutter and look professional to the customer. First impressions and trust from the customer are crucial for all businesses because the customer is the person that purchases the service and product. Running a Skincare Clinic is not just helping the customer feel beautiful after receiving the treatment. It is also about the trust gained from the customer. It is undeniable that clinic management is challenging to the clinic staff, whether in outpatient clinics, aesthetics clinics, dental clinics, or other clinics that are available in the market. This is because the problem of the patient is traditionally diagnosed by manual detecting and verifying, which is unsatisfying as the traditional diagnosis entirely depends on the professional skills and knowledge of the specialist in the skin care centre. The skin care management system is to solve the traditional way of diagnosing a patient's skin problem, which is subjective, requires great physical effort, and is time-consuming. Besides, the system also helps to encourage communication between the skincare specialist and the patient via online appointments. The proposed system is also designed to give flexibility to the staff and patient in concern of skincare service while using the system where they can use the mobile application. The system will use skin classification to replace the conventional way of diagnosis, which can give the specialist a better understanding of the patient's problem and provide appropriate feedback and guidance to the patient. Besides, the system allows online consultation, which enables the patient to reach the specialist in the skin care centre without physically going to the clinic. Other than that, the patient can make their appointment by using the system which is more flexible and convenient for both staff and patient because they do not require to negotiate the time to match with each other. The solution of this project is to improve the communication between the specialist and the patient to deliver feedback on the skin problem faced by the patient through online platform. The combination of video call plugin, TensorFlow Lite, Cloud Firestore Database and Flutter mobile application that able to manage the patient consultation and diagnose the skin condition of the patient is the aim of this project. The project has successfully fully completed with the features that allow the staff and patient to communicate using video call, manage the appointment booking, acne classification for acne type, and view the consultation history.

# TABLE OF CONTENTS

<b>TITLE PAGE .....</b>	<b>I</b>
<b>REPORT STATUS DECLARATION FORM .....</b>	<b>II</b>
<b>FYP THESIS SUBMISSION FORM .....</b>	<b>III</b>
<b>DECLARATION OF ORIGINALITY .....</b>	<b>IV</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>V</b>
<b>ABSTRACT.....</b>	<b>VI</b>
<b>TABLE OF CONTENTS .....</b>	<b>VII</b>
<b>LIST OF FIGURES .....</b>	<b>X</b>
<b>LIST OF TABLES .....</b>	<b>XVI</b>
<b>LIST OF ABBREVIATIONS .....</b>	<b>XVII</b>
<b>CHAPTER 1 INTRODUCTION .....</b>	<b>1</b>
1.1        Problem Statement and Motivation .....	1
1.2        Objectives .....	4
1.3        Project Scope and Direction.....	5
1.4        Contributions.....	7
1.5        Report Organization.....	8
<b>CHAPTER 2 LITERATURE REVIEW .....</b>	<b>9</b>
2.1 Previous Works / Related Works .....	9
2.1.1 Acne Classification using Traditional Machine Learning .....	9
2.1.2 Acne Classification using Deep Learning Algorithm.....	9
2.1.3 Limitation of Previous Studies.....	12
2.1.4 Proposed Solutions.....	12
2.2 Product Review .....	13
2.2.1 Zenoti [11] .....	13

2.2.2 WESS [12] .....	16
2.2.3 Nookal [13] .....	18
2.2.4 Clinicminds [14] .....	21
2.3 Summarization of Product Review .....	25
<b>CHAPTER 3 SYSTEM METHODOLOGY/APPROACH.....</b>	<b>27</b>
3.1 System Design Diagram .....	27
3.1.1 System Architecture Diagram.....	29
3.1.2 Use Case Diagram and Description .....	33
3.1.3 Activity Diagram .....	79
<b>CHAPTER 4 SYSTEM DESIGN .....</b>	<b>83</b>
4.1 Project Flow Diagram .....	83
4.2 Train Deep Learning Model using Transfer Learning.....	85
4.2.1 TensorFlow Lite Model Building Flowchart .....	86
4.2.2 Dataset Preparation Process .....	88
4.2.3 Pre-Training Preparation Process .....	92
4.2.4 Training and Testing Process.....	95
4.2.5 Output Process .....	97
4.3 System Design .....	99
4.3.1 Application System Flowchart.....	102
4.3.2 User Account Registration and Login.....	105
4.3.3 Home Interface.....	107
4.3.4 List Interface .....	109
4.3.5 Schedule Interface.....	111
4.3.6 Profile Interface .....	113
4.4 Database Design.....	115
<b>CHAPTER 5 SYSTEM IMPLEMENTATION .....</b>	<b>125</b>
5.1 Hardware Setup.....	125
5.2 Software Setup .....	126
5.3 Setting and Configuration .....	129
5.3.1 Flutter Android Configuration .....	129
5.4 System Operation.....	131
5.4.1 Account Login .....	131
5.4.2 Staff Module .....	133
5.4.3 Patient Module .....	145
5.5 Implementation Issues and Challenges .....	156



5.6 Concluding Remark .....	156
<b>CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION .....</b>	<b>157</b>
6.1 TensorFlow Model Performance Evaluation .....	157
6.1.1 Accuracy and Loss .....	157
6.2 Testing Setup and Result .....	160
6.2.1 Login Module.....	160
6.2.2 Staff & Patient Module .....	164
6.2.3 Staff Module .....	175
6.2.4 Patient Module .....	189
6.3 Project Challenges .....	199
6.3.1 Resolved Project Challenges.....	199
6.3.2 Unresolved Project Challenges .....	200
6.4 Objectives Evaluation .....	201
6.5 Concluding Remark .....	202
<b>CHAPTER 7 CONCLUSION AND RECOMMENDATION.....</b>	<b>203</b>
7.1 Conclusion .....	203
7.2 Recommendation .....	205
<b>REFERENCES.....</b>	<b>206</b>
<b>WEEKLY LOG.....</b>	<b>209</b>
<b>POSTER.....</b>	<b>222</b>
<b>PLAGIARISM CHECK RESULT .....</b>	<b>223</b>
<b>FYP2 CHECKLIST .....</b>	<b>227</b>

## LIST OF FIGURES

<b>Figure Number</b>	<b>Title</b>	<b>Page</b>
Figure 2.1.2.1	Architecture of pre-trained CNN and machine learning classifier	10
Figure 2.1.2.2	Formula to calculate Dice Coefficient	11
Figure 2.2.1.1	Form for online booking and appointment	13
Figure 2.2.1.2	Example of digital form for patient	14
Figure 2.2.1.3	General view for report and analytics features	15
Figure 2.2.2.1	Example of the appointment with detailed information	16
Figure 2.2.2.2	Example of automated appointment reminder	17
Figure 2.2.3.1	Sample of appointment and bookings form	18
Figure 2.2.3.2	Sample of appointment reminders using SMS	19
Figure 2.2.3.3	Sample of appointment reminders using email	19
Figure 2.2.3.4	Sample of clinical notes	20
Figure 2.2.4.1	Video call for online appointment meeting	22
Figure 2.2.4.2	Template for customer treatment records	23
Figure 2.2.4.3	Time schedule for appointment booking	24
Figure 3.1	Overall system architecture design of facial and skincare service system	28
Figure 3.1.1.1	Facial and Skincare Mobile Application Architecture Diagram for Staff Module	31
Figure 3.1.1.2	Facial and Skincare Mobile Application Architecture Diagram for Patient Module	32
Figure 3.1.2.1	Use Case Diagram for Facial and Skincare Mobile Application	33
Figure 3.1.3.1	Activity Diagram for Staff Module in Facial and Skincare Mobile Application	80
Figure 3.1.3.2	Activity Diagram for Patient Module in Facial and Skincare Mobile Application	82
Figure 4.1	Project Flow Diagram	83
Figure 4.2.1	Flowchart for Building TensorFlow Lite Model	86

Figure 4.2.3.1	Import Libraries	92
Figure 4.2.3.2	Initialization of variables	92
Figure 4.2.3.3	Import datasets	92
Figure 4.2.3.4	Snippet code for preprocess_input	92
Figure 4.2.3.5	Snippet Code for Data Augmentation	93
Figure 4.2.3.6	Snippet Code for initialization of EfficientNetV2B0 model	93
Figure 4.2.3.7	Snippet Code for freezing the convolutional base	94
Figure 4.2.4.1	Snippet Code for rebuild layers on top layer	95
Figure 4.2.4.2	Snippet Code for unfreeze some layers	95
Figure 4.2.4.3	Snippet Code for declaring hyperparameters	96
Figure 4.2.4.4	Snippet Code for training model	96
Figure 4.2.5.1	Snippet Code for Converting model to TensorFlow Lite Model	97
Figure 4.2.5.2	Snippet Code for Testing Converted TensorFlow Lite Model	97
Figure 4.2.5.3	Snippet Code for Saving Converted TensorFlow Lite Model	98
Figure 4.3	System Design of facial and skincare service system	99
Figure 4.3.1	Flowchart for Application System	103
Figure 4.3.2.1	Account Registration Block Diagram for Patient Module	105
Figure 4.3.2.2	Account Registration Block Diagram for Staff Module	105
Figure 4.3.2.3	Account Login Block Diagram	106
Figure 4.3.3.1	Home Interface Block Diagram for Patient Module	107
Figure 4.3.3.2	Home Interface Block Diagram for Staff Module	107
Figure 4.3.4.1	List Interface Block Diagram for Patient Module	109
Figure 4.3.4.2	List Interface Block Diagram for Staff Module	109
Figure 4.3.5.1	Schedule Interface Block Diagram for Patient Module	111
Figure 4.3.5.2	Schedule Interface Block Diagram for Staff Module	111
Figure 4.3.6.1	Profile Interface Block Diagram for Patient Module	113
Figure 4.3.6.2	Profile Interface Block Diagram for Staff Module	113
Figure 4.4.1	Database design for Facial and Skincare Service System	115
Figure 4.4.2	Staffs Entity and Leaves Entity	116
Figure 4.4.3	Staffs Entity and staff favourite Entity	116
Figure 4.4.4	Staffs Entity and appointments Entity	117
Figure 4.4.5	Staffs Entity and tokens Entity	117

Figure 4.4.6	Users Entity and favourite Entity	117
Figure 4.4.7	Users Entity and Appointments Entity	118
Figure 4.4.8	Users Entity and acne areas Entity	118
Figure 4.4.9	Users Entity and tokens Entity	118
Figure 4.4.10	leave Entity	119
Figure 4.4.11	Example documents for leave entity	119
Figure 4.4.12	staffFavourite Entity	119
Figure 4.4.13	Example documents for staffFavourtie entity	119
Figure 4.4.14	favourite Entity	120
Figure 4.4.15	Example documents for favourtie entity	120
Figure 4.4.16	acneAreas Entity	121
Figure 4.4.17	Example documents for acneAreas entity	121
Figure 4.4.18	staffs Entity	122
Figure 4.4.19	Example documents for staffs entity	122
Figure 4.4.20	appointments Entity	123
Figure 4.4.21	Example documents for appointments entity	123
Figure 4.4.22	users Entity	124
Figure 4.4.23	Example documents for users entity	124
Figure 4.4.24	tokens Entity	124
Figure 4.4.25	Example documents for tokens entity	124
Figure 5.2.1	Visual Studio Code Logo	126
Figure 5.2.2	Flutter Logo	126
Figure 5.2.3	Dart Logo	127
Figure 5.2.4	TensorFlow Logo	127
Figure 5.2.5	Keras Logo	128
Figure 5.3.1.1	Update the Dependencies in Build Gradle	129
Figure 5.3.1.2	Update the SDK Version in Build Gradle	129
Figure 5.3.1.3	Add In the aaptOptions in Build Gradle	130
Figure 5.3.1.4	Permission Access Used	130
Figure 5.4.1.1	Login Interface	131
Figure 5.4.1.2	Account Login verification - Successful	131
Figure 5.4.1.3	Account Login verification for password - Unsuccessful	132
Figure 5.4.1.4	Account Login verification for email address - Unsuccessful	132

Figure 5.4.2.1	Login Interface	134
Figure 5.4.2.2	Sign Up Form for Staff	134
Figure 5.4.2.3	Staff Account registration verification - Successful	135
Figure 5.4.2.4	Staff Account registration verification - Unsuccessful	135
Figure 5.4.2.5	On Leave Period in Homepage	136
Figure 5.4.2.6	Calendar Picker to pick start date and end date	136
Figure 5.4.2.7	Toast Message to alert user	137
Figure 5.4.2.8	Start Date and End Date shows in the text field	137
Figure 5.4.2.9	Patient List View	138
Figure 5.4.2.10	Patient Profile Details	138
Figure 5.4.2.11	List of Consultation History	139
Figure 5.4.2.12	No Consultation History Message	139
Figure 5.4.2.13	Favorite List Section in Homepage	140
Figure 5.4.2.14	Patient Profile Details for Favorite List	140
Figure 5.4.2.15	List of schedules - Staff	141
Figure 5.4.2.16	List of schedules is empty - Staff	141
Figure 5.4.2.17	Upcoming Interface -Staff	142
Figure 5.4.2.18	Complete Interface - Staff	142
Figure 5.4.2.19	Cancel Interface - Staff	142
Figure 5.4.2.20	Consultation Details for last appointment	143
Figure 5.4.2.21	Consultation Details for selected appointment	143
Figure 5.4.2.22	No Appointment Today	144
Figure 5.4.2.23	Appointment card for physical mode	144
Figure 5.4.2.24	Appointment card for online mode	144
Figure 5.4.2.25	Profile Details Interface	145
Figure 5.4.2.26	Edit Profile Details Interface	145
Figure 5.4.3.1	Login Interface	146
Figure 5.4.2.2	Sign Up Form for Patient	146
Figure 5.4.2.3	Patient Account registration verification - Successful	147
Figure 5.4.2.4	Patient Account registration verification - Unsuccessful	147
Figure 5.4.3.5	Specialist's List View	148
Figure 5.4.3.6	Specialist Profile Details	148
Figure 5.4.3.7	Appointment Booking Interface	149

Figure 5.4.3.8	Booking Confirmation Interface	149
Figure 5.4.3.9	Upcoming Interface-Patient	150
Figure 5.4.3.10	List of Schedule is empty	150
Figure 5.4.3.11	Complete Interface - Patient	151
Figure 5.4.3.12	Cancel Interface - Patient	151
Figure 5.4.3.13	Reschedule Appointment Interface- Patient	152
Figure 5.4.3.14	Consultation Details Interface- Patient	152
Figure 5.4.3.15	Favorite List Section in Homepage - Patient	153
Figure 5.4.3.16	Specialist Profile Details for Favorite List	153
Figure 5.4.3.17	Profile Details Interface - Patient	154
Figure 5.4.3.18	Edit Profile Details Interface - Patient	154
Figure 5.4.3.19	Acne Classification Interface - Patient	155
Figure 5.4.3.20	Consultation Details for last appointment - Patient	155
Figure 6.1.1.1	Snippet code for lists of hyperparameters	157
Figure 6.1.1.2	Accuracy and Loss for Hyperparameter 1	159
Figure 6.1.1.3	Accuracy and Loss for Hyperparameter 2	159
Figure 6.2.1.1	Test Result for test case TC_LUA_01	161
Figure 6.2.1.2	Test Result for invalid email and password (TC_LUA_02)	163
Figure 6.2.1.3	Test Result for invalid email (TC_LUA_02)	163
Figure 6.2.1.4	Test Result for non-existing user account (TC_LUA_02)	164
Figure 6.2.2.1	Test Result for test case TC_VC_01	165
Figure 6.2.2.2	Test Result for test case TC_VC_02	167
Figure 6.2.2.3	Test Result for Staff Module (TC_AIS_01)	168
Figure 6.2.2.4	Test Result for Patient Module (TC_AIS_01)	169
Figure 6.2.2.5	Test Result for Staff Module (TC_AIS_02)	170
Figure 6.2.2.6	Test Result for Patient Module (TC_AIS_02)	171
Figure 6.2.2.7	Test Result for Staff Module (TC_AFL_01)	172
Figure 6.2.2.8	Test Result for Patient Module (TC_AFL_01)	173
Figure 6.2.2.9	Test Result for Patient Module (TC_RFL_01)	174
Figure 6.2.2.10	Test Result for Patient Module (TC_RFL_01)	175
Figure 6.2.3.1	Test Result for test case TC_RUA_01	177
Figure 6.2.3.2	Test Result for test case TC_RUA_02	178
Figure 6.2.3.3	Initial Consultation Details in Cloud Firestore Database	180

Figure 6.2.3.4	Test Result for test case TC_ECD_01	181
Figure 6.2.3.5	Updated Consultation Details Store in Cloud Firestore Database	181
Figure 6.2.3.6	Test Result for test case TC_ACD_01	183
Figure 6.2.3.7	Added Consultation Details in Cloud Firestore Database	184
Figure 6.2.3.8	Initial Staff Profile Details in Cloud Firestore Database	186
Figure 6.2.3.9	Test Result for test case TC_EPD_01	186
Figure 6.2.3.10	Updated Staff Profile Details in Cloud Firestore Database	187
Figure 6.2.3.11	Test Result for test case TC_AOLP_01	188
Figure 6.2.4.1	Test Result for test case TC_RUA_01	190
Figure 6.2.4.2	Test Result for test case TC_RUA_02	191
Figure 6.2.4.3	Initial Patient Profile Details in Cloud Firestore Database	193
Figure 6.2.4.4	Test Result for test case TC_EPD_01	193
Figure 6.2.4.5	Updated Patient Profile Details in Cloud Firestore Database	194
Figure 6.2.4.6	Test Result for test case TC_AABD_01	195
Figure 6.2.4.7	Added Appointment Booking Details in Cloud Firestore	196
Figure 6.2.4.8	Initial Appointment Booking Details in Cloud Firestore Database	197
Figure 6.2.4.9	Test Result for test case TC_RAB_01	198
Figure 6.2.4.10	Updated Appointment Booking Details in Cloud Firestore Database	198

## LIST OF TABLES

<b>Table Number</b>	<b>Title</b>	<b>Page</b>
Table 2.3.1	Comparison of existing skin clinic management system	25
Table 4.2.1	Different types of acne provided in Kaggle	88
Table 4.2.2	Example of dataset with the labelling of acne grade	89
Table 5.1.1	Laptop used for system development	125
Table 5.1.2	Smartphone used for system development	125
Table 6.1	Accuracy and Hyperparameters	157
Table 6.2	Hyperparameters with Highest Accuracy	158



## LIST OF ABBREVIATIONS

<i>AI</i>	Artificial intelligence
<i>UI</i>	User Interface
<i>SDK</i>	Software Development Kit
<i>IDE</i>	Integrated Development Environment
<i>UV</i>	Ultraviolet
<i>PCOS</i>	Polycystic Ovary Syndrome
<i>CNN</i>	Convolutional Neural Networks
<i>ReLU</i>	Rectified Linear Unit
<i>VGG</i>	Visual Geometry Group
<i>SVM</i>	Support Vector Machine
<i>RF</i>	Random Forest
<i>LR</i>	Logistic regression
<i>NN</i>	Neural Networks
<i>PNG</i>	Portable Network Graphics
<i>HPC</i>	High-Performance Computing
<i>SGD</i>	Stochastic Gradient Descent
<i>Grand-CAM</i>	Gradient-weighted Class Activation Mapping
<i>LBP</i>	Local Binary Patterns
<i>HOG</i>	Histogram of Oriented Gradients
<i>SMO</i>	Sequential Minimal Optimization
<i>CRM</i>	Customer Relationship Management
<i>Email</i>	Electronic mail
<i>PDF</i>	Portable Document Format
<i>HIPAA</i>	Health Insurance Portability and Accountability Act
<i>PHI</i>	Protected Health Information
<i>GDPR</i>	General Data Protection Regulation
<i>API</i>	Application Programming Interface
<i>APK</i>	Android Application Package

# Chapter 1

## Introduction

This chapter will describe the project's context, motivation, contributions to the field of study, and outline.

### 1.1 Problem Statement and Motivation

The beauty and skincare industry has become popular because people look for a cure for their skin conditions. Moreover, people are concerned about their face and skin condition because they see other people from their physical appearance, which is more crucial when coming to job interviews and socializing. Hence, the demand for the resources and the services of skin care tips and products has become higher in the market. However, the acne skin problem is still hard to solve even though the services are available in the market.

Although AI is common in the skin care industry, the **manual diagnosis of acne** is still adopted in the skin care industry [1]. Clinic specialists must diagnose the acne category when the customer physically visits the store based on their knowledge and professional interactions. Therefore, the diagnosis is subjective and time-consuming. Furthermore, even though the appointment form will ask for the customer's skin condition, the customer might not know well about their skin condition. According to The Start [2], 85 % of Malaysian aged between 12 to 24 have the skin problem of acne. This indicates that many Malaysians do not know the procedure and tips for skin care. Hence, they might give inappropriate information regarding their skin condition, so the information provided is not helpful to analyze. Other than that, the clinic specialist is also hard to monitor the customer's skin condition, which will affect the performance of the skincare product. Therefore, the skin care product might not fully solve the customer's problem.

Even though skin care clinic is becoming more common now, most skin care clinic only provides **physical consultation** services. The skin care clinic only provides the appointment form for the customer interested in knowing the skin care services, and the consultation is done physically at the clinic. Therefore, customers need to meet the specialist physically to learn more about their skin condition. Furthermore, the **difficulty in booking the consultation**

**session** forces the customer to follow the time set by the clinic. It is not convenient for the customer who is an employee or a student. Other than that, it is also hard to monitor the customer's skin condition because the customer cannot reach the clinic specialist to ask for further guidelines and instructions that need to be followed when using the product given. Lack of procedures and education may affect the outcome and the effect of the skincare product. Hence, the acne skin problem cannot be solved as expected.

Hence, the motivation to propose a new mobile application for skin care service that uses machine learning methods to analyze the customer's skin type. The implementation of AI has helped to support the diagnosis in the healthcare industry, where the specialist is assisted in better understanding the needs of the people that require care. Therefore, a specialist can provide better feedback, advice, and instruction to help people with their health problems. Furthermore, after researching the paper related to the project title, the implementation of AI classification we provide in the mobile application can help the skin care clinic specialist diagnose the patient's skin type. Therefore, the specialist can provide better guidance to their customer to solve the skin problem effectively. Besides, the analysis is essential for improving the skin care industry because the data is collected, and skin care professionals can use the data collected to identify a better cure for the skin problem, which can contribute benefits to the skin care industry.

Other than that, consultation is important as it help the customer to know about their skin type and receive advice from a specialist to prevent them from trying the product without any direction [3]. Therefore, the proposal aims to identify a way to improve the communication between the customer and the specialist of the skincare clinic through online consultation. According to [4], virtual consultation has become popular during the outbreak of Covid pandemic which has change the face-to-face contact to online platform. Therefore, this project would like to develop a video call consultation to promote better service to the customer that helps them easily seek help from the specialist and gain information regarding the tips and procedures to protect their skin correctly and efficiently. Choosing a suitable product to cure skin problems is crucial, especially for people with sensitive skin. Otherwise, the skin condition will get worsen. However, the pathway to getting guidance and a deep understanding of skin conditions is through consultation. Therefore, consolation is essential to help the customer

recover from the skin problem because the specialist can help the customer identify the product suitable for their skin with the appropriate method and procedure.

Besides that, the appointment booking process in the system is more flexible as the booking calendar can be access by the patient which allow them to choose and decide the appointment date, time, and mode for the consultation without direct communication with the specialist to discuss about the appointment session. According to the [5], patients often prefer the option of online scheduling as they can book the appointment in a convenient way without calling the front desk or schedule the appointment during business hours. Other than that, it helps the patient to decide the appointment by themselves and they have more flexibility in rescheduling the appointment from time to time if they cannot attend the consultation session that have been booked earlier. Furthermore, the online scheduling can also help to minimize the appointment gap where the practice of online scheduling allows the patient to look for the slot that they available which reduce the risk of no-show during the consultation slot. Besides, online scheduling also helps to reduce the burden task for the staff where they can help the front desk staff to focus on the incoming phone calls and inquiries so that they will not be overwhelmed by the scheduling appointment. Online scheduling also give better patient experience of consultation period as the patient does not need to wait for their turn when going to the center physically.

In short, this project aims to enhance the customer experience and alleviate skin issues by improving the services that can be offered by the skincare Centre.

## 1.2 Objectives

The project aims to help the staff at the skin care Centre to classify the patient's skin problem, reducing the staff's effort to make the diagnosis manually based on knowledge and experience. Besides, the staff can also perform online video appointments that allow the patient to consult the specialist more easily. Therefore, the objective of this project in the development of the system is:

- **To provide video call consultation using video call plugin**

The staff can use the plugin video call service using ZEGOCLOUD video call plugin that built into the system to reach the patient. The service is mainly used to reach the patient through an online meeting platform. Furthermore, online video appointment allows the staff to meet the patient at any place and time and share skincare tips through the video meeting. This can increase the efficiency of the treatment because the patient can get more information regarding the skin care procedure and use the correct method to apply the product that the specialist has suggested.

- **To develop appointment booking in the mobile application**

The user can book their desire date for the consultation with the Flutter calendar with the specialist without asking for the availability of the time slot that the specialist have with is not convenient through phone call or texting. The appointment booking features allow the user to book their consultation time slot at anytime and anywhere. They can also book the appointment with another specialist if they wish which is more flexible and easier to operate.

- **To implement skin classification on acne type**

The staff and patient are allowed to use the skin classifier to diagnose the skin problem which develop using TensorFlow lite. The skin classifier can help classify the image taken using a camera or the image uploaded from the phone gallery. Therefore, the staff can take a photo of the patient when the patient visits the clinic. The patient can use the feature by uploading the images from the gallery or capturing the photo. The classification result will display when the process of classification is done. There is also some suggested specialist for the patient according to the classification result that they get.

### **1.3 Project Scope and Direction**

The scope of the project includes developing a Facial and Skincare service system on a mobile application that can improve the existing system. The facial and Skincare service system is a system that helps the Skincare Centre with their clinic management. The development includes a new feature, skin classification, used to support the specialist diagnose the skin problem and giving better feedback and guidance to the customer. Other than that, the development of online appointment is included in the system calling the user to join a one-to-one video call room, which allow the customer to join the meeting for an online consultation session.

The skin clinic management system is a mobile application with an android operating system. However, only a few existing systems support android devices, and the system with a better feature only provides an iOS operating system. Therefore, the development of the mobile application was using Flutter and Dart programming language to develop the interface of the mobile app. Google has developed an open-source UI SDK for the development of iOS/Android aps which is Flutter that use Dart as the programming language. The implementation of the app is using Visual Studio Code 1.73.1 to run the flutter. Visual Studio Code 1.73.1 is a popular IDE alternative tool that easier to use for building and debugging the project. Other than that, TensorFlow Lite is used to develop the skin classification feature because it requires a machine learning technique to perform the classification process. ZEGOCLOUD is one of the cloud-based platforms that offers a range of communication and collaboration services. The services that provide by ZEGOCLOUD include instant messaging, video conferencing, webinars, and file sharing. This platform's main objective is to help enhance communication and collaboration among staff members, clients, and suppliers in an organization.

The Skincare service system is provided for the skin care Centre to manage the staff and patient. Therefore, the system focuses on the staff module and user module. The staff for the skincare Centre, also known as the specialist. In the staff module, the specialist can perform tasks that include updating the on-leave date, view appointment schedule, joining video call, taking photos, uploading images, view patients' profile, view and edit treatment and consultation reports, editing own profile. Moreover, the user module is for the patient which is the customer of the skincare Centre. In the user module, the patient can edit their own profile, use upload images, take photo, view latest consultation report, view appointment schedule,

reschedule the appointment, search for desire specialist, book an appointment, joining video call room, and cancel appointment.

The flow of the system is the user need to book an appointment with the staff by choosing the consultation date, time slot, and consultation mode. When the user confirms to book the appointment, the appointment record of the user will then checked by the system if the user does not have the same appointment on the same date or the same staff in a day, the appointment booking record will be record in the database and the appointment booking record will not be recorded if the user have clashing appointment record in the database. When the appointment is booked, the staff will receive a notification regarding the appointment to notify the staff that they have a new appointment. Besides that, the staff and user will be able to see the consultation record in the schedules page. The system will check the consultation mode of the appointment, if the consultation mode is online, then both the staff and user need to join the call to perform the consultation session and the user is requested to update the skin type before joining the call. However, if the consultation mode is physical, then the user is required to visit the physical skincare Centre and the staff need to use the skin classification tools to classify the skin type of the patient. After completed the consultation session, the staff need to complete the consultation details or report to record the skin condition of the user. Both the staff and user can retrieve back the consultation details or report where the staff can search for the patient and view the consultation history in the profile of the patient and the user can view the latest consultation report in their profile page.

In short, the system is focused on delivering a convenient way to do the consultation through a video appointment using an online meeting platform where the staff and user need to join the one-to-one video call room. Besides that, user can book the appointment in a more flexible way without direct interacting with the staff. Moreover, the system provides an easier way to diagnose the skin condition by using the captured image or uploading the photo.

## 1.4 Contributions

The goal of this project is to create a mobile app that helps the Skincare Centre and its clients in various ways to deal with skin issues. AI skin classifier is implemented for skin classification according to the grade of the skin condition to provide a clear understanding of the skin condition, skin problem, and the critical activities that need to be taken care. Furthermore, provide a video appointment feature that allows two-way communication, which meet each other using an online platform. The video appointment feature is a third-party meeting application that requires the embedment of the video meeting plugin in the system to provide the video call features. The mobile application allows the specialist to work at any place by using the phone to track the customer's appointment and the customer's consultation history. Other than that, the specialist can monitor the customer's skin condition through a video appointment where the specialist does not need to physically meet the customer, reducing the time required to review the customer's skin condition.

Other than that, the mobile application allows the specialist in the skincare Centre to classify the skin condition of the customer by using the implemented AI classifier and store the result of the customer in the customer profile for reference. Customers can make an appointment and choose online consultations using video appointments through the proposed mobile application in this project. Customers can consult the specialist online without physically meeting the specialist, which is going to the clinic, requires driving to the place, and time is spent on the road. Therefore, the skincare mobile application provides one-click video meetings that allow the staff and customer to join into a one-to-one video call room. The customer needs to join the meeting for the consultation. Hence, it allows the consultation to be done at any time and place because customers can have a video meeting from their homes which is more convenient. Besides that, video consultation also allows the specialist to clearly share the information and guideline with the customer so that the customer can follow the correct method to take care of their skin.

In short, having this service system can help the skincare specialist to organize the appointment using an alternative method which is through online consultation. Moreover, the customer can book for their desire consultation date without direct communication with the specialist. Other than that, the skincare specialist can also use the AI scan to get the analysis of the skin condition for future use and giving a more accurate treatment for the patient.



## **1.5 Report Organization**

This report is organized into 7 chapters: where Chapter 1 is the Introduction, Chapter 2 is about the Literature Review, while Chapter 3 explain the System Methodology/ Approach. Chapter 4 is the System Design, Chapter 5 is about the System Implementation while Chapter 6 System Evaluation and Discussion and last chapter is Chapter 7 which is the Conclusion and Recommendation. The first section of this report is an overview of the project, including such topics as the project's stated issue, its stated purpose, its stated scope and direction, its stated aims, its stated contribution, and its stated organization. he second section, previous works in this field is examined, which is the previous works and related products that are currently available on the market for assessing the good design and weak design of products. Flowchart that depicting the system's technique is illustrate and explain in Chapter 3. The architecture of the mobile application system, the procedure for creating and deploying machine learning models, and the general flow of the project are discussed in Chapter 4.

In addition, hardware and software installation, environment configuration, and system operation are covered in detail in chapter five. The sixth section discusses the system assessment's result and the results from analysis of the machine learning model, the performance of the deployment for mobile applications, challenges in the project and objective evaluation. Lastly, the seventh chapter is the conclusion and discusses future work.

# Chapter 2

## Literature Review

The deep learning algorithm is widely used to perform all the detection and classification that involve artificial intelligence. Therefore, it is important to provide statistical results and a predictive model that contributes to better diagnosis and interpretation of data [6]. At the same time, CNN is the deep learning algorithm used most for image recognition and processing. In this project, the highlighted part will be classifying and grading skin acne.

### 2.1 Previous Works / Related Works

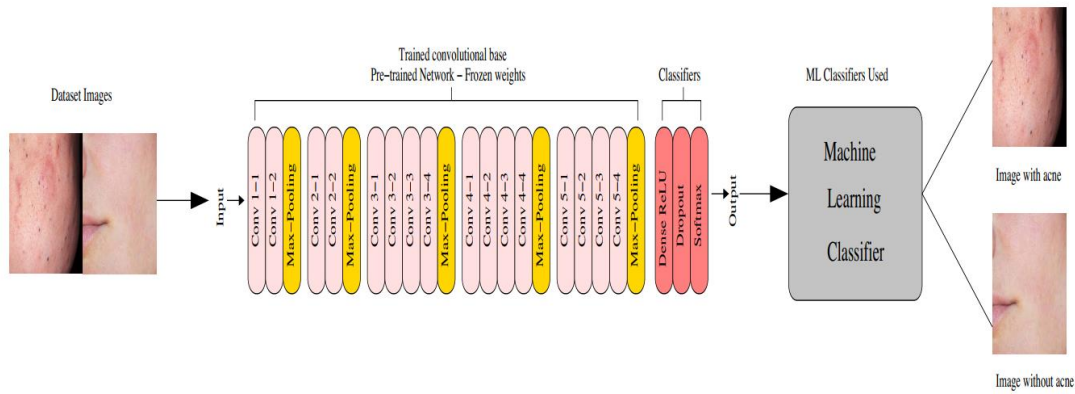
#### 2.1.1 Acne Classification using Traditional Machine Learning

According to the study conducted by [10], they have applied two different methods; the first method used is deep learning. The second method is using the same dataset for classification, which is traditional machine learning using Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG) and the use Sequential Minimal Optimization (SMO), which is the machine learning classifier to train Support Vector Machine (SVM) to perform classification. The author [10] has chosen 164 images used in various classification experiments. The accuracy obtained from the traditional machine learning technique is 74%.

#### 2.1.2 Acne Classification using Deep Learning Algorithm

There are several studies related to acne classification using deep learning algorithms have been selected for review and study. In the research paper, the authors use different types of CNN Models to classify the acne category. CNN is mainly used to process images and detect objects. Layers such as the Convolution Layer, ReLU, Pooling Layer, and the Fully Connected Layer are all part of a Deep Neural Network. [7]

The author in [8] has used VGG16, VGG19, and Inception V3, pre-trained CNN models on the input dataset set images to extract the features from the images. As stated in [8], this research proposed is to improve the previous research, which did not focus on categorizing acne. The author also applies SVM, RF, LR, NN, and AdaBoost as the machine learning classifier to classify the image that has acne and the image that does not has acne after the feature is extracted successfully [8].



**Figure 2.1.2.1: Architecture of pre-trained CNN and machine learning classifier**  
**Source: Adapted from [8].**

The author [8] mentioned that 200 images had been used, consisting of 100 normal skin images and 100 acne skin images with various backgrounds. Therefore, pre-processing image technique is required to crop the images, and the GrabCut tool is used to remove the remainder backdrop. After the pre-processing, the images are resized with the same height and width. According to [8], 99.5% is the highest accuracy rate result from the Inception V3 model using the logistic regression algorithm. On the other hand, the VGG19 model has the highest accuracy rate which is at 99%, which use kNN and a random forest algorithm. AdaBoost, kNN, and SVM algorithms for the VGG16 model obtained 99% for the maximum accuracy rate. In short, the best model used to classify the images is inception v3 that use logistic regression as the classifier because the accuracy obtained is the highest among the three models used in the study.

According to the reference [9], the author has collected the largest datasets compared to other studies. The dataset has been obtained from CelebMask-HQ, which has 30,000 face images with a high-resolution size of  $512 \times 512$ . In contrast, 70,000 high-quality resolution of PNG images of human faces has been obtained from Flickr-Face-HQ (FFHQ) with a size of  $1024 \times 1024$  that contains variation in age, ethnicity, and background of the images. The images from Flickr-Face-HQ (FFHQ) have been reannotated with the label of 0 – No Acne, 1 – Light Acne, 2- Moderate Acne, and 3 – Severe Acne). The author [8] mentioned that the previous studies extracted the region usually affected by acne, such as the cheeks, forehead, and chin, as their sample data. However, in the study [9], the author excludes the region that does not belong to the face and not the acne-affected area and uses a semantic segmentation model to analyze the area affected by the acne. DenseNet121 architecture is used for the classification of acne. The

author in [9] has used the Dice Coefficient to evaluating how well the semantic segmentation model works The Dice coefficient is calculated using the following formula:

$$\text{Dice Coefficient} = 2 \cdot \frac{\text{Area Intersection}}{\text{Area Union}}$$

**Figure 2.1.2.2: Formula to calculate Dice Coefficient**  
**Source: Adapted from [9].**

The author [9] mentioned that HPC Cluster that equips with two NVIDIA Tesla T4s is used to train the segmentation model, and the SGD is used as an optimizer. The final average of the Dice coefficient is 86.1% after 80.000 iterations. Grad-CAM has been applied to the acne samples to validate the result's correctness and validity.

According to the study conducted by [10], they have applied two different methods. The first method is Deep learning, where GoogLeNet architecture is used and implemented using the Caffe framework, while the traditional method will be discussed in section 2.1.2. There are 23,000 images of various skin diseases that have been obtained from the dataset of DermNet. The dataset consists of different ethnic groups, including African, Arabic, Chinese, Malaysian, and White British. However, the only useful dataset has been collected from the DermNet dataset. The author [10] has chosen 164 images used in various classification experiments. The accuracy obtained from the deep learning technique has given the highest accuracy rate of 85%.

### **2.1.3 Limitation of Previous Studies**

There are several limitations in the previous studies that can be improved in future work. The study [8] did not cover people with different skin colors even though the accuracy achieved is high, which is 99.5%. There is different ethnic group across the country with different skin colors, which need to be considered in the dataset to achieved higher accuracy of the skin classification.

Based on the study [9], the author claimed that the Flickr-Face-HQ dataset was annotated correctly with the scale for some dataset, which may cause a problem when processing the sample and affect the accuracy of the result. Furthermore, the dataset used in the experiment also does not match each other. Again, this will affect the result of the classification.

According to the study in [10], the dataset is collected in the environment that have been controlled, which may lead to the image inconsistency, affecting the integrity and accuracy. Besides, the images obtained include the facial skin condition and body parts with different diseases, which will affect the classification that focuses on facial skin conditions. Hence, the accuracy of classifying the images with acne will be affected because another disease might be considered as the detection and classified into the group where the images with acne.

### **2.1.4 Proposed Solutions**

The limitation of the previous study in [8] can be overcome by expanding the dataset images that include various ethnic in the dataset. In addition, choosing the dataset images that provide the people with different skin colors to help improve the accuracy in all types of skin classification.

Other than that, the limitation of the study in [9] can be overcome by finding the dataset that has a proper annotation to prevent the problem of the dataset that will cause the overall result to be affected. Therefore, chosen the dataset correctly is important to avoid unnecessary problems when training the data.

Lastly, the limitation in the study in [10] can be solved by the ongoing collection of consistent, collected images from a diverse range of people and with a range of social activities but the need to ensure the high quality of images is collected.

## 2.2 Product Review

### 2.2.1 Zenoti [11]

Zenoti has provided an all-in-one medspa software that includes the features needed for wellness medspa, skin clinic, weight loss clinic and so forth. The medspa software provided by Zenoti is to help the clinic increase profit by fulfilling the customer's special needs. The Zenoti Medspa software also reduces the admin's involvement, which needs to handle the overall data of the clinic's management. In addition, the software encourages paperless with all the features built into the software. The following features that provided by Zenoti:

- **Online Booking and appointment**

The appointment feature is operated 24/7, which can meet the target of a fully optimized appointment book. Booking recommendation is based on customer preference and the availability of the time slot. It also supports complex requirements where it can help to avoid schedule gaps. This can help the skin care centre fully utilize their time slot for the patient needed for a physical appointment for their consultation and treatment process. Figure 2.2.1.1 shows the form of online booking.

The image shows a screenshot of the Zenoti online booking and payment interface. The main window is titled "Selected Services" and contains a table with the following data:

Services	Provider	Details
Basic Manicure	Any	New York, USA Thu, Sep 8:15 AM - 9:00 AM

Below the table, there is a "Membership Applied" section with a green checkmark and the text "Annual 20% Discount Membership". The price is shown as ~~-\$25.00~~ \$0.00. There is also an "Add-ons" section with a radio button and a "Have any other preference" section with a text input field. A "Use a New Card" link is visible in the top right corner of the payment section.

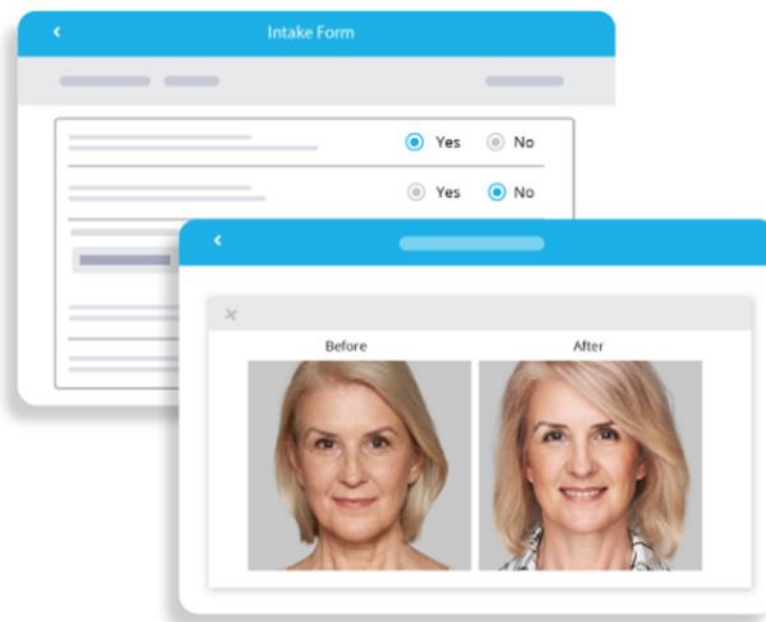
The payment section is titled "Payment" and shows a "VISA" card ending with "1234" with a green checkmark. There are two buttons: "Cancel" and "Confirm Booking".

*Figure 2.2.1.1: Form for online booking and appointment*

*Source: Adapted from [11].*

- **Digital Forms**

Digital forms help exclude the dependence on physical storage, which stores data in a centralized database. Therefore, the digital form can help to increase the efficiency of work where human error can be reduced because there is no paperwork needed for the staff in the centre to collect personal patient appointments by using their own devices to ensure safety. It also allows sharing of information across multiple locations because it is stored in centralized data. In addition, the data can be retrieved easily for important uses, such as diagnosing the treatment required for the patient and the consultation required for the patient to monitor the patient's skin condition. Figure 2.2.1.2 is the sample of the digital form.

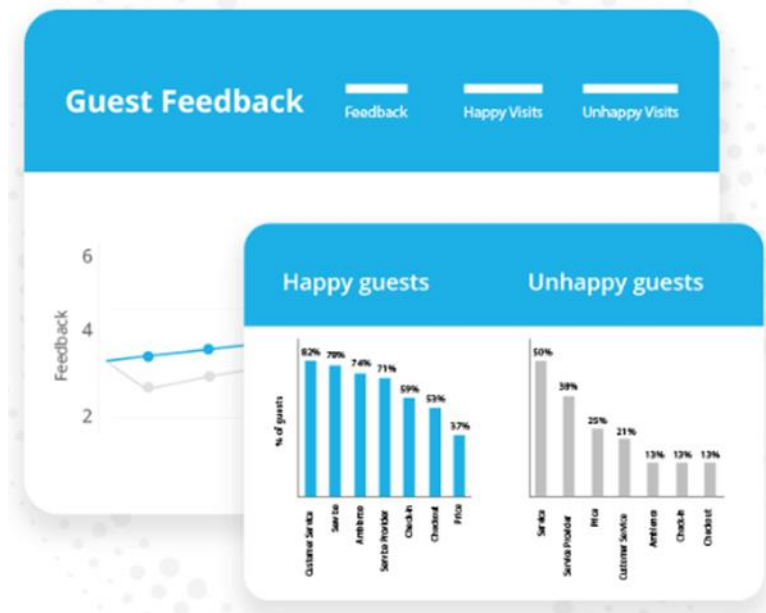


***Figure 2.2.1.2: Example of digital form for patient***

***Source: Adapted from [11].***

- **Report and Analytics**

The comprehensive and accurate report is useful for data-driven decisions and viewing the performance of the business. In addition, reports and analytics allow better business performance and payroll monitoring. The sample of the report and analytics features is as shown in Figure 2.2.1.3.



*Figure 2.2.1.3: General view for report and analytics features*

*Source: Adapted from [11].*

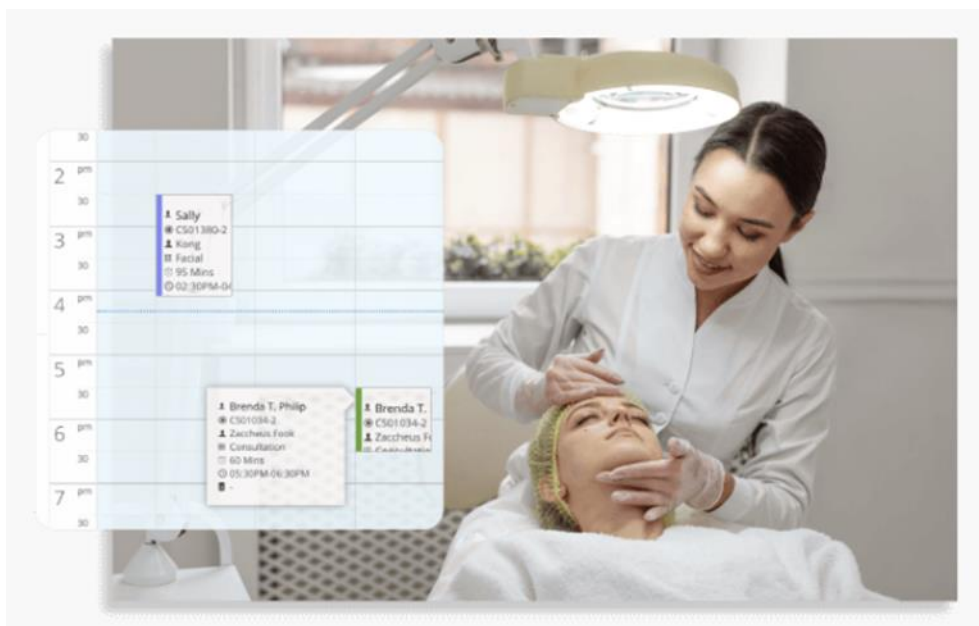


### 2.2.2 WESS [12]

WESS has provided the system with three important features that allow the Aesthetic Clinic to manage the clinic efficiently so that the first impression of the clinic is good and gain trust from the customer for their next appointment. The following explanation is the features provided by WESS to allow the management of the Aesthetic Clinic to look more professional:

- **Appointment management**

WESS has provided an appointment feature that can be easily managed, which helps avoid appointment mixing up. The appointment feature allows proper booking time slots with the complete details of the doctor and customer. The details of the appointment can be easily retrieved with the information required during the consultation, which allows the arrangement to be easily made. As a result, the doctor can review the patient's details before the consultation, give feedback on the treatment required, and suggest the next treatment needed for the patient. The appointment function is shown in Figure 2.2.2.1 as an example.



*Figure 2.2.2.1: Example of the appointment with detailed information  
Source: Adapted from [12].*

- **CRM software**

WESS has utilized the features of CRM to make it easy to track the customer's medical history and consultation notes to understand the customer and deliver service smoothly to the customer. CRM allows better interaction between the company and the customer because customer feedback and preference are collected through sales and marketing. Customers can provide their feedback to the company through the customer feedback platform, which can help the company enhance its services. SMS and email are embedded to keep in touch with the customer. Automated appointment reminder is provided by sending SMS automatically to the customer to reduce no-shows during the consultation hours. Figure 2.2.2.2 shows the automated appointment reminder in SMS.



***Figure 2.2.2.2: Example of automated appointment reminder***

***Source: Adapted from [12].***

### 2.2.3 Nookal [13]

Nookal is a software development company that has provided a comprehensive management system for the clinic since 2010, focusing on creating an easy-to-use cloud-based solution. It is a trusted skin clinic software that offers several systems features for clinic management. Nookal provides the following features:

- **Appointment and Bookings**

Nookal has provided the features of appointment and booking features which help the skin clinic to manage appointments easily with one-off reminders and team on-track function with the flexible calendar platform. The appointment booking consists of three categories which are group bookings. The system allows multiple customers to make the booking and purchase the sessions in bulk, saving time for the next appointment. Besides, a staff schedule is included to enable the arrangement of the staff for the consultation with the customer. Furthermore, the waiting list is provided to allow the customer to book for their preferred times. Therefore, the customer will be placed on the waiting list for their preferred times, allowing the consultation hours to be more convenient. An example of the appointment booking form is shown in Figure 2.2.3.1.

Appointment Details	
Current Date:	Thursday, 4th February 2021
Start Time:	5:00 PM
Duration:	30 minutes
Provider:	Alistair Mischke
Consultation:	Private - Extended Consultation
Type:	Private

Recurring Appointments Options	
Consultation:	Private - Extended Consultation
Duration:	30 minutes
Frequency:	Weekly
Repetitions:	5

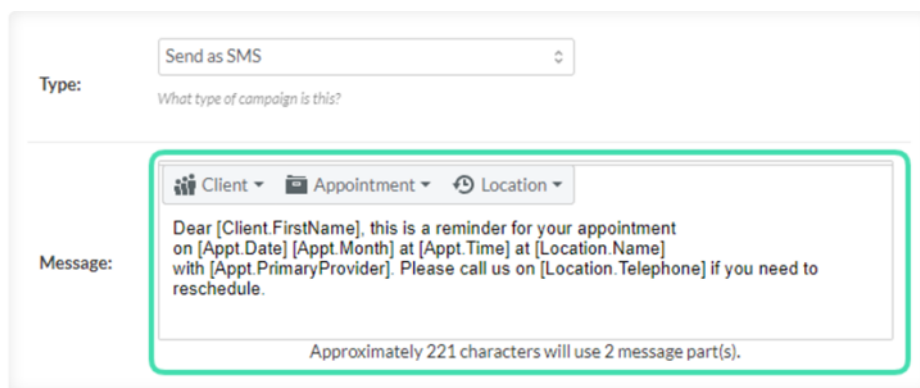
Availability Status						
Mon	Tue	Wed	Thur	Fri	Sat	Sun
X	X	✓	X	X	X	X

*Figure 2.2.3.1: Sample of appointment and bookings form*

*Source: Adapted from [13].*

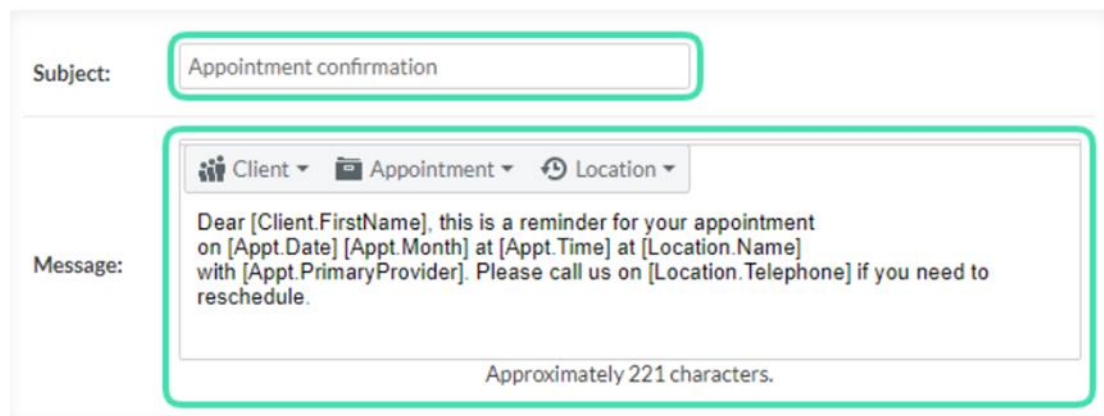
- **Automate Bookings and Reminders**

The customer is allowed to make an online booking, and the appointment reminder is automated, which helps prevent the appointment's cancellation. The reminder is to avoid the non-attendance of the customer. The reminder is sent through SMS and Email, which is automated to alert the customer of their appointment. The SMS services allow two-way communication where the customer can reply to the SMS to confirm or rearrange their appointment. Figure 2.2.3.2 shows the SMS reminder, while figure 2.2.3.3 shows the email reminder.



*Figure 2.2.3.2: Sample of appointment reminders using SMS*

*Source: Adapted from [13].*

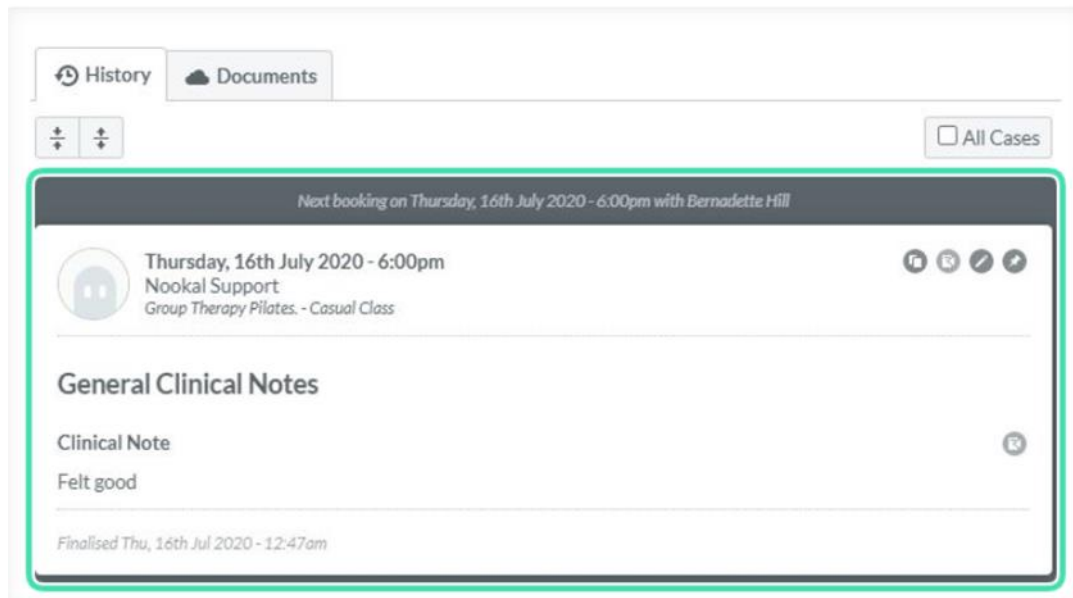


*Figure 2.2.3.3: Sample of appointment reminders using email*

*Source: Adapted from [13].*

- **Clinical Notes**

The note template is provided for the doctor to take the clinical notes during each consultation. It is a customized treatment note where the doctor can use the clinic note template to take down the important note for the needed treatment. In addition, shortcuts and symbols are provided in the template, which can be used easily to make a remark on the customer treatment result.



*Figure 2.2.3.4: Sample of clinical notes*

*Source: Adapted from [12].*

- **Document and File storage**

The record of the customer will be stored in the file storage so that the document can be accessed anytime, anywhere. The software also provides the features of file attachment, document uploading, and images to customer records, allowing the doctor to view the records anywhere. In addition, the cloud-based storage allows better management of the records for the customer because the history of the consultation and treatment can be easily retrieved from the system.

#### **2.2.4 Clinicminds [14]**

Clinicminds provides the software and electronic medical record system utilized by aesthetic clinics. Clinicminds has provided six essential features that can help run clinic management smoothly. The following are the features of aesthetic clinic management software that provided by Clinicminds:

- **Payments**

Provide invoices of treatments and product purchases by the customer, which can help the clinic management to prevent mistakes made on the invoice and provide paper-free for the clinic. Customers can use online transactions to make payments, and the invoices will send the revenue reports as a PDF file through e-mail to the customer.

- **Support Device**

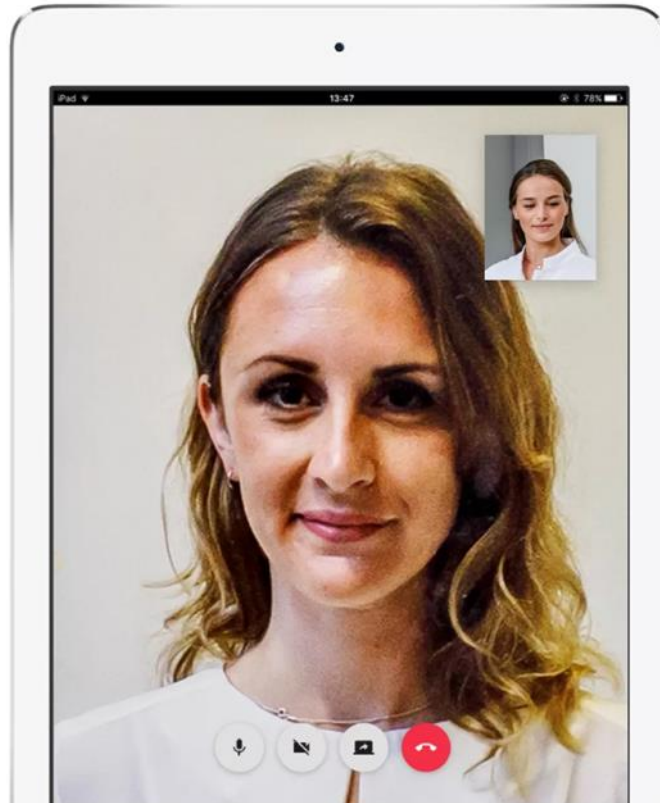
The application can support tablets, desktops, or laptops because no installation is required. The application is provided on a web-based platform. The mobile application can only view the ongoing schedule of appointment and patient records but need to connect with the iPhone app.

- **Software security**

HIPAA and GDPR manage software security. HIPAA is responsible for dealing with Protected Health Information (PHI), while GDPR is the standard for sensitive personal data. The security of the software complies with industry-standard security practices to ensure the data for the patient is securely stored in the cloud-based database.

- **Video appointments**

Online consultation using video meetings allows the customer to consult the specialist at any time and place. The meeting link will be sent through email and SMS, and the reminder will be sent to the customer a few minutes before the video appointment start. The video call feature is shown in Figure 2.2.4.1.

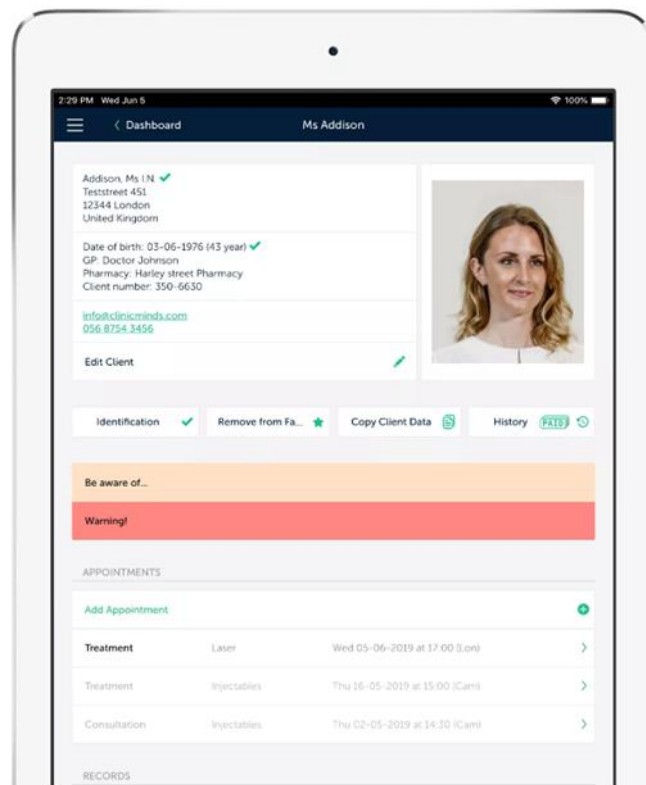


*Figure 2.2.4.1: Video call for online appointment meeting*

*Source: Adapted from [14].*

- **Treatment records**

Figure 2.2.4.2 shows the template of the customer treatment records with the customer's basic details and the treatment history, as well as the consultation history. Each customer's record is stored in a structured sequence with consultation history, treatment history, and treatment review. In addition, an overview of past and current treatments is also provided to make the comparison easy.



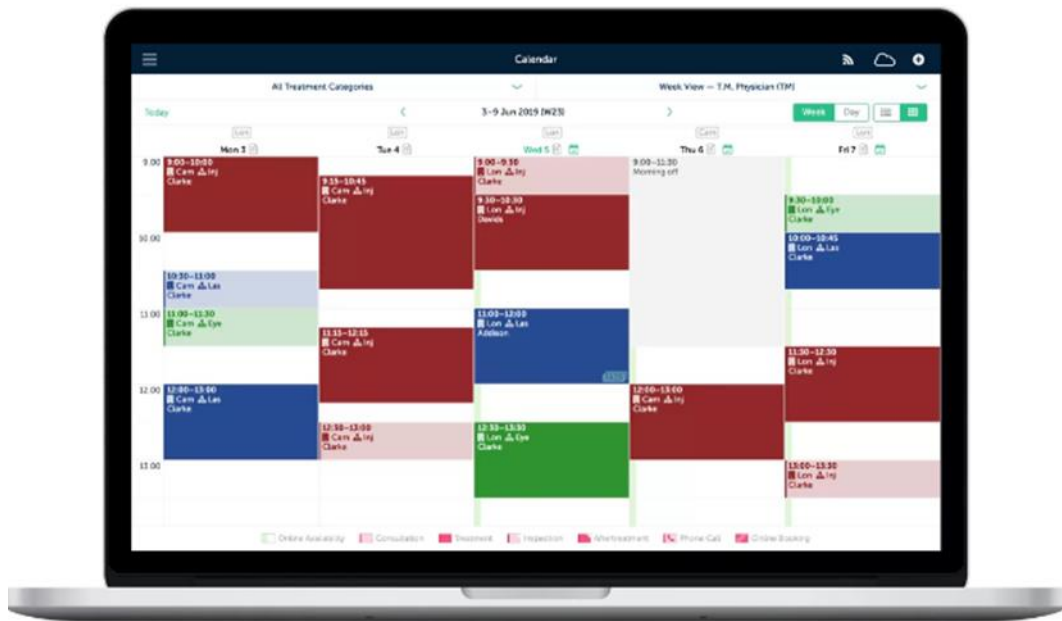
**Figure 2.2.4.2: Template for customer treatment records**

**Source: Adapted from [14].**



- **Scheduling**

Online booking appointment is provided based on availability and resources at the clinic, which can help prevent clashing in the appointment and improve customer experience. In addition, a reminder will be sent to the customer through emails or SMS texts to avoid no-shows and to save time. Figure 2.2.4.3 shows the online appointment booking with color-coded week and day.



**Figure 2.2.4.3: Time schedule for appointment booking**

*Source: Adapted from [14].*

### 2.3 Summarization of Product Review

	<b>Zenoti [11]</b>	<b>WESS [12]</b>	<b>Nookal [13]</b>	<b>Clinicminds [14]</b>	<b>Proposed System</b>
<b>Track Consultation History</b>	Yes. Report and Analysis can be generated.	Yes. CRM features to track the record of the customer.	Yes. The record is store in the file storage.	Yes. Treatment and consultation record for the patient	Yes. Treatment and consultation record stored in the system.
<b>Availability of SMS service</b>	No. Use digital form to get the information of the customer.	Yes. To keep in touch with the customer and appointment reminder.	Yes. Appointment reminder for the customer.	Yes. Appointment reminder to avoid time wasting on appointment day.	No. Use Firebase Cloud Messaging to notify the patient regarding their appointment.
<b>Online Consultation</b>	No. It provides online appointment and booking features	No. . It provides online appointment and booking features.	Yes. Telehealth and remote tools that will redirect to the third-party software.	Yes. Video appointment is provided for online consultation.	Yes. Video appointment for online consultation.
<b>Skin Type Classification</b>	No.	No.	No	No	Yes. Skin Classifier using TensorFlow Lite.
<b>Usable in mobile</b>	Yes. Applicable for both iOS and Android.	Not Applicable.	Not Applicable.	Yes. Applicable for iOS only.	Yes. Applicable for Android only.

*Table 2.3.1: Comparison of existing skin clinic management system*

Table 2.3.1 shows the comparison of all the existing skin clinic management systems that have been launched in the market and the proposed system. Based on the review from Table 2.3.1, "Zenoti", "WESS", "Nookal", and "Clinicminds" can track the record of the consultation for the customer for further review. "Zenoti" do not use any SMS services in the system to contact the customer. The digital form is used instead of SMS services to update the customer's skin condition. "WESS", "Nookal", and "Clinicminds" share the same features in which they use SMS service to communicate with the customer for a different purpose but mostly use it as a reminder feature to alert the customer regarding their appointment. While propose system is using Firebase Cloud Messaging to send the alert to the patient regarding their appointment date and when to join the video call consultation. "Zenoti" and "WESS" do not use online consultation methods. They only provide an online appointment. "Nookal" has the feature of video appointments, known as telehealth, but the feature is an additional feature for the system. "Zenoti" is the only system that supports both mobile operating systems, while the other two systems are only web-based, and "Clinicminds" only supports iOS. As for "Clinicminds", it is a better system than the other three as it implements video appointments that ease the online consultation process.

Another thing to highlight is that the system does not provide the function of skin classification. However, the proposed system provides the skin classification function in the mobile application, which help the skin center to evaluate the skin type of the patient quickly. Other than that, video appointment is implemented for online consultation, and Firebase Cloud Messaging are used to send the video appointment notification to the patient. Furthermore, the proposed system also allows the consultation history of the patient store in the Cloud Firestore Database and supports a mobile platform. Hence, the proposed system enhances the existing system with more features.

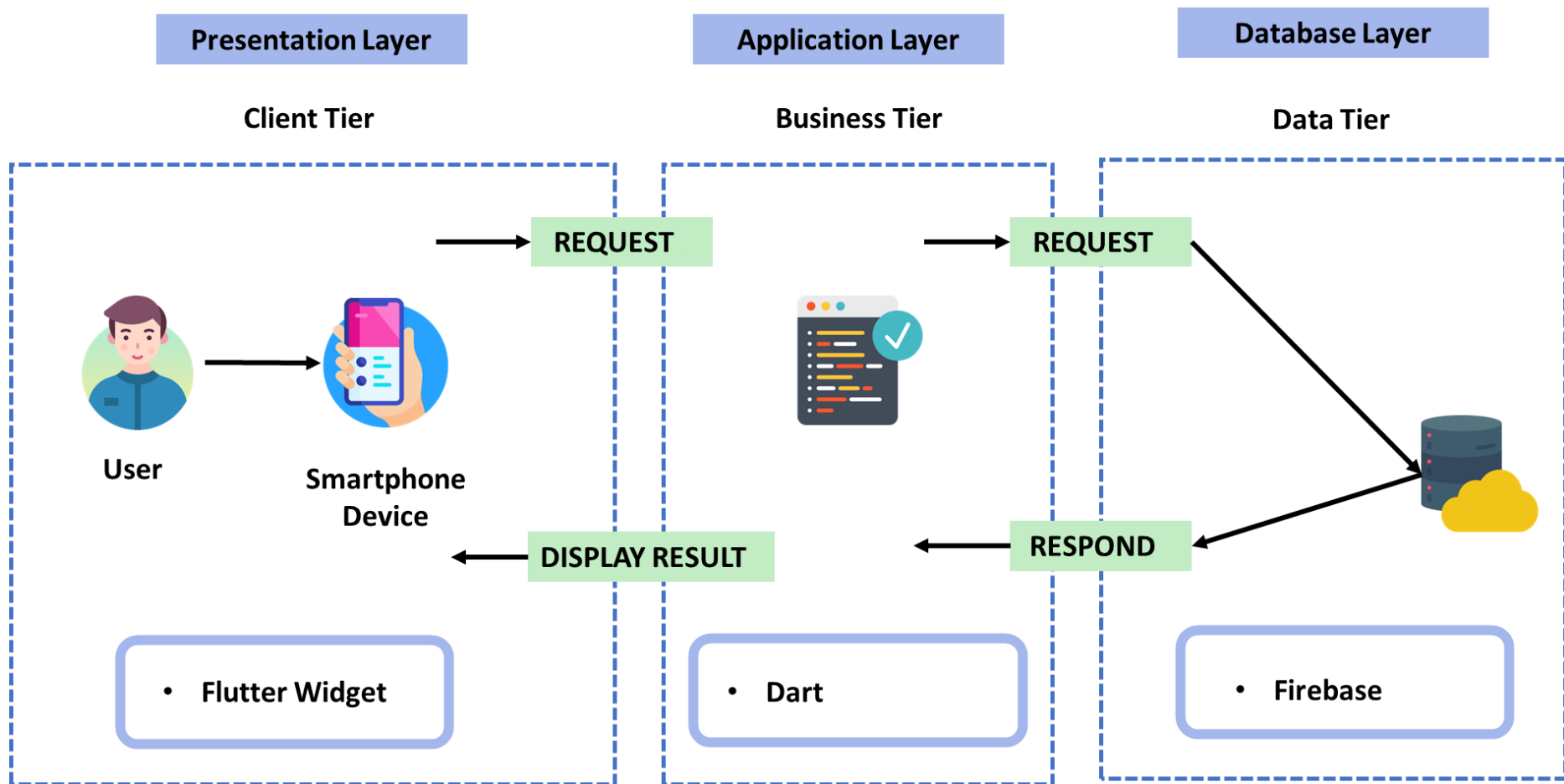
# Chapter 3

## System Methodology/Approach

In this chapter, the diagram shown in the subsection will be used to depict the flow of information across this project's system. The diagrams are aimed at enhancing the visualization of how the system is being built.

### 3.1 System Design Diagram

The frontend, also known as the **presentation layer**, is shown in Figure 3.1. This is the part of the system where the user interacts with the interface. For the **application layer**, it is the interface between the client and data tier. This layer helps to facilitate the communication between client and data tier. Furthermore, the application layer is responsible for the business logic of the mobile application which is displaying the content related to the business logic based on the user module as each module perform different functions. Finally, the **database layer** is where all the data is securely stored such as user email, password, details, report etc. All users are allowed to access the service system via smartphone. In the client tier, it will display the user interface for each of the functionalities to the users based on the functionalities that available in each module so that they will have better visualization on how to interact with the system to perform activities such as joining video call, view consultation report, book appointment, update profile details and so forth. When the user request something to be display on the screen by clicking button to trigger an event, the request will be sent to the application layers which act as a communication channel among the client and database to retrieve the information requested by the users. The request will then send to the database to retrieve the data. The information is then retrieved from the database and sending back the response to the application layer. The application layer will then sending back the requested information result to the client to be display on their screen.



*Figure 3.1: Overall system architecture design of facial and skincare service system*

### 3.1.1 System Architecture Diagram

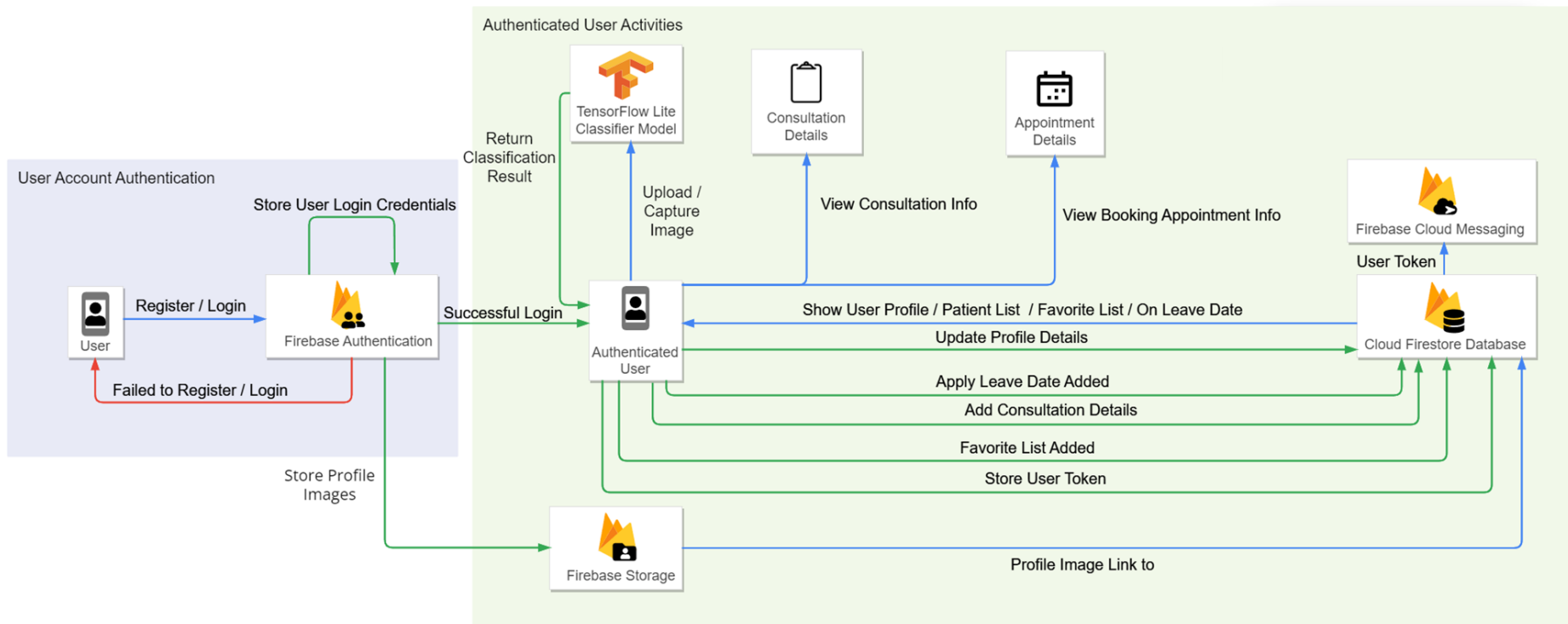
A system architectural diagram for the staff module may be seen in Figure 3.1.1.1. This model was created during the development of a mobile application and Figure 3.1.1.2 is a system architecture diagram for patient module in the development of mobile application. Firebase platform is the backend architecture that has been use to storing the data of the staff such as profile details, start date and end date for the leave application, unique token, favorite list., details of the patient, and the information in the consultation report. Other than that, Dart programming language is used for the Flutter mobile application development and Python programming language is used for TensorFlow and Keras in building the deep learning model. First, the arrows in the diagram is using three different colours that indicate the status of the information flow. The arrow with blue colours is representing the primary path of the system flow which is the steps that need to takes by the system to achieve the intended function. For the arrow with green colours, it is representing the user action is successfully perform while for the red colours is represent the unssuccessful action that has been done by the user.

Based on Firgure 3.1.1.1 and Figure 3.1.1.2 , both having the user account authentication which handle the authencation login information of the system which highlight using the blue box in User Account Authentication. Before the users are permitted to do the activity in the system, they are needed to log in to the system. The system will be prompted the user to create a new account if they are not exisiting user, which will then be managed by the Firebase Authentication backend service. If the user is able to successfully log in or register an account, the system will provide the user a notification indicating that they were successful. When the user is unable to successfully log in to their account or register for one, the error message will be shown. Once, the user is successfully register themselves with the system, the Firebase When a user registers for an account, their login credentials are saved in the authentication database. These credentials may be used in the future to authenticate the user when they login into the system. If the user is new to the system, Firebase Authentication will redirect the stored of the user details to the Cloud Firestore Database to store the information.

For Authenticated User Activities in Figure 3.1.1.1 and Figure 3.1.1.2, upon the user successfully login into the system, the user can perform different task in the system which include update profile details, add leave application date, add consultation details, view consultation information, book an appointment, view booking appointment information and

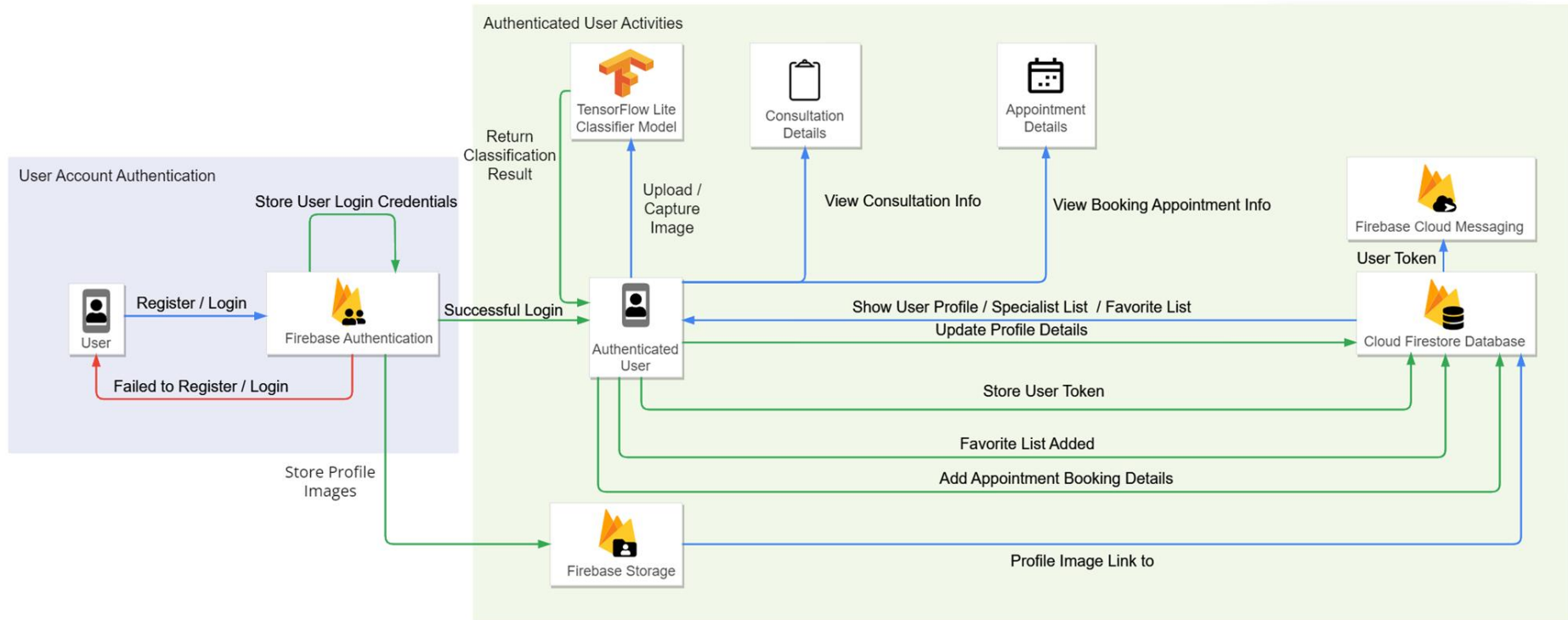
perform classification on the acne type. Upon user login, a unique token will be stored to the Cloud Firestore Database and this will be used by the Firebase Cloud Messaging in the future to send the notification. The Cloud Firestore database serves as a storage platform for data, which can be retrieved whenever a user requests to view their profile, patient list, favorite list, or the start and end dates of their leave applications. User who has been authenticated may alter their profile data, after which those details will be updated and stored in the appropriate location in the Cloud Firestore Database. Furthermore, users have the ability to edit the start and end dates of their leave application, which in turn updates the booking calendar on the patient's side. Once the staff add the start and end dates for the leave, the data is stored in the Cloud Firestore Database. This stored data is then retrieved and utilized in the booking calendar, which restricts users from booking any appointments during the designated period set by the staff.

Besides, the authenticated user in the staff module can also add the consultation details, this consultation details is only editable by the staff where the consultation details form will be available when the consultation is done with the patient. When editing consultation details, users are required to click the trigger event button to save any changes made to the consultation details into the Cloud Firestore Database. Additionally, users have the option to add or remove items from their favorite list, with the data being stored in the Cloud Firestore Database accordingly. If a profile is added to the favorite list, the relevant data is saved, but when it is removed, the associated data is deleted from the Cloud Firestore Database. Furthermore, classification model for acne type in TensorFlow lite is an offline module as a build-in feature in the system that allow the authenticated user to perform classification on image that help to classify the type of the acne. Users can choose to upload images to the system by selecting a photo from their gallery or capturing a new photo using their camera. The system will display the results accordingly. Furthermore, users can view their consultation details, which include a report of the consultation content and the patient's skin condition. In addition, appointment details such as the booking date, time, and mode of appointment will be recorded.



**Figure 3.1.1.1: Facial and Skincare Mobile Application Architecture Diagram for Staff Module**





**Figure 3.1.1.2: Facial and Skincare Mobile Application Architecture Diagram for Patient Module**



### 3.1.2.1 Use Case Description

#### Register User Account

Use Case Name: Sign Up User Account	ID: <u>101</u>	Importance Level: <u>High</u>
Primary Actor: New Staff / New Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: New Staff / New Patient - wants to register an account		
Brief Description: This use case describes how the new staff / new patient can register an account		
Trigger: New staff / new patient wants to register a new account to use the features in the system . Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. The new staff/ new patient clicks “Sign Up” to navigate to Sign Up page.</li> <li>2. The new staff/ new patient chooses the role and fill up all the required fields. S-1: Account Register is performed</li> <li>3. The new staff/ new patient is required to clicks “Register” to register for a new account after he/she has done entering the required fields.</li> <li>4. The system will validate the information entered.</li> <li>5. The system will save the valid information entered into the system database.</li> <li>6. The system will redirect new staff/ new patient to home page of the system.</li> </ol>		
Sub Flows: S-1: Account Register <ol style="list-style-type: none"> <li>1. The new staff/ new patient can choose their role to register an account for the system If new staff chooses the role as “Staff”</li> </ol>		

the S-2: Insert Staff Account Register Details is performed

If the new patient chooses the role as “User”

the S-3: Insert User Account Register Details is performed

#### S-2: Insert Staff Account Register Details

1. The new staff needs to insert his/her work ID.
2. The new staff needs to insert his/her email address.
3. The new staff needs to insert his/her password.
4. The new staff needs to insert his/her confirm password.

#### S-3 Insert User Account Register Details

1. The new patient needs to insert his/her email address.
2. The new patient needs to insert his/her password.
3. The new patient needs to insert his/her confirm password.

#### Alternate / Exceptional Flows:

S-2, 1a: The system display message “Work ID cannot be empty” if the work ID is empty.

S-2, 1b: The system display message “Please enter a valid work ID” if the work ID entered is invalid.

S-2, 2a: The system display message “Email cannot be empty” if the email address is empty.

S-2, 2b: The system display message “Please enter a valid email address” if the email address entered is invalid.

S-2, 3a: The system display message “Password cannot be empty” if the password is empty.

S-2, 3b: The system display message “Please enter valid password min. 6 character” if the password entered is invalid.

S-2, 4a: The system display message “Password did not match” if the confirm password entered is not the same with the password.

S-3, 1a: The system display message “Email cannot be empty” if the email address is empty.

S-3, 1b: The system display message “Please enter a valid email address” if the email address entered is invalid.

S-3, 2a: The system display message “Password cannot be empty” if the password is empty.

S-3, 2b: The system display message “Please enter valid password min. 6 character” if the password entered is invalid.

S-3, 3a: The system display message “Password did not match” if the confirm password entered is not the same with the password.

4a: The system display message “Unsuccessful Registration” if the email address entered has exist in the database.

## Login User Account

Use Case Name: Login User Account	ID: <u>102</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to login to the system		
Brief Description: This use case describes how the existing staff/ existing patient login to the system		
Trigger: Existing Staff/ Existing Patient wants to login to the existing account to use the features in the system . Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient is on the Login Page. If the existing staff/ existing enter his/her login credentials, the S-1: Account Login is performed 2. The existing staff/ existing patient is required to clicks “Sign In” to sign in for his/her account after he/she has done entering the required fields. 3. The system will validate the entered login credentials. 4. The system will redirect existing staff/ existing patient to home page of the system.		
Sub Flows: S-1: Account Login 1. The existing staff/ existing patient needs to insert his/her email address. 2. The existing staff/ existing patient needs to insert his/her password.		
Alternate / Exceptional Flows: S-1: 1a: The system display message “Invalid email or password. Please try again.” if the email address and password entered has does not exist in the database.		

3a: The system display message “Unsuccessful Login” if the email address or password entered has does not exist in the database.

## Sign Out User Account

Use Case Name: Sign Out User Account	ID: <u>103</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to sign out from the system		
Brief Description: This use case describes how the existing staff/ existing patient to exit the system through sign out feature		
Trigger: Existing Staff/ Existing Patient wants to exit the system by logging out his/her account. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient click on “Profile” on bottom navigation button. 2. The existing staff/ existing patient click on the “Logout” on the top right corner of the page. 3. The system will redirect existing staff/ existing patient to Login page of the system.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		



## Join Video Meeting

Use Case Name: Join Video Meeting	ID: <u>104</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to join online video call meeting		
Brief Description: This use case describes how the existing staff/ existing patient joining the online video call meeting to conduct or attend the online consultation session		
Trigger: Existing Staff wants to conduct an online consultation with patient and Existing Patient wants to attend the online consultation to meet his/her consultant. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient click on “Join Call” on home page at the “Appointment Today” section. If the existing staff/ existing patient is first time joining call, the S-1: Device Permission is performed		
Sub Flows: S-1: Device Permission 1. The system will pop up a dialog box with two options (“Allow” and “Deny”) to gain permission access to device camera and microphone 2. The existing staff/ existing patient choose his/her option 3. The services system validates the option If the existing staff/ existing patient is “Allowed” , redirect the existing staff/ existing patient to video call room. If the existing staff/ existing patient is “Denied”, redirect the existing staff/ existing patient back to the homepage.		

Alternate / Exceptional Flows: Not Applicable

## Manage Profile

Use Case Name: Manage Profile	ID: <u>105</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to manage the user account profile		
Brief Description: This use case describes how the existing staff/ existing patient manage his/her user account profile		
Trigger: Existing Staff/ Existing Patient wants to manage his/her user account profile in the system. Type: External		
Relationships: Association: - Include: Edit Profile Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient is logging into the system successfully. 2. The existing staff/ existing patient is required to clicks “Profile” to navigate to Profile page. 3. If the existing staff/ existing patient clicks the profile icon Execute the Edit Profile user case.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## View Consultation Report [Existing Patient]

Use Case Name: View Consultation Report	ID: <u>106</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to view the consultation report		
Brief Description: This use case describes how the existing patient can view the consultation report for the previous consultation details		
Trigger: Existing Patient wants to view his/her previous record for consultation in system . Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing patient is logging into the system successfully. 2. The existing patient can choose “Schedule page” or “Profile page” to view the consultation report. If the existing patient choose “Schedule page”, the S-1: View Report on Schedule Page is performed If the existing patient choose “Profile Page”, the S-2: View Report on Profile Page is performed		
Sub Flows: S-1: View Report on Schedule Page 1. The existing patient is required to clicks “Schedule” on the bottom navigation button. 2. The system will navigate the current page to the Schedule page 3. The existing patient is required to clicks “complete” on the horizontal navigational tab.		

4. The existing patient is required to clicks “View Consultation Details” on selected appointment card.
5. The system will navigate the current page to the Consultation Report page

S-2: View Report on Profile Page

1. The existing patient is required to clicks “Profile” on the bottom navigation button.
2. The system will navigate the current page to the Profile page.
3. The existing patient is required to clicks “View Latest Report” which is below the displayed profile details.
4. The system will navigate the current page to the Consultation Report page

Alternate / Exceptional Flows: Not Applicable

## View Consultation Report [Existing Staff]

Use Case Name: View Consultation Report	ID: <u>107</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to view the consultation report		
Brief Description: This use case describes how the existing staff can view the consultation report for the previous consultation details		
Trigger: Existing Staff wants to view his/her patient’s previous record for consultation in system . Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff is logging into the system successfully. 2. The existing staff can choose “Patient List page” or “Schedule page” to view the consultation report. If the existing staff choose “Patient List page”, the S-1: View Report on Patient List page is performed If the existing staff choose “Schedule page”, the S-2: View Report on Schedule Page is performed		
Sub Flows: S-1: View Report on Patient List page 1. The existing staff is required to clicks “Patient” on the bottom navigation button. 2. The system will redirect the existing staff to the Patient List page 3. The existing staff is required to click on the selected patient profile from the patient list. 4. The system will redirect the existing staff to the Patient profile page.		

5. The existing staff is required to clicks “Consultation History” which located at the bottom of the Patient profile page.

#### S-2: View Report on Schedule Page

1. The existing staff is required to clicks “Schedule” on the bottom navigation button.
2. The system will redirect the existing staff to the Schedule page.
3. The existing staff can choose “upcoming” to view the consultation report
4. The existing staff is required to clicks “View Consultation Details” on selected appointment card.
5. The system will redirect the existing staff to the Consultation Report page

Alternate / Exceptional Flows: Not Applicable

## Manage Favourite List

Use Case Name: Manage Favourite List	ID: <u>108</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to manage favourite list		
Brief Description: This use case describes how the existing staff/ existing patient can manage favourite list that allow his/ her to find desire profile easily in the system		
Trigger: Existing Staff / Existing Patient wants to manage a favourite list that he/she can find the desire profile easily in the system. Type: External		
Relationships: Association: - Include: Add Favourite, Delete Favourite Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient is logging into the system successfully. 2. The system identifies the role of the user in the system If existing staff is logging into the system, the S-1: Staff Manage Favourite List is performed. If existing patient is logging into the system, the S-2: Patient Manage Favourite List is performed.		
Sub Flows: S-1: Staff Manage Favourite List 1. The existing staff is required to clicks “Patient” on the bottom navigation button. 2. The system will redirect the existing staff to the Patient List page 3. The existing staff is required to click on the selected patient profile from the patient list. 4. The system will redirect the existing staff to the Patient profile page.		



5. If the existing staff clicks the “heart” icon

If the “heart” icon is filled with red colour

Execute the Add Favourite use case

If the “heart” icon is not filled with red colour

Execute the Delete Favourite use case

#### S-2: Patient Manage Favourite List

1. The existing patient is required to clicks “Specialist” on the bottom navigation button.
2. The system will redirect the existing patient to the Specialist page
3. The existing patient is required to click on the selected specialist profile from the specialist’s list.
4. The system will redirect the existing patient to the Specialist profile page.
5. If the existing staff clicks the “heart” icon
  - If the “heart” icon is filled with red colour
    - Execute the Add Favourite use case
  - If the “heart” icon is not filled with red colour
    - Execute the Delete Favourite use case

Alternate / Exceptional Flows: Not Applicable

## Classify Acne Type

Use Case Name: Classify Acne Type	ID: <u>109</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to classify the acne type		
Brief Description: This use case describes how the existing staff/ existing patient can classify the acne type by using the acne classification feature in the system		
Trigger: Existing Staff/ Existing Patient wants to classify the acne type to identify the acne category. Type: External		
Relationships: Association: - Include: Upload Image, Take Photo Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient is logging into the system successfully. 2. The system identifies the role of the user in the system If existing staff is logging into the system, the S-1: Staff Classify Acne Type is performed. If existing patient is logging into the system, the S-2: Patient Classify Acne Type is performed.		
Sub Flows: S-1: Staff Classify Acne Type 1. The existing staff is required to clicks on the “AI Scan” on the homepage when the “Appointment Today” is not empty and the appointment mode is physical. 2. The system will redirect the existing staff to the AI Scan page. 3. The existing staff can choose either take photo or upload image to use acne classification feature		

If the existing staff choose take photo,

Execute the Take Photo use case.

If the existing staff choose upload image,

Execute the Upload Image use case.

#### S-2: Patient Classify Acne Type

1. The existing patient is required to clicks “Profile” on the bottom navigation button.
2. The system will redirect the existing patient to the Profile page.
3. The existing patient is required to clicks on the “AI Scan” button
4. The system will redirect the existing patient to the AI Scan page.
5. The existing patient can choose either take photo or upload image to use acne classification feature

If the existing patient choose take photo,

Execute the Take Photo use case.

If the existing patient choose upload image,

Execute the Upload Image use case.

Alternate / Exceptional Flows: Not Applicable

### View Specialist List

Use Case Name: View Specialist List	ID: <u>110</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to view the list of specialists displayed in the system		
Brief Description: This use case describes how the existing patient can view the displayed list of the specialist in specialist page in the system		
Trigger: Existing Patient wants to view the list of the specialist in the system. Type: External		
Relationships: Association: - Include: - Extend: View Specialist Details Generalization: -		
Normal Flow of Events: 1. The existing patient is logging into the system successfully. 2. The existing patient is required to clicks “Specialist” on the bottom navigation button. 3. The system will redirect the existing patient to the Specialist page. 4. The existing patient can scroll over the list of specialists in Specialist page. If the existing patient wants to view the details profile of the specialist Execute the View Specialist Details use case.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

### Manage Appointment

Use Case Name: Manage Appointment	ID: <u>111</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to make, change or cancel an appointment		
Brief Description: This use case describes how the existing patient make an appointment as well as changing or cancelling an appointment for previously seen patient		
Trigger: Existing Patient wants to make a new appointment, make changes, or cancel an existing appointment. Type: External		
Relationships: Association: - Include: Book New Appointment, Reschedule Appointment, Cancel Appointment Extend: - Generalization: -		
Normal Flow of Events: 1. The existing patient is logging into the system successfully. 2. If the existing patient wants to make new appointment Execute the Book New Appointment use case. 3. If the existing patient wants to reschedule an existing appointment Execute the Reschedule Appointment use case. 4. If the existing patient wants to cancel an existing appointment Execute the Cancel Appointment use case.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

### View Patient List

Use Case Name: View Patient List	ID: <u>112</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to view the list of patients displayed in the system		
Brief Description: This use case describes how the existing staff can view the displayed list of the patient in patients page in the system		
Trigger: Existing Staff wants to view the list of the patient in the system. Type: External		
Relationships: Association: - Include: - Extend: View Patient Details Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"><li>1. The existing staff is logging into the system successfully.</li><li>2. The existing staff is required to clicks “Patient” on the bottom navigation button.</li><li>3. The system will redirect the existing staff to the Patient page.</li><li>4. The existing staff can scroll over the list of patients in Patient page.</li><li>5. If the existing staff wants to view the details profile of the patient Execute the View Patient Details use case</li></ol>		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## Manage Consultation Details

Use Case Name: Manage Consultation Details	ID: <u>113</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to manage consultation details		
Brief Description: This use case describes how the existing staff can manage his/her patient’s consultation details		
Trigger: Existing Staff wants to manage the patient’s consultation details. Type: External		
Relationships: Association: - Include: Edit Consultation Details Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff is logging into the system successfully. 2. The existing staff can manage the consultation details in homepage or schedule page. If the existing staff choose to manage consultation details in homepage S-1: Manage Consultation Details in Homepage If the existing staff choose to manage consultation details in schedule page S-2: Manage Consultation Details in Schedule page		
Sub Flows: S-1: Manage Consultation Details in Homepage 1. The existing staff is required to clicks “complete” after completing the consultation session. 2. The system will redirect the existing staff to the Consultation Details page 3. If the existing staff wants to edit the consultation details Execute the Edit Consultation Details use case  S-2: Manage Consultation Details in Schedule page		

1. The existing staff is required to clicks “Schedule” on bottom navigation button.
2. The system will redirect the existing staff to the Schedule page.
3. The existing staff is required to clicks “complete” on the horizontal navigational bar.
4. The system will redirect the existing staff to the Consultation Details page
5. If the existing staff wants to edit the consultation details

Execute the Edit Consultation Details use case

Alternate / Exceptional Flows: Not Applicable



## Manage Appointment Date

Use Case Name: Manage Appointment Date	ID: <u>114</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to manage appointment date		
Brief Description: This use case describes how the existing staff can manage his/her appointment date		
Trigger: Existing Staff wants to manage the appointment date in his/her profile. Type: External		
Relationships: Association: - Include: Edit On Leave Date Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff is logging into the system successfully. 2. The existing staff can manage the appointment date in homepage. If the existing staff wants to manage appointment date Execute Edit On Leave Date use case		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## Edit Profile

Use Case Name: Edit Profile	ID: <u>201</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to edit the user account profile		
Brief Description: This use case describes how the existing staff/ existing patient edit his/her user account profile		
Trigger: Existing Staff/ Existing Patient wants to edit his/her user account profile in the system. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing staff/ existing patient will redirect to the Edit Profile page. 2. The system will identify the role of the user in the system. If the role of the user is Staff, the S-1: Edit Staff Profile is performed. If the role of the user is User, the S-2: Edit Patient Profile is performed. 3. The existing staff/ existing patient is required to click save button to accept changes make to the profile. 4. The system will validate the entered information. 5. The system will update the valid information to the system database.		
Sub Flows: S-1: Edit Staff Profile 1. If the existing staff wants to edit the profile picture, the S-3: Edit Profile Picture is performed.		

2. If the existing staff wants to edit the name  
the S-4: Edit Name is performed.
3. If the existing staff wants to edit gender,  
the S-5: Edit Gender is performed.
4. If the existing staff wants to edit Job Position,  
the S-6: Edit Job Position is performed.
5. If the existing staff wants to edit Qualification,  
the S-7: Edit Qualification is performed.
6. If the existing staff wants to edit Experience,  
the S-8: Edit Experience is performed.
7. If the existing staff wants to edit Expertise Area,  
the S-9: Edit Expertise Area is performed.
8. If the existing staff wants to edit Training Place,  
the S-10: Edit Training Place is performed.
9. If the existing staff wants to edit Phone Number,  
the S-11: Edit Phone Number is performed.

#### S-2: Edit Patient Profile

1. If the existing patient wants to edit the profile picture,  
the S-3: Edit Profile Picture is performed.
2. If the existing patient wants to edit gender,  
the S-5: Edit Gender is performed.
3. If the existing patient wants to edit date of birth,  
the S-12: Edit Date of Birth is performed.
4. If the existing staff wants to edit Phone Number,  
the S-11: Edit Phone Number is performed.

#### S-3: Edit Profile Picture

1. The existing staff/ existing patient needs to click on the profile image.
2. The system will pop up a dialog box with two option “Allow” or “Deny” to gain permission access to device gallery.  
If the existing staff/ existing patient choose “Allow”,  
the S-13: Access Device Gallery is performed.  
If the existing staff/ existing patient choose “Deny”,  
the S-14: Deny Access to Device Gallery is performed.

S-4: Edit Name

1. The existing staff/ existing patient need insert the new name.

S-5: Edit Gender

1. The existing staff/ existing patient need insert the gender.

S-6: Edit Job Position

1. The existing staff need insert the new job position.

S-7: Edit Qualification

1. The existing staff need insert the new qualification.

S-8: Edit Experience

2. The existing staff need insert the new experience.

S-9: Edit Expertise Area

1. The existing staff need insert the new expertise area.

S-10: Edit Training Place

1. The existing staff need insert the new training place.

S-11: Edit Phone Number

1. The existing staff/ existing patient need insert the new phone number.

S-12: Edit Date of Birth

1. The existing patient need insert the date of birth.

S-13: Access Device Gallery

1. The system will navigate the current page to the device gallery.
2. The existing staff/ existing patient need to select a photo from the gallery.
3. The system will redirect the existing staff/ existing patient back to the Edit Profile page if he/she has done picking a new profile picture from the gallery.

S-14: Deny Access to Device Gallery

1. The system will close the dialog box and return to the Edit Profile page.

Alternate / Exceptional Flows:

S-4, 1a: The system display message “Please enter a name” if the name is empty.

S-7, 1a: The system display message “Please enter a valid qualification” if the qualification is empty.

S-8, 1a: The system display message “Please enter a valid experience” if the experience is empty.

S-10, 1a: The system display message “Please enter a valid training place” if the training place is empty.

S-11, 1a: The system display message “Phone number cannot be empty” if the phone number is empty.

S-11, 1b: The system display message “Please enter a valid phone number” if the phone number is not a valid phone number.

S-13, 2a: The system display message “No file selected” if the no photo selected.

## Add Favourite

Use Case Name: Add Favourite	ID: <u>202</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to delete favourite profile		
Brief Description: This use case describes how the existing staff/ existing patient can remove favourite profile from the favourite list in the system		
Trigger: Existing Staff / Existing Patient wants to remove the favourite profile from the favourite list in the system. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The system removes the filled red colour on the “heart” icon. 2. The system deletes the data of the profile details from the database for favourite list.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## Delete Favorite

Use Case Name: Delete Favourite	ID: <u>203</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to add new favourite profile		
Brief Description: This use case describes how the existing staff/ existing patient can add a new favourite profile to the favourite list in the system		
Trigger: Existing Staff / Existing Patient wants to add the favourite profile to the favourite list in the system. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 3. The system filled red colour on the “heart” icon. 4. The system adds the data of the profile details from the database for favourite list.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

### View Specialist Details

Use Case Name: View Specialist Details	ID: <u>204</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to read or view the details on the selected specialist		
Brief Description: This use case describes how the existing patient can read or view the selected specialist profile details in the specialist details page		
Trigger: Existing Patient wants to read or view the specialist details. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"><li>1. The existing patient click on the specialist in the list of the specialist in the specialist page.</li><li>2. The system will redirect the existing patient to the Specialist Details page.</li><li>3. The existing patient can read the details. the S-1: Read Specialist Details is performed.</li></ol>		
Sub Flows: S-1: Read Specialist Details <ol style="list-style-type: none"><li>1. The existing patient can view the name.</li><li>2. The existing patient can view the qualification.</li><li>3. The existing patient can view the company.</li><li>4. The existing patient can view the expertise area.</li><li>5. The existing patient can view the experience.</li><li>6. The existing patient can view the short description about the specialist.</li></ol>		
Alternate / Exceptional Flows: Not Applicable		



### View Patient Details

Use Case Name: View Patient List	ID: <u>205</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to view the list of patients displayed in the system		
Brief Description: This use case describes how the existing staff can view the displayed list of the patient in patients page in the system		
Trigger: Existing Staff wants to view the list of the patient in the system. Type: External		
Relationships: Association: - Include: - Extend: View Patient Details Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"><li>1. The existing staff click on the patient in the list of the patient in the patient page.</li><li>2. The system will redirect the existing staff to the Patient Details page.</li><li>3. The existing staff can read the details. the S-1: Read Patient Details is performed.</li></ol>		
Sub Flows: S-1: Read Patient Details <ol style="list-style-type: none"><li>1. The existing staff can view the name.</li><li>2. The existing staff can view the gender.</li><li>3. The existing staff can view the age.</li><li>4. The existing staff can view the category of acne type.</li><li>5. The existing staff can view the date of birth.</li><li>6. The existing staff can view the phone number.</li><li>7. The existing staff can view the email.</li></ol>		
Alternate / Exceptional Flows: Not Applicable		

### Book New Appointment

Use Case Name: Book New Appointment	ID: <u>206</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to make an appointment		
Brief Description: This use case describes how the existing patient make an appointment		
Trigger: Existing Patient wants to make a new appointment Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"><li>1. The existing patient is required to clicks “Specialist” on the bottom navigation button.</li><li>2. The system will redirect the existing patient to the Specialist page.</li><li>3. The existing patient is required to select and click a preferred specialist.</li><li>4. The system will redirect the existing patient to the Specialist Details page.</li><li>5. The existing patient needs to click “Book Appointment” in the Specialist Details page.</li><li>6. The system will navigate the current page to Appointment page.</li><li>7. The existing patient needs to select an appointment date.</li><li>8. The existing patient needs to select an appointment time.</li><li>9. The existing patient needs to select an appointment mode.</li><li>10. The existing patient needs to click on the “Make Appointment” button to complete the process.</li><li>11. The system will navigate the current page to the Confirmation Booking page</li><li>12. The system will validate the appointment details.  If the appointment details are valid,  the S-1: Valid appointment details is performed.</li></ol>		

<p>If the appointment details are invalid,  the S-2: Invalid appointment details is performed.</p> <p>13. The existing patient is required to click the “Back to Home Page” to complete the booking appointment process.</p> <p>14. The system will redirect the existing patient to the Specialist page.</p>
<p>Sub Flows:</p> <p>S-1: Valid appointment details</p> <ol style="list-style-type: none"> <li>1. The system will display successfully booked an appointment message.</li> <li>2. The system will add the booking details to the database of the system.</li> </ol> <p>S-2: Invalid appointment details</p> <ol style="list-style-type: none"> <li>1. The system will display unsuccessfully booked an appointment message.</li> </ol>
<p>Alternate / Exceptional Flows: Not Applicable</p>

## Reschedule Appointment

Use Case Name: Reschedule Appointment	ID: <u>207</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to make changes on an existing appointment		
Brief Description: This use case describes how the existing patient make changes on an existing appointment		
Trigger: Existing Patient wants to reschedule the existing appointment Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. The existing patient is required to clicks “Schedule” on the bottom navigation button.</li> <li>2. The system will redirect the existing patient to the Schedule page.</li> <li>3. The existing patient can choose to edit the appointment in “upcoming” or “cancel”  If the existing patient choose to edit the appointment in upcoming,  the S-1: Edit Upcoming Appointment is performed.  If the existing patient choose to edit the appointment in cancel,  the S-2: Edit Cancel Appointment is performed.</li> </ol>		
Sub Flows: S-1: Edit Upcoming Appointment <ol style="list-style-type: none"> <li>1. The existing patient is required to click “Reschedule” on the selected appointment card.</li> <li>2. The system will navigate the current page to Appointment page.</li> <li>3. The existing patient needs to select new appointment date.</li> <li>4. The existing patient needs to select new appointment time.</li> <li>5. The existing patient needs to select new appointment mode.</li> </ol>		

6. The existing patient needs to click on the “Reschedule Appointment” button to complete the process.
7. The system will navigate the current page to the Confirmation Booking page
8. The system will validate the appointment details.
  - If the appointment details are valid,
    - the S-3: Valid appointment details is performed.
  - If the appointment details are invalid,
    - the S-4: Invalid appointment details is performed.
9. The existing patient is required to click the “Back to Home Page” to complete the booking appointment process.
10. The system will redirect the existing patient to the Specialist page.

#### S-2: Edit Cancel Appointment

1. The existing patient is required to click “cancel” on the horizontal navigational bar.
2. The system will redirect the existing patient to the list of cancels appointment in Schedule page.
3. The existing patient is required to click “Reschedule” on the selected appointment card.
4. The system will navigate the current page to Appointment page.
5. The existing patient needs to select new appointment date.
6. The existing patient needs to select new appointment time.
7. The existing patient needs to select new appointment mode.
8. The existing patient needs to click on the “Reschedule Appointment” button to complete the process.
9. The system will navigate the current page to the Confirmation Booking page
10. The system will validate the appointment details.
  - If the appointment details are valid,
    - the S-3: Valid appointment details is performed.
  - If the appointment details are invalid,
    - the S-4: Invalid appointment details is performed.
11. The existing patient is required to click the “Back to Home Page” to complete the booking appointment process.

12. The system will redirect the existing patient to the Specialist page.

S-3: Valid appointment details

1. The system will display successfully booked an appointment message.
2. The system will update the new booking details to the database of the system.

S-4: Invalid appointment details

1. The system will display unsuccessfully booked an appointment message.

Alternate / Exceptional Flows: Not Applicable

## Cancel Appointment

Use Case Name: Cancel Appointment	ID: <u>208</u>	Importance Level: <u>High</u>
Primary Actor: Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Patient – wants to cancel an existing appointment		
Brief Description: This use case describes how the existing patient cancel on an existing appointment		
Trigger: Existing Patient wants to cancel the existing appointment Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The existing patient can choose to cancel the existing appointment in “Homepage” or “Schedule page.” If the existing patient choose to cancel existing appointment in “Homepage” the S-1: Cancel Homepage Appointment is performed. If the existing patient choose to cancel existing appointment in “Schedule page” the S-2: Cancel Upcoming Appointment is performed.		
Sub Flows: S-1: Cancel Homepage Appointment 1. The existing patient is required to click “Cancel” on the appointment card. 2. The system updates the status of the appointment to “cancel” in the database.  S-2: Cancel Upcoming Appointment 1. The existing patient is required to click “Schedule” on the bottom navigation button. 2. The system will redirect the existing patient to Schedule page.		

3. The existing patient is required to click “Cancel” on the selected appointment card.
4. The system updates the status of the appointment to “cancel” in the database.

Alternate / Exceptional Flows: Not Applicable



### Edit Consultation Details

Use Case Name: Edit Consultation Details	ID: <u>209</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to edit consultation details		
Brief Description: This use case describes how the existing staff can edit his/her patient’s consultation details		
Trigger: Existing Staff wants to edit the patient’s consultation details. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"> <li>1. The existing staff need to select the acne location.</li> <li>2. The existing staff need to select the skin condition.</li> <li>3. The existing staff need to select the product used.</li> <li>4. The existing staff need to select the consultation content.</li> <li>5. The existing staff need to insert the important note.</li> <li>6. The existing staff is required to click the save button to submit the details of the consultation.</li> <li>7. The system will add the details of the consultation to the database of the system.</li> </ol>		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## Edit On Leave Date

Use Case Name: Edit On Leave Date	ID: <u>210</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff – wants to edit on leave date		
Brief Description: This use case describes how the existing staff can edit his/her on leave start date and end date to manage the appointment date.		
Trigger: Existing Staff wants to edit his/her on leave date. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: <ol style="list-style-type: none"><li>1. The existing staff need to select the start date.</li><li>2. The existing staff is required to click “OK” after choosing the date.</li><li>3. The system will save the start date into the database of the system.</li><li>4. The existing staff need to select the end date.</li><li>5. The existing staff is required to click “OK” after choosing the date.</li><li>6. The system will save the end date into the database of the system.</li></ol>		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

## Upload Image

Use Case Name: Upload Image	ID: <u>211</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to use the acne classification feature by uploading an image		
Brief Description: This use case describes how the existing staff/ existing patient can upload an image from the device gallery to use the acne classification feature		
Trigger: Existing Staff/ Existing Patient wants to use the acne classification feature by uploading an image. Type: External		
Relationships: Association: - Include: Image Classification Result Extend: - Generalization: -		
Normal Flow of Events: 1. The system will pop up a dialog box with two option “Allow” or “Deny” to gain permission access to device gallery. If the existing staff/ existing patient choose “Allow”, the S-1: Access Device Gallery is performed. If the existing staff/ existing patient choose “Deny”, the S-2: Deny Access to Device Gallery is performed.		
Sub Flows: S-1: Access Device Gallery 1. The existing staff/ existing patient is required to choose photo from the device Gallery. If the existing staff/ existing patient choose photo from the device Gallery.		

Execute Image Classification Result use case

If the existing staff/ existing patient did not choose photo from the device Gallery.

the S-3: Return page is performed.

S-2: Deny Access to Device Gallery

1. The system will redirect the existing patient to the AI Scan page.

S-3: Return page

1. The system will redirect the existing patient to the AI Scan page.

Alternate / Exceptional Flows: Not Applicable

## Take Photo

Use Case Name: Take Photo	ID: <u>212</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to use the acne classification feature by taking a photo		
Brief Description: This use case describes how the existing staff/ existing patient can take a photo using the device camera to use the acne classification feature		
Trigger: Existing Staff/ Existing Patient wants to use the acne classification feature by taking photo. Type: External		
Relationships: Association: - Include: Image Classification Result Extend: - Generalization: -		
Normal Flow of Events: 1. The system will pop up a dialog box with two option “Allow” or “Deny” to gain permission access to device camera. If the existing staff/ existing patient choose “Allow,” the S-1: Access Device Camera is performed. If the existing staff/ existing patient choose “Deny,” the S-2: Deny Access to Device Camera is performed.		
Sub Flows: S-1: Access Device Camera 1. The existing staff/ existing patient is required to use the device camera to take the picture of the face. If the existing staff/ existing patient take picture of the face.		

Execute Image Classification Result use case

If the existing staff/ existing patient did not take picture of the face.  
the S-3: Return page is performed.

S-2: Deny Access to Device Camera

1. The system will redirect the existing patient to the AI Scan page.

S-3: Return page

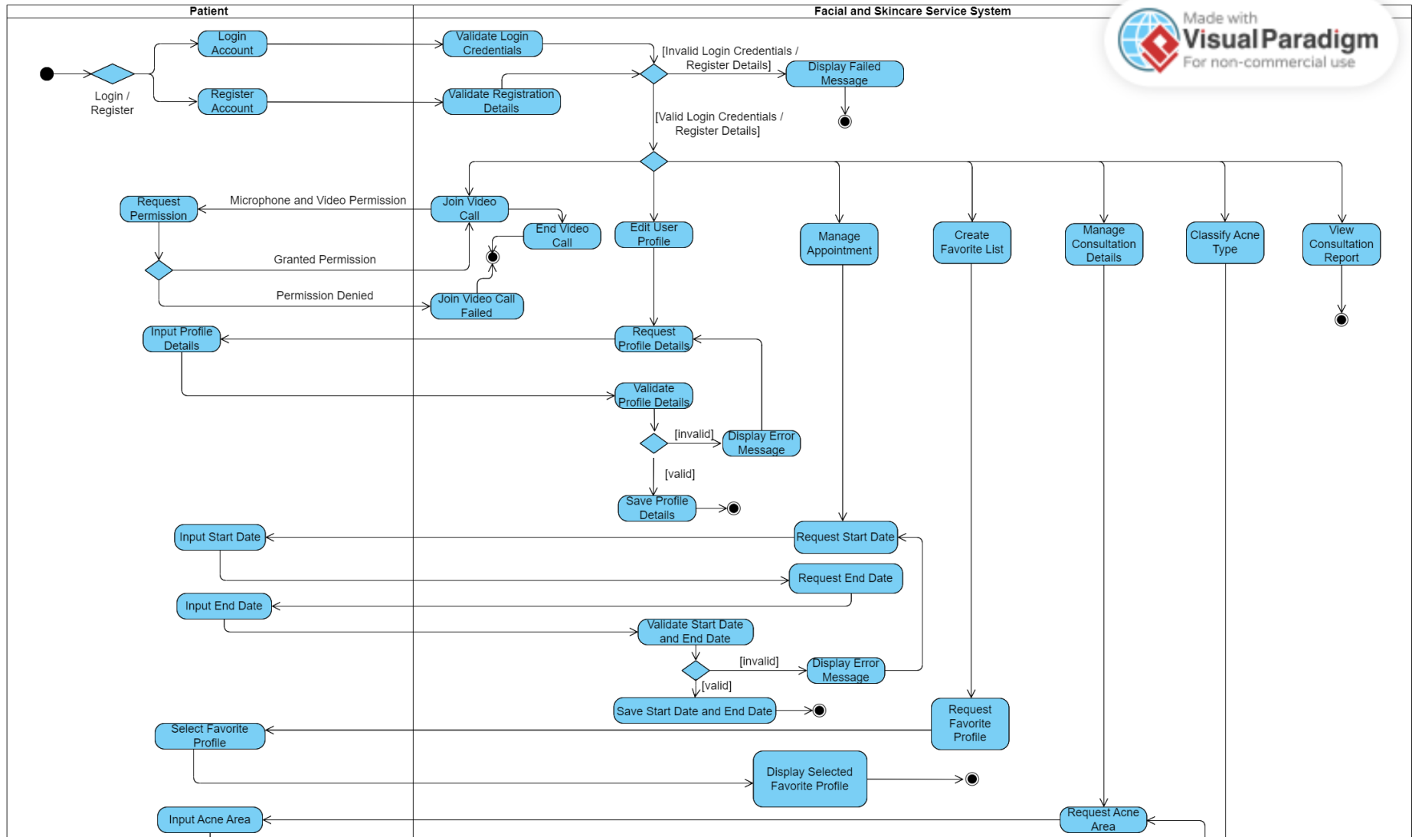
1. The system will redirect the existing patient to the AI Scan page.

Alternate / Exceptional Flows: Not Applicable

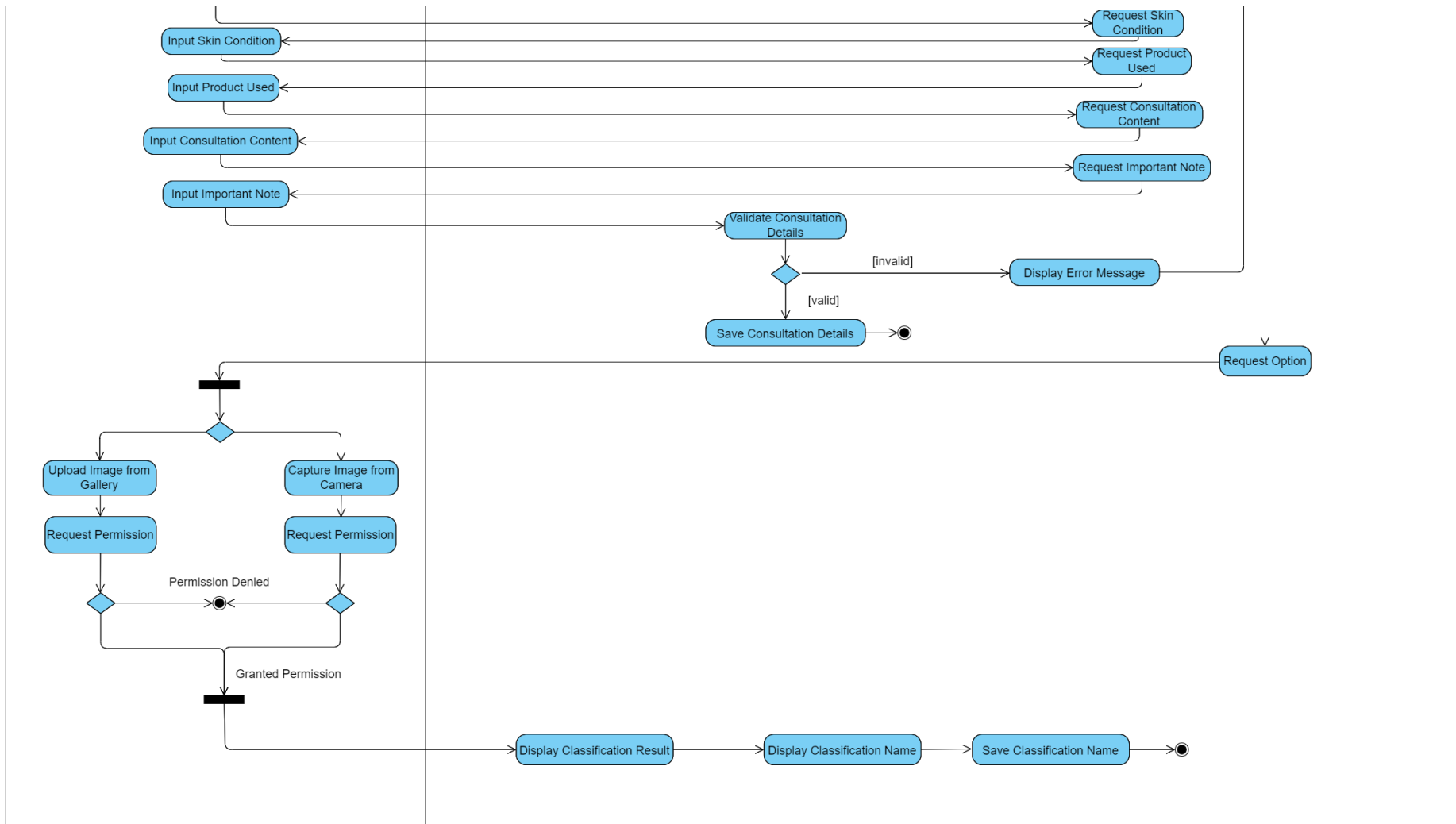
## View Image Classification Result

Use Case Name: View Image Classification Result	ID: <u>213</u>	Importance Level: <u>High</u>
Primary Actor: Existing Staff/ Existing Patient	Use Case Type: Detail, Essential	
Stakeholders and Interests: Existing Staff/ Existing Patient – wants to view or read the classification result		
Brief Description: This use case describes how the existing staff/ existing patient view or read the classification result of the acne type after uploading the image or take photo		
Trigger: Existing Staff/ Existing Patient wants to view the classification result after upload or take the acne skin. Type: External		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The system will display the result of the classification result in the AI Scan page. 2. The system will save the classification result in the database of the system.		
Sub Flows: Not Applicable		
Alternate / Exceptional Flows: Not Applicable		

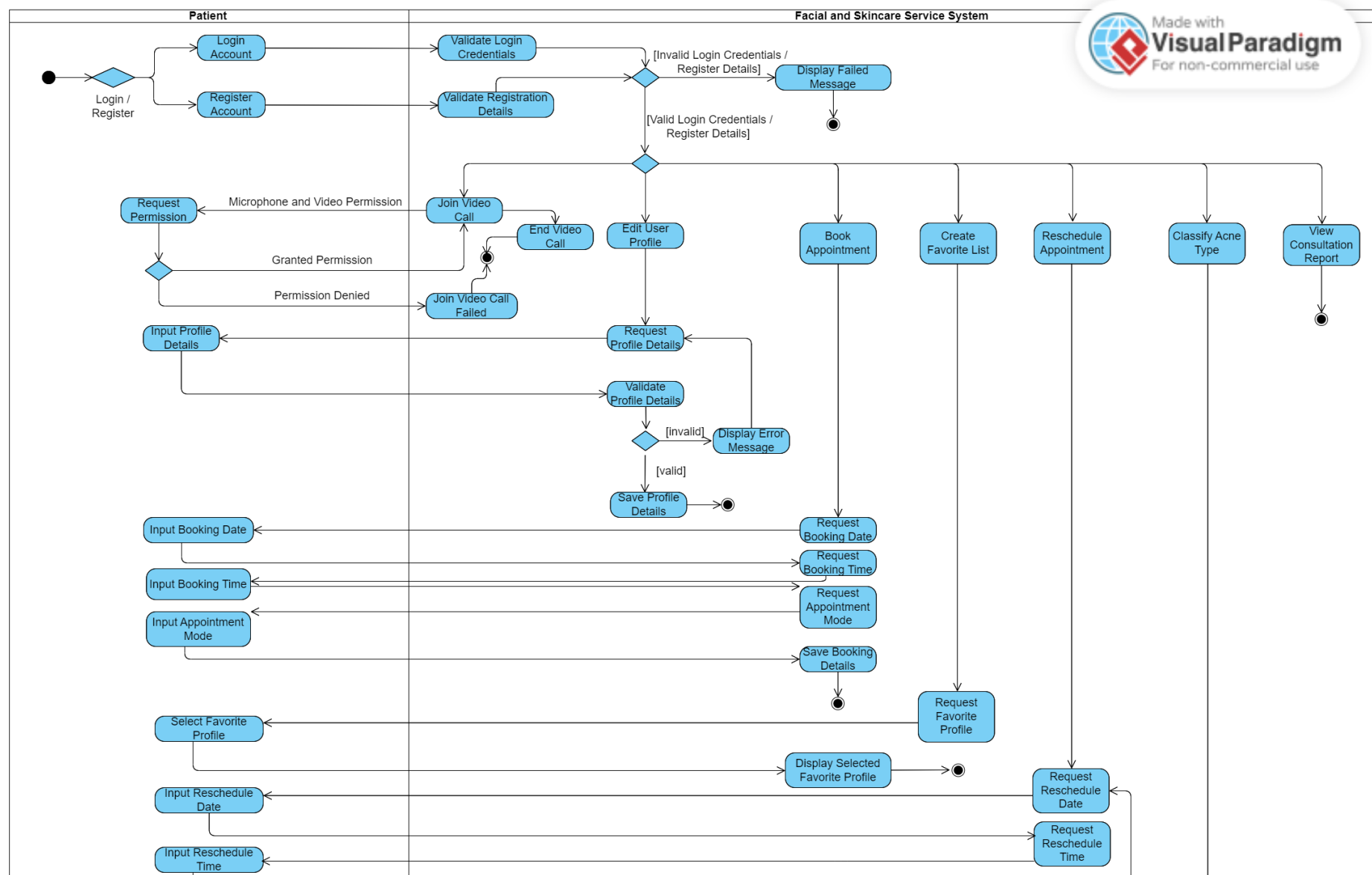
### 3.1.3 Activity Diagram

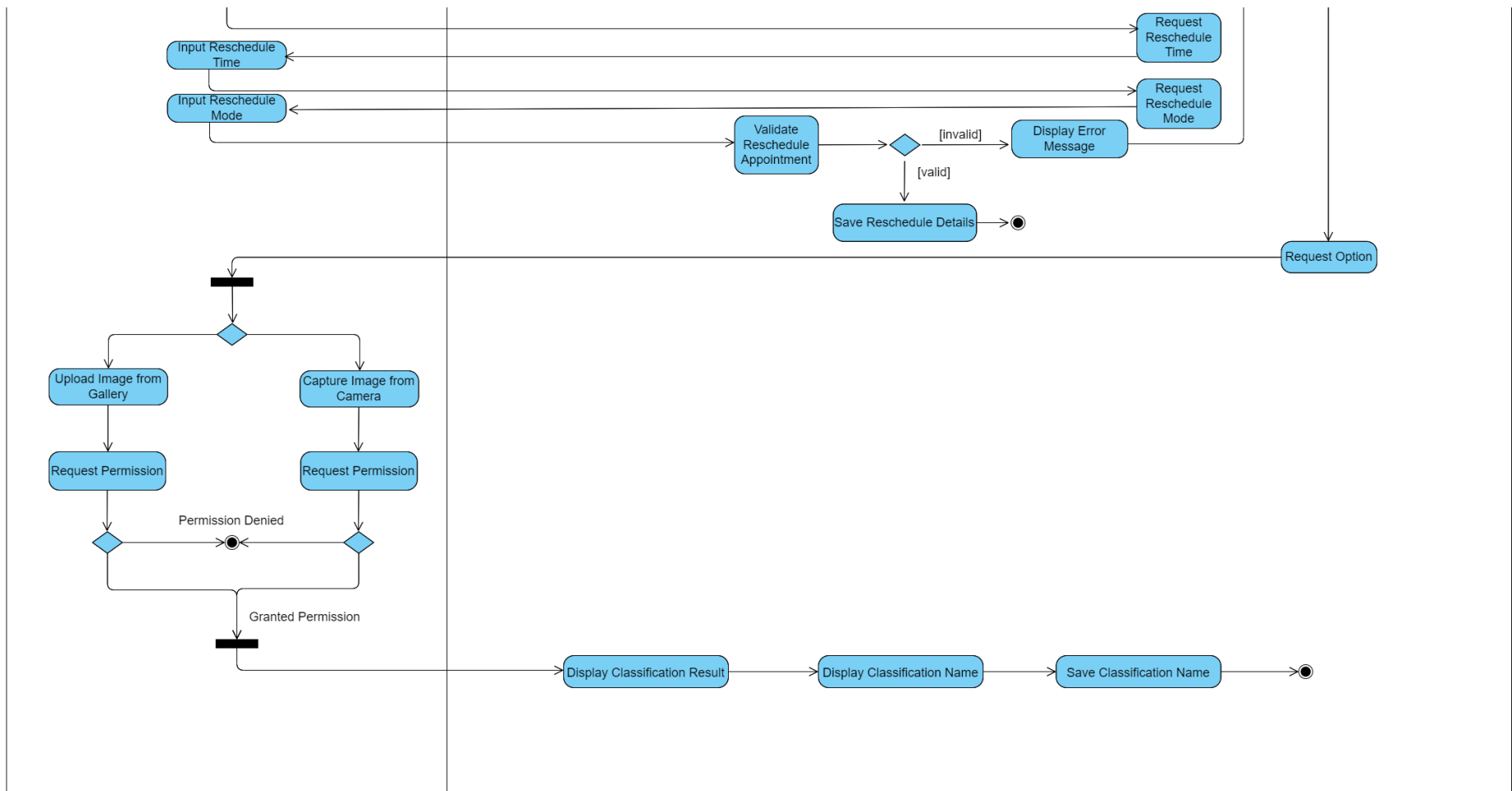






**Figure 3.1.3.1: Activity Diagram for Staff Module in Facial and Skincare Mobile Application**





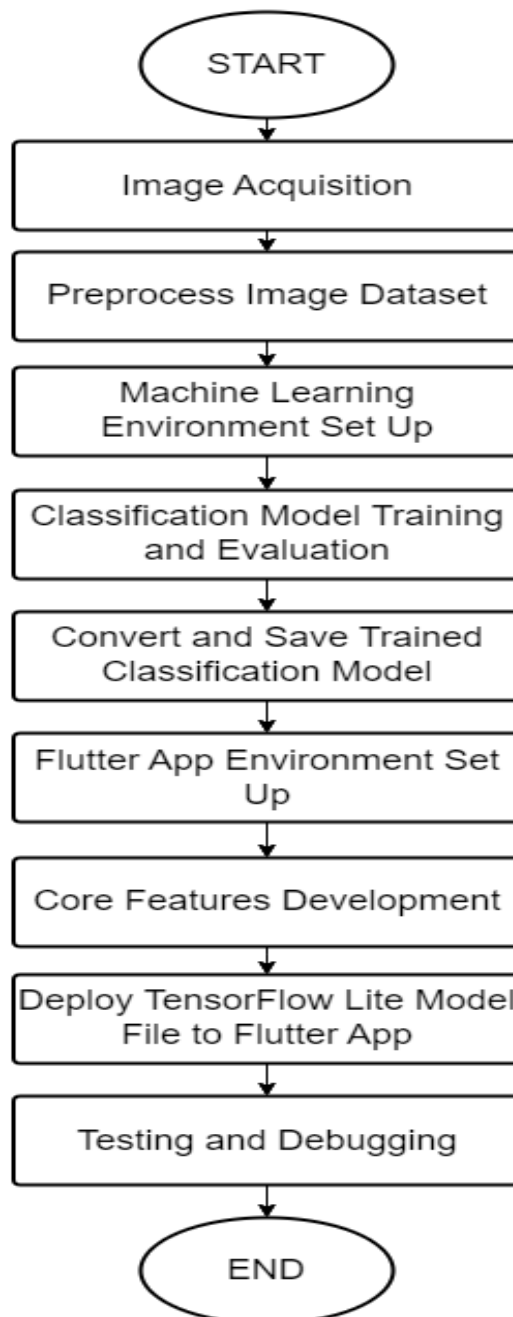
**Figure 3.1.3.2: Activity Diagram for Patient Module in Facial and Skincare Mobile Application**

# Chapter 4

## System Design

In this chapter, the flow of this project and the method on how the system is design and developed will be clearly explained.

### 4.1 Project Flow Diagram



*Figure 4.1: Project Flow Diagram*

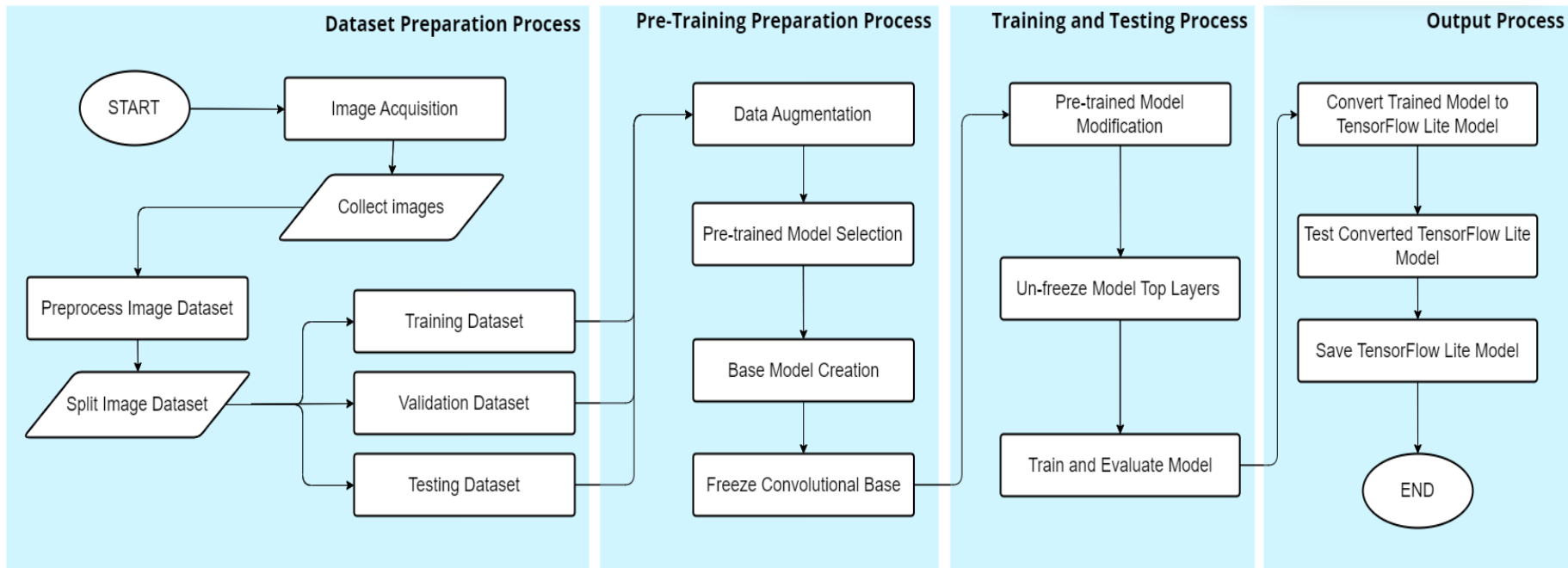
Figure 4.1 illustrates the detailed project flow diagram, which includes the entire process of the project, starting from image acquisition and leading up to the development of the Flutter Android application. The first step of the project is acquiring the images that will be used to train the TensorFlow model. This step involves downloading the pre-existing images from the dataset provided by the author on Kaggle. Once the phase of acquiring photos has been finished, the next stage is the preparation of the dataset, which involves dividing the dataset into a training dataset, a validation dataset, and a testing dataset. After that, the step of data preprocessing must be performed to guarantee that the format of the images can be used by the TensorFlow Model. The environment for machine learning needs to be set up before training the deep learning model. Once the environment has been done setting up, the TensorFlow model will need to be trained on the preprocessed data. This step involves the using of transfer learning for the deep learning that available in the TensorFlow library. After the model has been trained, model evaluation is needed to ensure the accuracy and reliability of the model. This step is done by testing the model on the set of test data. Once the model has been evaluated, the optimization of the model is required to convert into TensorFlow Lite model to suit the mobile devices. With the help of the TensorFlow Lite Converter, the TensorFlow model can be converted into a TensorFlow Lite Model File (.tflite).

Next, to deploy the model to the Flutter application, the environment of the Flutter application needs to be set up first. This step involves creating the user interface, integrating useful packages and plugins, and configuring the app to work with the TensorFlow Lite model and Firebase. The next step is to develop the core features of the application. This step includes the implementation of user authentication, database integration. Once the model has been optimized and the flutter application has been set up, the model can now be deployed to the flutter application. The subsequent stage involves developing the fundamental features of the application, such as user authentication and database integration. After optimizing the model and setting up the Flutter application, the model can be deployed to the application.

## **4.2 Train Deep Learning Model using Transfer Learning**

This section outlines the steps involved in building, training, and testing the deep learning model using transfer learning. Basically, transfer learning is a method that involves utilizing a pre-trained model and building a new layer on top of it to fit the specific requirements of a new dataset[15]. This approach has become popular for training deep learning models due to several advantages, such as reduced training time, improved efficiency of the neural network's functioning, and most importantly, suitability for projects with limited amounts of data. The deep learning architecture chosen for the transfer learning method is EfficientNetV2-B0, which is the latest addition to the family of EfficientNet architectures. EfficientNetV2-B0 is chosen because it is the smallest and least complex model among the EfficientNetV2 family. It is also more suitable for transfer learning on small dataset because it reduced the complexity, and it is less likely to overfit the data. Visual Studio Code has been used as the environment to train the model and it required the installation of TensorFlow. The installation of Jupyter is need by installing the package from the extension.

### 4.2.1 TensorFlow Lite Model Building Flowchart



*Figure 4.2.1: Flowchart for Building TensorFlow Lite Model*

Figure 4.2.1 describe the flow of each process to build the TensorFlow Lite Model. There are 4 processes need to be done to build the TensorFlow Lite Model. The first process is dataset preparation where the images is collected and the collected images will then split into training dataset, validation dataset and testing dataset. After splitting the dataset, pre- training preparation will be carried out with data augmentation on the dataset and the pre-trained model is identify. After it has been determined which pre-trained model will be utilized, a base model should be created by using the pre-trained mode, and then all of the convolutional base layers should be frozen. After the preparation of the model, the next step is to modify the pre-trained model to suit the output of the training data. This involves unfreezing the top layers of the model before training. Next, identify the suitable optimizer, learning rate, epochs and other relevant parameters to train the model. After training process is done, the model needs to be evaluated to ensure the accuracy and reliability. If the model is ready to be use, then convert the trained model to TensorFlow Lite model using converter. After converting the trained model, the converted model needs to be tested before saving it with the test data. Finally, save the converted model and the model is ready to be deployed in the Flutter app.



## 4.2.2 Dataset Preparation Process

In this section, the detailed of the preparation for dataset and the source of the dataset is explained.

### 1. Image Acquisition

The project dataset is collecting from online platform which is Kaggle which is a popular platform for data scientists to share and discover datasets, as well as to participate in machine learning competitions.

#### Acne grading classification dataset (Kaggle)

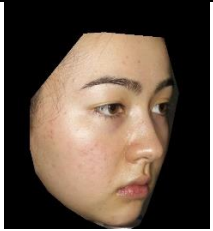






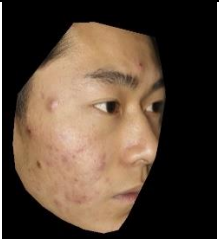


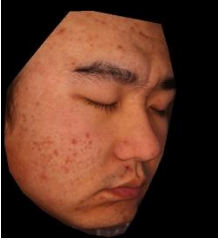

Based on the dataset obtained from Kaggle.com, a contributor named Rutvik Lathiya Teksun [16] provided an acne grading classification dataset with labels.

Types of Acne	Total Number of Images	Cropped	With Black Colour Background
Level_0 (Mild)	387	Yes	Yes
Level_1 (Moderate)	473	Yes	Yes
Level_2 (Severe)	139	Yes	Yes

*Table 4.2.1: Different types of acne provided in Kaggle*

Based on Table 4.2.1, the dataset is labeled according to the acne grading, which is ""Acne Level\_0"", ""Acne Level\_1,"" and ""Acne Level\_2"". This dataset is chosen because it fulfills this project's requirement to classify the skin condition according to the acne grading to diagnose the skin condition on the patient's skin. According to [16], Level\_0 classes consist of 387 skin images, 473 skin images for Level\_1, and 139 skin images in Level\_2.

Based on Table 4.2.1, the example for Level\_0 shows the skin condition with a few small pimples and some red breakouts usually found on the cheeks. In Level\_1, the dataset example shows the skin condition with more blemishes than Level\_0, and there are papules and pustules on the face, usually called pimples. The skin condition becomes more severe in Level\_2, which has a lot of papules and pustules over the face, and it can spread and merge. Thus, there will be more redness, and the face will be swelling. According to the author in [17], faces with few papules are considered under grade 1, which is mild acne; therefore, the dataset labeled as Level\_0 can be classified as mild acne. The dataset labeled Level 2 is equivalent to the description of [17], where there are multiple papules and pustules on the face under the moderate acne category. Lastly, moderate severe and severe acne falls under Level\_3 in the dataset, where there are inflamed pustules and nodules on the faces, according to the article in [17].

	Level_0 (Mild)	Level_1 (Moderate)	Level_2 (Severe)
<b>Example 1</b>			
<b>Example 3</b>			
<b>Example 4</b>			
<b>Example 6</b>			

**Table 4.2.2: Example of dataset with the labelling of acne grade**

*Source: Adapted from [16].*

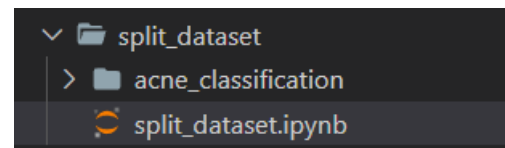
## 2. Preprocess Image Dataset

The initial file is store according to the acne label which is Level\_0, Level\_1, and Level\_2. Therefore, the file is renaming from Level\_0 to Mild, Level\_1 to Moderate and Level\_2 to Severe. After changing the file name accordingly, the file will then split into training dataset, validation dataset and testing dataset.

### Steps in splitting dataset:

#### 1. File directory

Create a new Jupyter file in Visual Studio Code. Make sure the downloaded dataset is in the same file as the Jupyter file



#### 2. Import Python libraries

Import the relevant Python libraries to the after creating the Jupyter file.

```
import os
import shutil
import tensorflow as tf
```

#### 3. Import data and initialize parameters

Initialize the base directory which is the file path of the dataset, and the sub file folder name is initialize as “names”.

```
BASE_DIR = 'acne_classification/'
names = ["Mild", "Moderate", "Severe"]

tf.random.set_seed(1)
```

#### 4. Reorganize folder structure

Create a new sub folder name as train, val, and test.

```
if not os.path.isdir(BASE_DIR + 'train/'):
    for name in names:
        os.makedirs(BASE_DIR + 'train/' + name)
        os.makedirs(BASE_DIR + 'val/' + name)
        os.makedirs(BASE_DIR + 'test/' + name)
```

## 5. Move images to new sub folder

```
orig_folders = ["Mild/", "Moderate/", "Severe/"]
for folder_idx, folder in enumerate(orig_folders):
    files = os.listdir(BASE_DIR + folder)
    number_of_images = len([name for name in files])
    n_train = int((number_of_images * 0.6) + 0.5)
    n_valid = int((number_of_images*0.25) + 0.5)
    n_test = number_of_images - n_train - n_valid
    print(number_of_images, n_train, n_valid, n_test)
    for idx, file in enumerate(files):
        file_name = BASE_DIR + folder + file
        if idx < n_train:
            shutil.move(file_name, BASE_DIR + "train/" + names[folder_idx])
        elif idx < n_train + n_valid:
            shutil.move(file_name, BASE_DIR + "val/" + names[folder_idx])
        else:
            shutil.move(file_name, BASE_DIR + "test/" + names[folder_idx])
```

### 4.2.3 Pre-Training Preparation Process

In this section, the detailed of the preparation before the training of model. There are several steps need to be done to prepare the environment and selection of the pre-trained model is explained.

#### 1. Setup Development Environment

After the preparation of dataset process is done, the python environment can be setup by install the extension of Jupyter in the Visual Studio Code. Create a new folder and paste in the dataset that have been preprocessed and open a new notebook (.ipynb). Next, import all the important libraries, image dataset and hyperparameters and execute in the runtime and autotune is used to improve the performance of input pipeline. Besides that, preprocess\_input is use because it helps to normalize the input data and make it more compatible with the requirements of the model.

```
import matplotlib.pyplot as plt
import tensorflow as tf
from itertools import product
```

*Figure 4.2.3.1: Import Libraries*

```
BATCH_SIZE = 32
IMG_SIZE = (224, 224)
IMG_SHAPE = IMG_SIZE + (3,)
```

*Figure 4.2.3.2: Initialization of variables*

```
train_dataset = tf.keras.utils.image_dataset_from_directory('acne_classification/train',
                                                           shuffle=True,
                                                           batch_size=BATCH_SIZE,
                                                           image_size=IMG_SIZE)

validation_dataset = tf.keras.utils.image_dataset_from_directory('acne_classification/val',
                                                                shuffle=True,
                                                                batch_size=BATCH_SIZE,
                                                                image_size=IMG_SIZE)

test_dataset = tf.keras.utils.image_dataset_from_directory('acne_classification/test',
                                                           shuffle=False,
                                                           batch_size=BATCH_SIZE,
                                                           image_size=IMG_SIZE)
```

*Figure 4.2.3.3: Import datasets*

```
preprocess_input = tf.keras.applications.efficientnet_v2.preprocess_input
```

*Figure 4.2.3.4: Snippet code for preprocess\_input*

## 2. Data Augmentation

Based on the Figure 4.2.3.4, the data augmentation model includes three layers that perform random modifications on the input images. The first layer randomly flips the image horizontally with a 50% probability, while the second layer rotates the image by up to 0.2 radians in either direction. The third layer zooms the image by up to a factor of 0.2 in either direction. These modifications help to simulate variations in image orientation, angle, and scale, which can be useful in tasks such as object detection, facial recognition, scene classification, and medical image analysis.

```
data_augmentation = tf.keras.Sequential([
    tf.keras.layers.RandomFlip('horizontal'),
    tf.keras.layers.RandomRotation(0.2),
    tf.keras.layers.RandomZoom(0.2),
])
```

*Figure 4.2.3.5: Snippet Code for Data Augmentation*

## 3. Pretrained Model Selection

EfficientNetV2-B0 is chosen for the project therefore the package is imported into the runtime through the Keras Application library.

## 4. Based Model Creation

After identifying the pre-trained model that will be incorporated into the deep learning model, the EfficientNetV2B0 model from the Keras deep learning framework must be initialized with certain parameters [18] as shown in Figure 4.3.2.1.

```
base_model = tf.keras.applications.EfficientNetV2B0(
    input_shape=IMG_SHAPE,
    include_top=False,
    weights='imagenet')
```

*Figure 4.2.3.6: Snippet Code for initialization of EfficientNetV2B0 model*

## 5. Freeze Convolutional Base

Based on Figure 4.3.2.2, the “trainable” attribute of the model object is set to “False” to fixed and not updated the weight loaded from the ImageNet dataset during the subsequent training. This is because transfer learning is use in this training model process by keeping the pre-trained weights fixed, the model will retain the features from ImageNet [18].

```
base_model.trainable = False
```

*Figure 4.2.3.7: Snippet Code for freezing the convolutional base*

## 4.2.4 Training and Testing Process

### 1. Pre-trained Model Modification

There are three layers added to rebuild the top layer for the model. Based on Figure 4.2.3.3, the code creates a Keras model for a classification task. The model consists of an input layer that specifies the shape of the input images, a data augmentation pipeline to increase the model's robustness, and an EfficientNet model as the feature extractor. The EfficientNet model is frozen, which means that its weights are not updated during training. To reduce the number of parameters in the model, a GlobalAveragePooling2D layer is applied to the output of the EfficientNet architecture. A Dropout layer is used for regularization to prevent overfitting during training. Finally, a Dense layer with 3 output units is used in the process of producing the model's final output, corresponding to the 3 classes in the classification task. The model is defined using the Keras Model API, which takes the input and output layers as arguments, and all the intermediate layers are defined in between [18].

```
global_average_layer = tf.keras.layers.GlobalAveragePooling2D()
prediction_layer = tf.keras.layers.Dense(3)

inputs = tf.keras.Input(shape=(224, 224, 3))
x = data_augmentation(inputs)
x = preprocess_input(x)
x = base_model(x, training=False)
x = global_average_layer(x)
x = tf.keras.layers.Dropout(dr)(x)
outputs = prediction_layer(x)
model = tf.keras.Model(inputs, outputs)
```

*Figure 4.2.4.1: Snippet Code for rebuild layers on top layer*

### 2. Un-freeze Model Top Layers

Based on Figure 4.2.3.9 `base_model.trainable` is set to `True` in TensorFlow Keras, it indicates that the layers in a pre-trained model can have their weights updated during the training process. This is used to fine-tuning the model, which is a technique to adapt a pre-trained model to a new task or dataset.

```
base_model.trainable = True
```

*Figure 4.2.4.2: Snippet Code for unfreeze some layers*



### 3. Train and Evaluate Model

The code for the Keras model is shown in Figure 4.2.4.3, where it specifies the loss function, optimizer, and evaluation metrics. The SparseCategoricalCrossentropy is a loss function that measures the difference between the predicted probability distribution and the true probability distribution of the target class. In this case, When from\_logits is set to True, it assumes that the output of the model is in the form of logits (unnormalized log probabilities) rather than probabilities. Adam is an optimizer that uses an adaptive learning rate to update the model parameters during training.

```
# Compile the model
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=lr),
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

*Figure 4.2.4.3: Snippet Code for declaring hyperparameters*

Based on Figure 4.2.4.11, The model.fit() method accepts several arguments that include train\_dataset, epochs, validation\_data and verbose. The train\_dataset refer to the training dataset, while the epochs is the number of times the model will be trained on the entire training dataset. The validation\_data is referring to the validation dataset and the verbose is set to 0 will run the training process silently.

```
# Train the model
history = model.fit(train_dataset,
                    epochs=ie,
                    validation_data=validation_dataset,
                    verbose=0)
```

*Figure 4.2.4.4: Snippet Code for training model*

## 4.2.5 Output Process

### 1. Convert Trained Model to TensorFlow Lite Model

TFLiteConverter is used to perform the conversion of the model into TensorFlow Lite Model. The converter will get the TensorFlow model and assign as converter. The converter will then be used to perform “convert()” function and assign to tflite\_model [19].

```
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
```

*Figure 4.2.5.1: Snippet Code for Converting model to TensorFlow Lite Model*

### 2. Test Converted TensorFlow Lite Model

Based on Figure 5.2.5.2, the code evaluates a TensorFlow Lite model on a test dataset by making predictions using the interpreter. The accuracy for each class is calculated by comparing the predicted class name with the true class name and is then printed. The code assumes the presence of a list of class names and a test dataset consisting of input and label batches. The TensorFlow Lite model is expected to be saved as a file, and the TensorFlow Lite Python interpreter should be imported.

```
# Load the TensorFlow Lite model
interpreter = tf.lite.Interpreter(tflite_model)
interpreter.allocate_tensors()

# Get input and output details
input_details = interpreter.get_input_details()
output_details = interpreter.get_output_details()

# Evaluate the model on the test dataset
correct = {name: 0 for name in class_names}
total = {name: 0 for name in class_names}
for i in range(len(test_batches)):
    inputs, labels = test_batches[i]
    for j in range(inputs.shape[0]):
        interpreter.set_tensor(input_details[0]['index'], inputs[j:j+1])
        interpreter.invoke()
        output_data = interpreter.get_tensor(output_details[0]['index'])
        predicted_class_index = np.argmax(output_data)
        predicted_class_name = class_names[predicted_class_index]
        true_class_index = int(labels[j])
        true_class_name = class_names[true_class_index]
        if predicted_class_name == true_class_name:
            correct[true_class_name] += 1
            total[true_class_name] += 1

# Print the accuracy for each class
for name in class_names:
    accuracy = correct[name] / total[name]
    print(f'Accuracy for {name}: {accuracy}')
```

*Figure 4.2.5.2: Snippet Code for Testing Converted TensorFlow Lite Model*

### 3. Save TensorFlow Lite Model

After testing the TensorFlow Lite model, create a file to save the model. The code in Figure 4.2.5.3 is to save the TensorFlow Lite model as a binary file named model.tflite using the with open statement and writes the model's binary content into the file. It also saves the list of labels into a labels.txt file named using the with open statement and writes each label on a new line using the join function. The code assumes the presence of a directory named model to save the model and label files into, and that the list of labels is already defined in the label's variable [19].

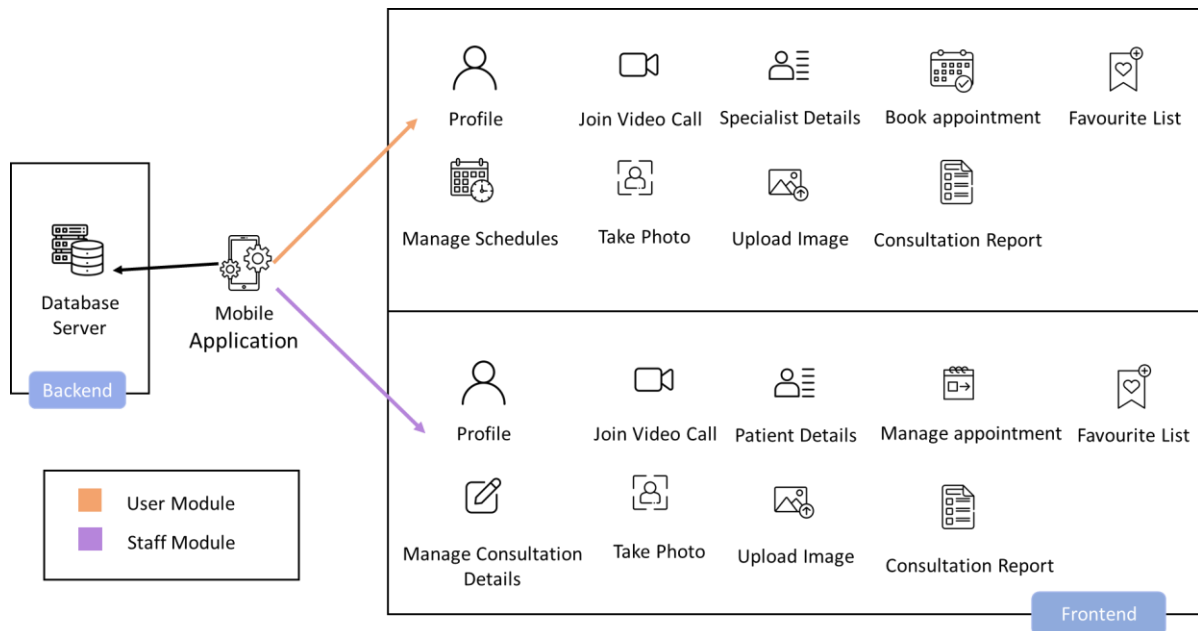
```
with open('model/model.tflite', 'wb') as f:
    f.write(tflite_model)

# Save the labels to a labels.txt file
labels = ['Mild', 'Moderate', 'Severe']
with open('model/labels.txt', 'w') as f:
    f.write('\n'.join(labels))
```

*Figure 4.2.5.3: Snippet Code for Saving Converted TensorFlow Lite Model*

### 4.3 System Design

In this service system, it is distributed into 2 different modules which is staff module and patient module for different roles such as staff and patient as shown in Figure 4.3. The details for each of the functionalities will be explained further as below.



*Figure 4.3: System Design of facial and skincare service system*

First, the service system required all users to enroll into the application in order to verify their role and gain access to the appropriate module. Therefore, the **login and register user account** feature is important to indicate the user's role to access the right module. New users are required to create an account before accessing the system. During the registration process, users are prompted to select their specific role, which is then assigned to them by the system. The following features is that is accessible by all the users is **manage profile**. The user can make use of the feature to edit the basic information of their profile.

Besides that, the system also provides a feature for all users to create a **favorite list**, which allows users to add profiles to or remove profiles from their favorite list. Moreover, the **consultation report** can also be use by all the users to view the consultation report regarding the consultation history and details. Both patient and staff will be having the features of **join video call**. This join video call function is for the purpose of conducting online consultation

based on the appointment mode select by the patient. For instance, if the patient chooses online consultation, then both patient and staff will be able to see the join video call button when the consultation session is ready to be conduct.

Apart from that, the feature of **taking photo** and **uploading image** is available in both staff and patient module which is for the acne classification service. In the patient module, they user can use it any time because it is available in their profile page. The patient can update their acne condition by using this feature. However, in staff module, the staff can only use this feature when the appointment of the patient is in physical mode.

In short, these are the functionalities that will be shared among those 2 modules, while the following will be the functionalities that are uniquely exists in their respective modules.

#### **a) Staff Module**

- **View Patient Details**

This feature allows the staff to search the list of the patient and see the details of the patient such as name, age, acne type, and other basic details of the patient. Other than that, there is an extra feature in this feature were the consultation history of the patient is accessible in the patient details which allow the staff to trace back the skin problem the patient.

- **Manage Appointment Date**

This feature is mainly for the staff to edit the booking calendar at the side of the patient. The booking calendar can be edited by staff members who have the ability to submit leave applications. They can enter a start date and an end date for their leave, and the calendar will be automatically updated to disable those dates, making them unavailable for bookings during that period. This is to ensure that when the staff and taking any leave from the work and is unavailable to conduct any consultation session, the patient will not be able to reach the staff for that period to avoid any misplacement of the appointment schedules.

- **Manage Consultation Details**

This is the main feature for the staff module where the staff is the only person that can edit the consultation details. This consultation details are helping to record the skin condition, acne location on the face, product used and the consultation content as well as the important note for the patient. The staff is responsible to complete this consultation details form every time the consultation session is completed. This information will be saved with the appointment details which allow the staff and patient to trace back the record according to the appointment's date and time.

**b) Patient Module**

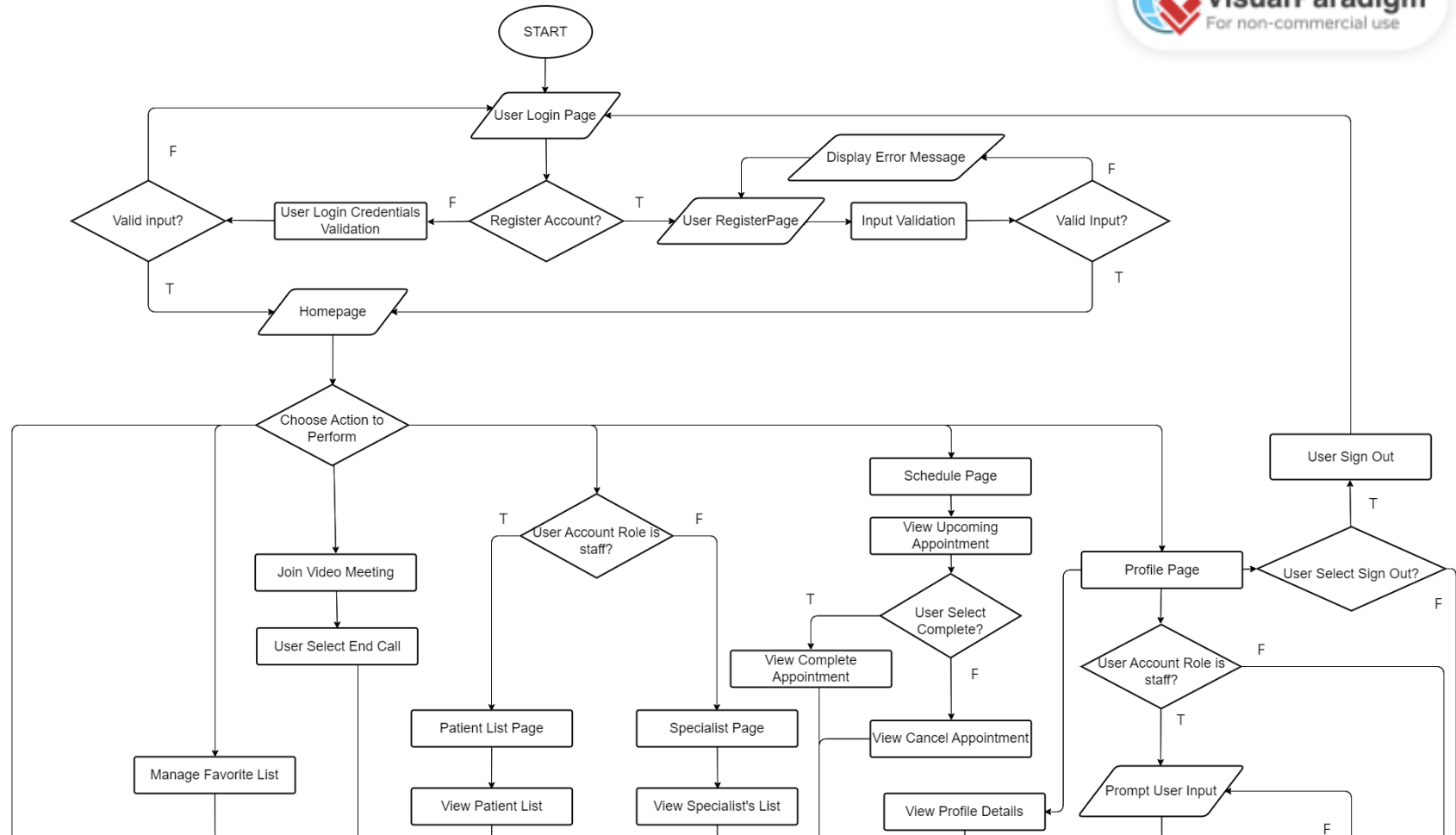
- **View Specialist Details**

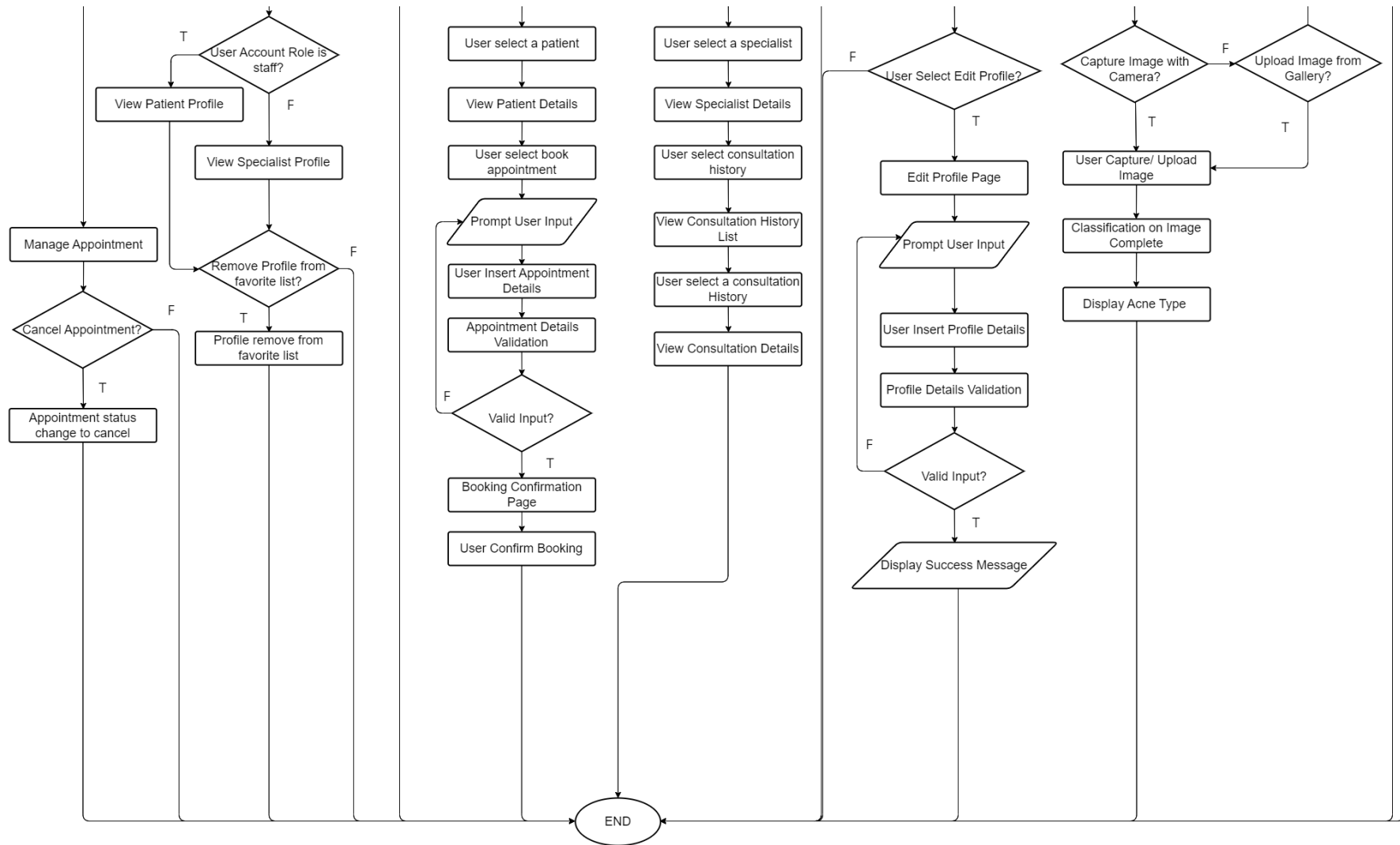
This feature allows the patient to search the list of the patient and see the details of the patient such as name, experience, specialist area, and other basic details of the staff. In this feature, there are favorite button and book appointment button which allow the patient to add the specialist to their favorite list so that easy to find to book for next appointment. The book appointment button allows the user to book an appointment with the selected specialist.

- **Manage Appointment**

This is the main feature in patient module where the patient can book an appointment with the specialist which is also known as the staff via online. The patient does not need to ask for the available time slot from the staff to book an appointment which is more convenient. Other than that, the patient can reschedule the appointment or cancel the appointment if the appointment that have been booked earlier cannot be attend. The appointment booking information will update to the staff immediately after the patient has made any changes regarding their appointment date, time, and mode.

### 4.3.1 Application System Flowchart





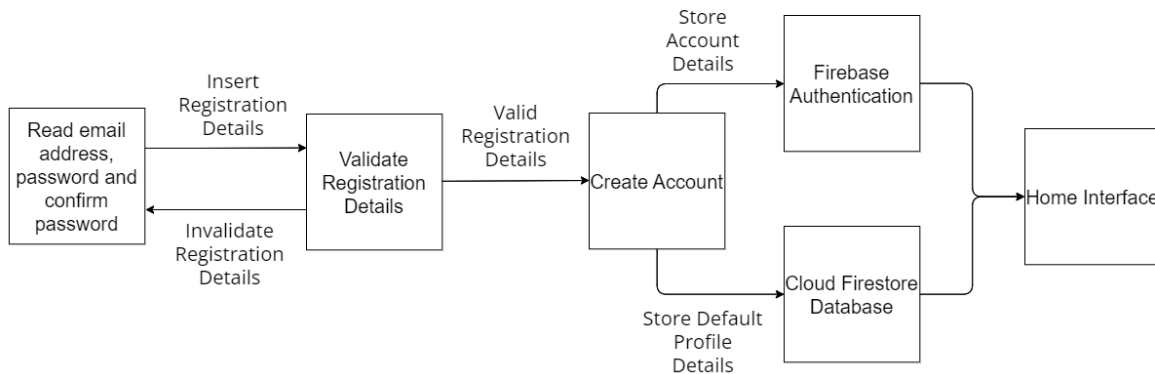
**Figure 4.3.1: Flowchart for Application System**



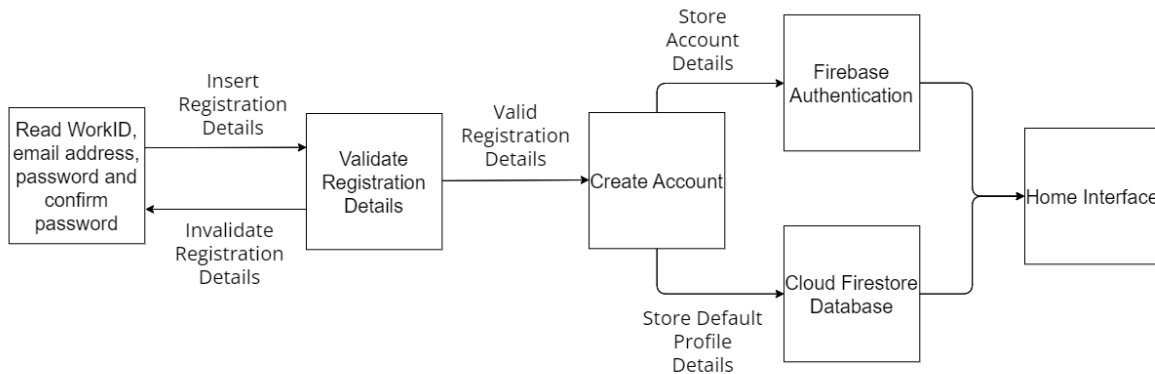
Based on Figure 4.3.1, it is illustrated the flow of the mobile application system which contain the important features of the system. When the application is launched for the first time, the user will interact with the User Login page. Upon accessing the User Login page, the user will be presented with the option to either register a new account or login with an existing one. If the user chooses to register an account, they will be redirected to the User Register page. To register an account, the user needs to provide accurate information. If the user provides incorrect information during registration, an error message will be displayed, and the user will be prompted to correct their input. However, if the user successfully provides the correct information, they will be registered as a new user and be able to login to the system. If the user chooses to log in to their account, the user is required to provide the correct login credentials. When the user chooses to login, they will be prompted to enter their correct login credentials. If the information provided is incorrect, the system will display an error message. If the login is successful, the user will be redirected to the system's homepage. When a user successfully logs into the system, they will be directed to the homepage where they can access the system's features and functionalities. The use can choose to perform any action to interact with the features in the system. The details of each feature will be discussed in the sub chapters using the block diagram for more in-depth visualization on the operation of the application system.

### 4.3.2 User Account Registration and Login

- **User Account Registration**



**Figure 4.3.2.1: Account Registration Block Diagram for Patient Module**

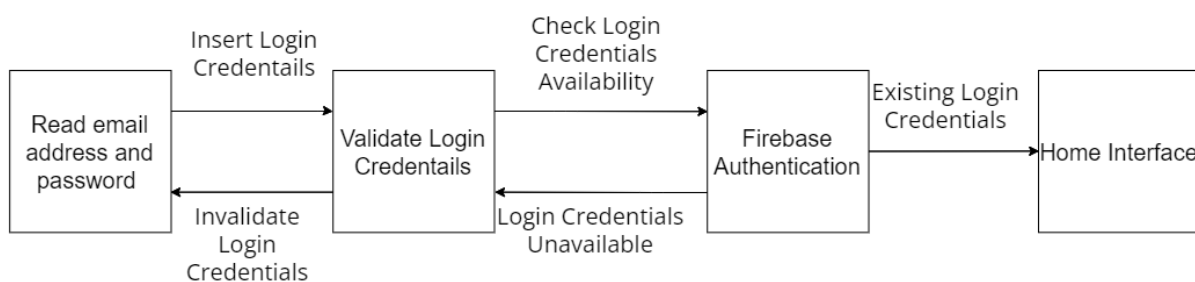


**Figure 4.3.2.2: Account Registration Block Diagram for Staff Module**

Based on Figure 4.3.2.1, the design of the user account registration involved the use of Firebase Authentication, Firebase Storage and Cloud Firestore Database which provide by the Firebase platform for mobile development. The use of Firebase Authentication is to create an authenticated account during the registration process. In this system, the initial profile image is set by default in the system for all new users is stored using Firebase Storage. This allows the system to easily retrieve and display the default profile image for each new user who logs in. This default profile picture will then store in the Cloud Firestore Database as an image path with the data type of string. Other than that, when a new user registers and logs in for the first time, the system initializes and stores their default details in the Cloud Firestore Database. The user is required to select their role to login the system before they filled up other details. If the user selects the role as staff, there will be one extra information need to be provided which is

the work ID as shown in Figure 4.3.2.2. Work ID is the employee ID that the company provide. This work ID act as the secret key for the system to assign the correct interface to the user according to their role in the system. When registering for an account, the user is required to give a number of essential pieces of information, including their email address and a password. When the email address and password provided by the user are valid, the system will redirect the user to the home interface and display a successful registration message. However, if the data is invalid, an error message will be displayed.

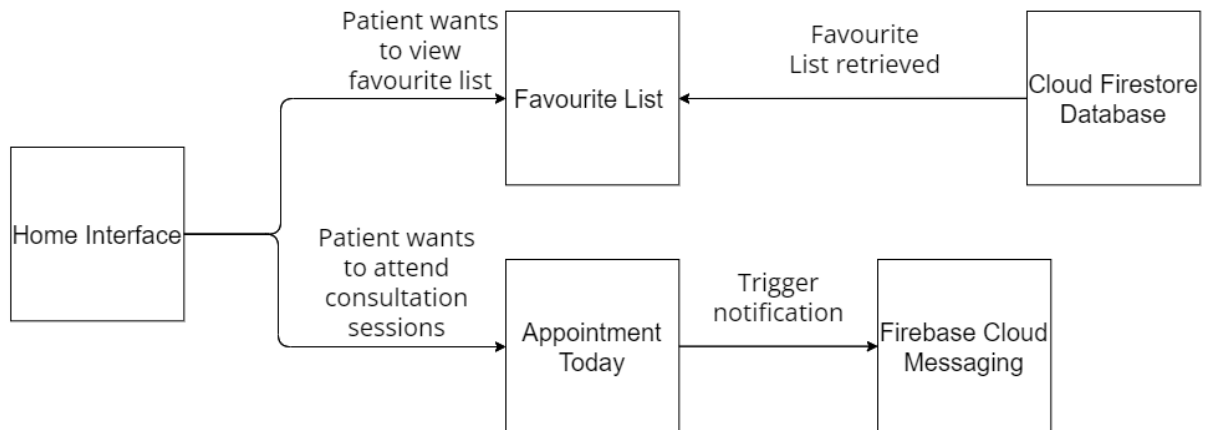
- **User Account Login**



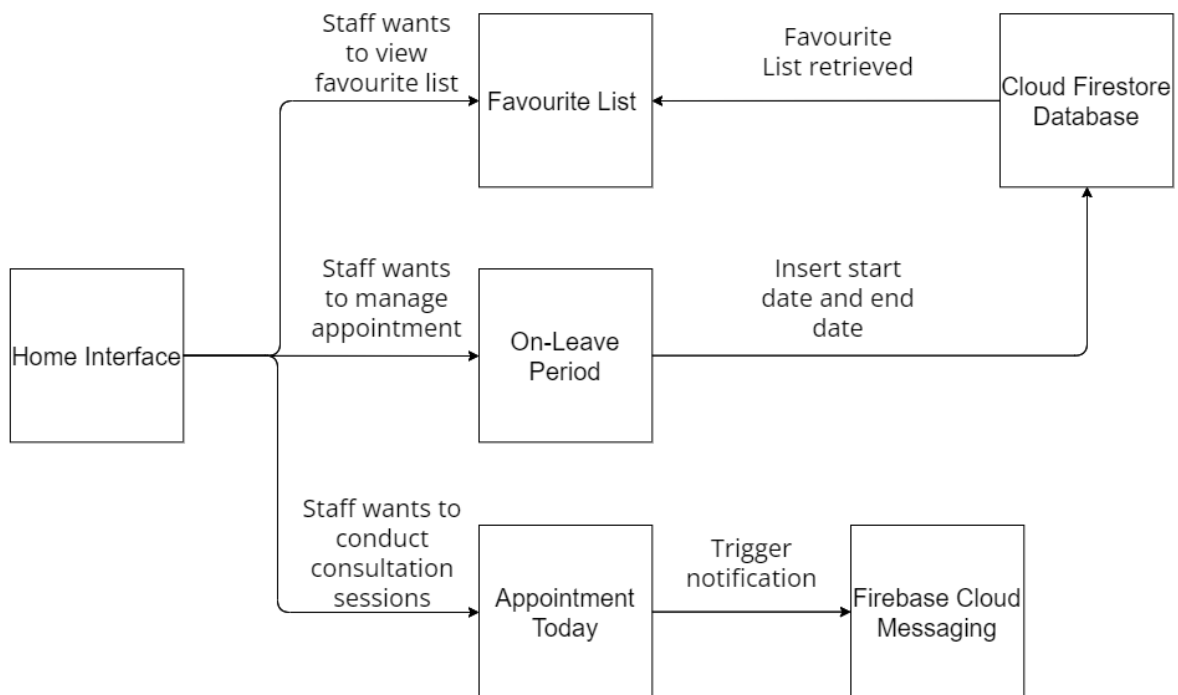
**Figure 4.3.2.3: Account Login Block Diagram**

Based on Figure 4.3.2.3, the design of the user account login involved the use of Firestore Authentication which is responsible to validate the account login credentials during the user login process. To access the system, the user is prompted to provide their email address and password through the login interface. The login process requires the user to input their email address and password. The system will then validate the email address and password. If the email address or password provided is incorrect, the user will not be able to login and an error message will be displayed prompting the user to re-enter their login details. If the user provides the correct login credentials, then the system will authenticate the user's credentials and allow them to access the home interface of the system.

### 4.3.3 Home Interface



**Figure 4.3.3.1: Home Interface Block Diagram for Patient Module**



**Figure 4.3.3.2: Home Interface Block Diagram for Staff Module**

According to the Figure 4.3.3.1 and Figure 4.3.3.2, the home interface design involved the use of Cloud Firestore Database and Firebase Cloud Messaging. Upon the user login, the data will be retrieved from the Cloud Firestore Database to display on the home interface. Firebase Cloud Messaging is responsible to send the notification to the user if they have an appointment today. The home interface is different among the role of the user.

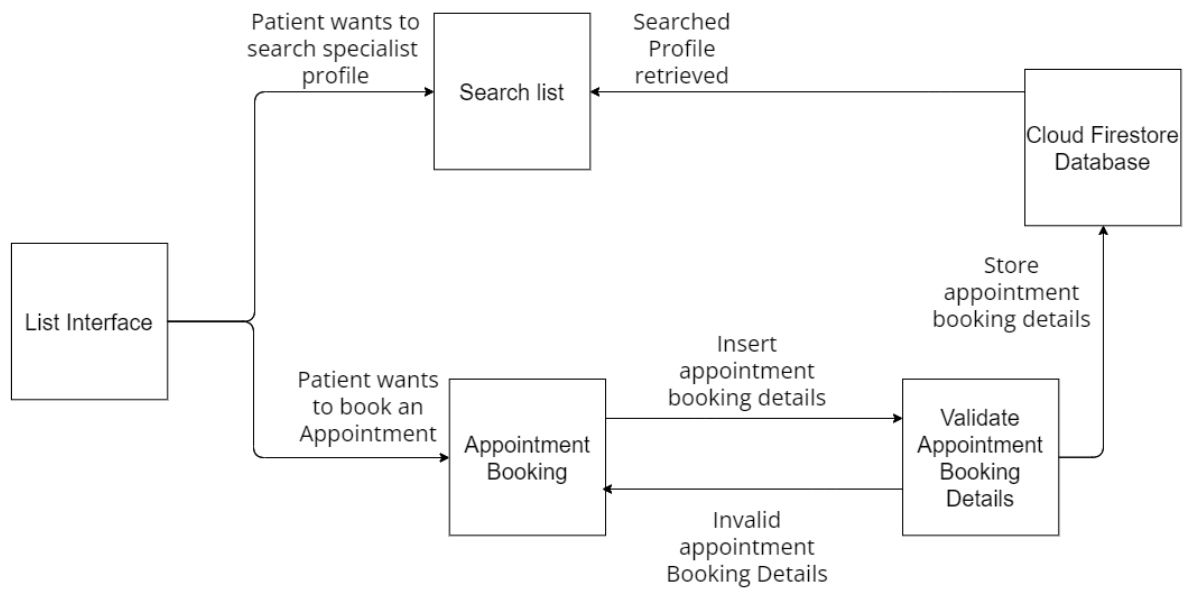
- **Home Interface (Patient Module)**

The home interface for patient has the name and profile picture on the top and the appointment today will be display in the white area as the first features in the home interface of the patient. The appointment today card will have different button depends on the appointment mode. If the appointment mode is physical, then the button will be cancel. The patient can choose to cancel the appointment if they cannot attend the consultation session. If the appointment mode is online, the button shows will be Join call and cancel. The patient is required to join the call during the consultation session to meet with their consultant. Lastly, the favorite list the last features in the home interface where the patient can add the specialist profile by searching the profile in the List interface and click the add favorite icon. The profile will then appear in the Favorite list section on the home interface as a shortcut way to search for the specialist. The user can remove the favorite profile by click into the profile and press the heart icon and the profile will be remove from the home interface.

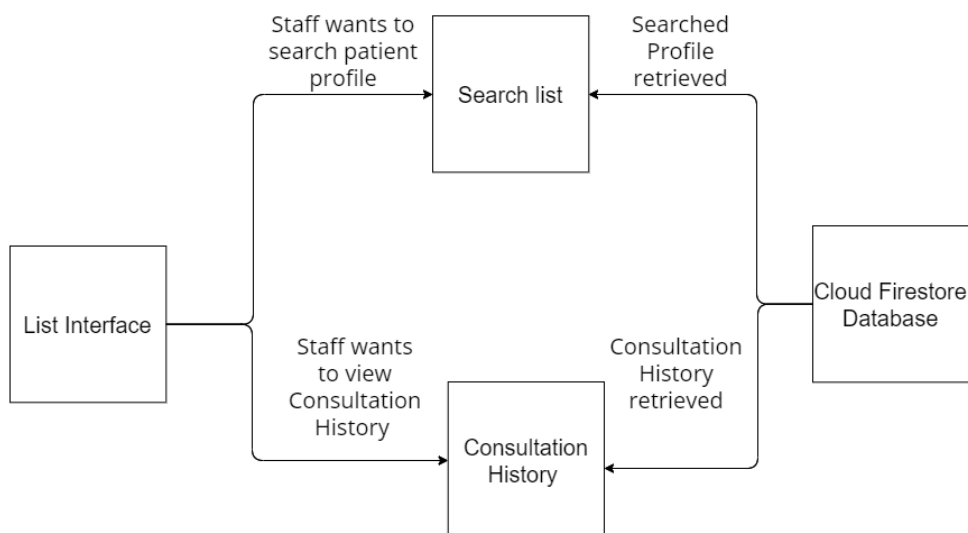
- **Home Interface (Staff Module)**

The home interface for staff has the name and profile picture on the top and the appointment today will be display in the white area as the first features in the home interface of the staff. The appointment today card will have different button depends on the appointment mode. If the appointment mode is physical, then the button will be AI Scan and Complete. The staff need to use the AI Scan to scan the face of the patient to update the skin category. When the consultation session is done, the staff is required to filled up the details of the consultation to complete the consultation. If the appointment mode is online, the button shows will be Join call and complete. The staff can click on the join call to conduct an online consultation session with the patient. For the staff, they have addition feature on the home interface, which is the on-leave date, if the staff have the record for apply leave, then the data will be retrieve from the database and display in the section respectively. Lastly, the favorite list performs the same function as in the home interface of the patient, but the list in the home interface for staff is patient list.

### 4.3.4 List Interface



**Figure 4.3.4.1: List Interface Block Diagram for Patient Module**



**Figure 4.3.4.2: List Interface Block Diagram for Staff Module**

Based on Figure 4.3.4.1 and Figure 4.3.4.2, the design of the list interface involved the use of Cloud Firestore Database. Upon the user login, the data will be retrieved from the Cloud Firestore Database to display on the list interface. The List interface is different from the module because each role is required to perform different task. The common feature in both module is the user can search the list using the search box by typing the name. Other than that, when the user selects a profile to view, the user will be redirect to the profile details interface which is all the details for the selected profile. The details are retrieved from the Cloud Firestore

Database and display on the profile details interface. The user can only view the profile details to gain information.

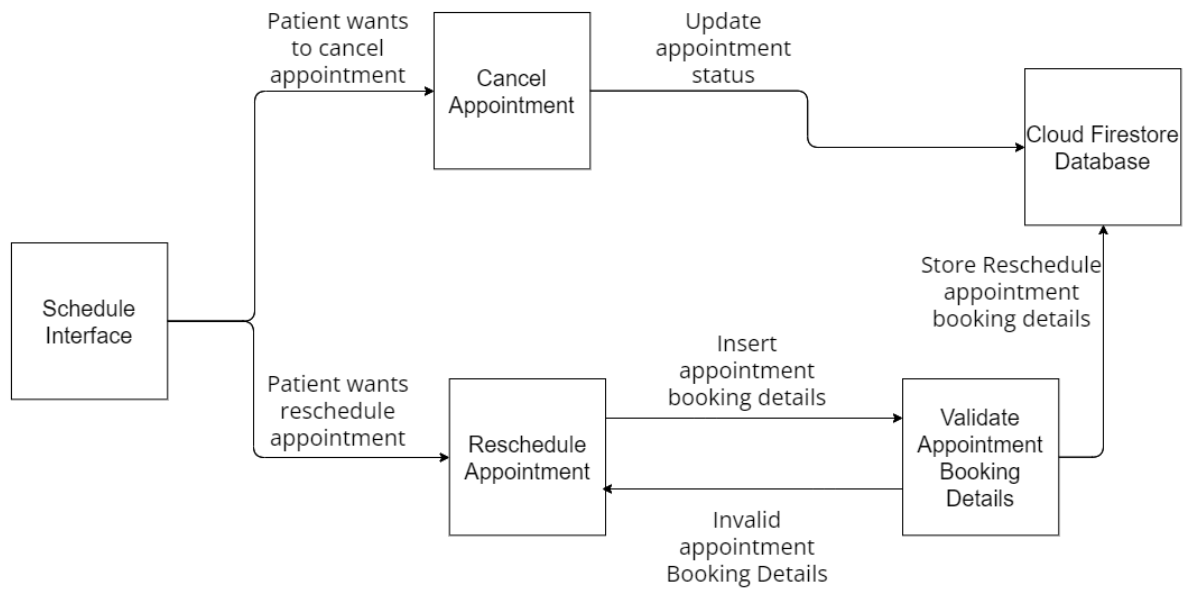
- **Specialist Profile Details Interface (Patient Module)**

The patient can click on the “Book Appointment” button to book an appointment with the selected specialist. After clicking the button, the patient will be redirect to the appointment interface. The patient is required to select the appointment date, appointment time, and appointment mode to book the appointment. Once the patient has selected the date, time and mode, the button “Make Appointment” will change color from Grey to purple which indicates that the button is clickable. The patient is required to press on the button and the current page will navigate to the booking confirmation page where the patient need to press in the back to homepage to complete the whole process of the booking. In the booking confirmation page, the successful message will show when the appointment is valid and can be booked but if the appointment is not valid which is when the patient has an appointment on the same date and time the appointment is invalid and also when the patient has booked more than 1 appointment with the same specialist on the same date the error message will be display on the booking confirmation page. The valid appointment details will then save to the Cloud Firestore Database.

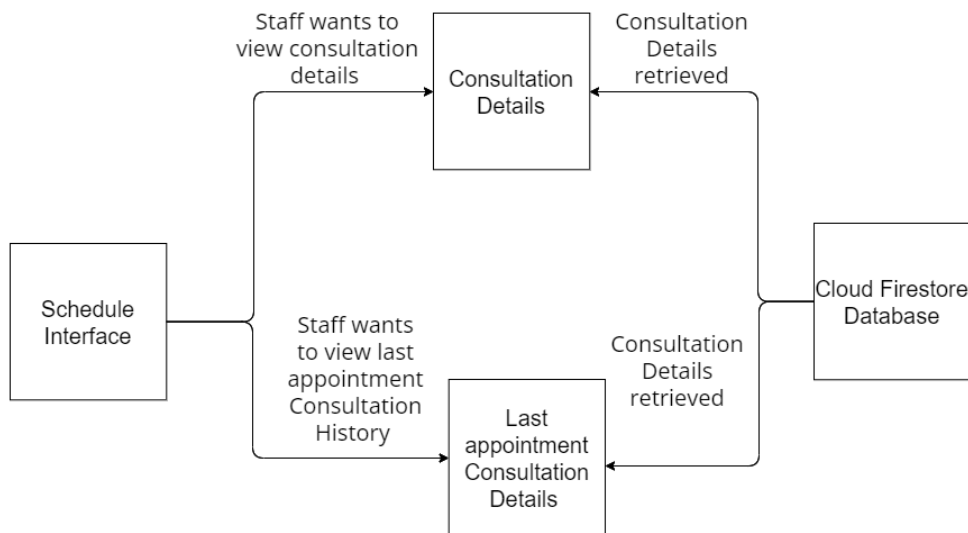
- **Patient Profile Details Interface (Staff Module)**

The staff can click on the “Consultation History” button to view the consultation history on the selected patient. When the staff click on the button, the staff will be redirect to the consultation history interface. In the consultation history interface, the staff can choose the desired consultation to view the details of the consultation by clicking the “View Consultation Details” button. The staff will then further redirect to the consultation details interface. In the consultation details interface, the staff can only view the details. The consultation details are retrieved from the Cloud Firestore Database and display it on the interface.

### 4.3.5 Schedule Interface



**Figure 4.3.5.1: Schedule Interface Block Diagram for Patient Module**



**Figure 4.3.5.2: Schedule Interface Block Diagram for Staff Module**

Based on Figure 4.3.5.1 and Figure 4.3.5.2, the design of the schedule interface involved the use of Cloud Firestore Database. Upon the user login, the data will be retrieved from the Cloud Firestore Database to display on the schedule interface. The schedule interface is different from the module because each role is required to perform different task. The common feature in both module is the user can swap the tab between upcoming, complete and cancel to view different appointment which is in different status.



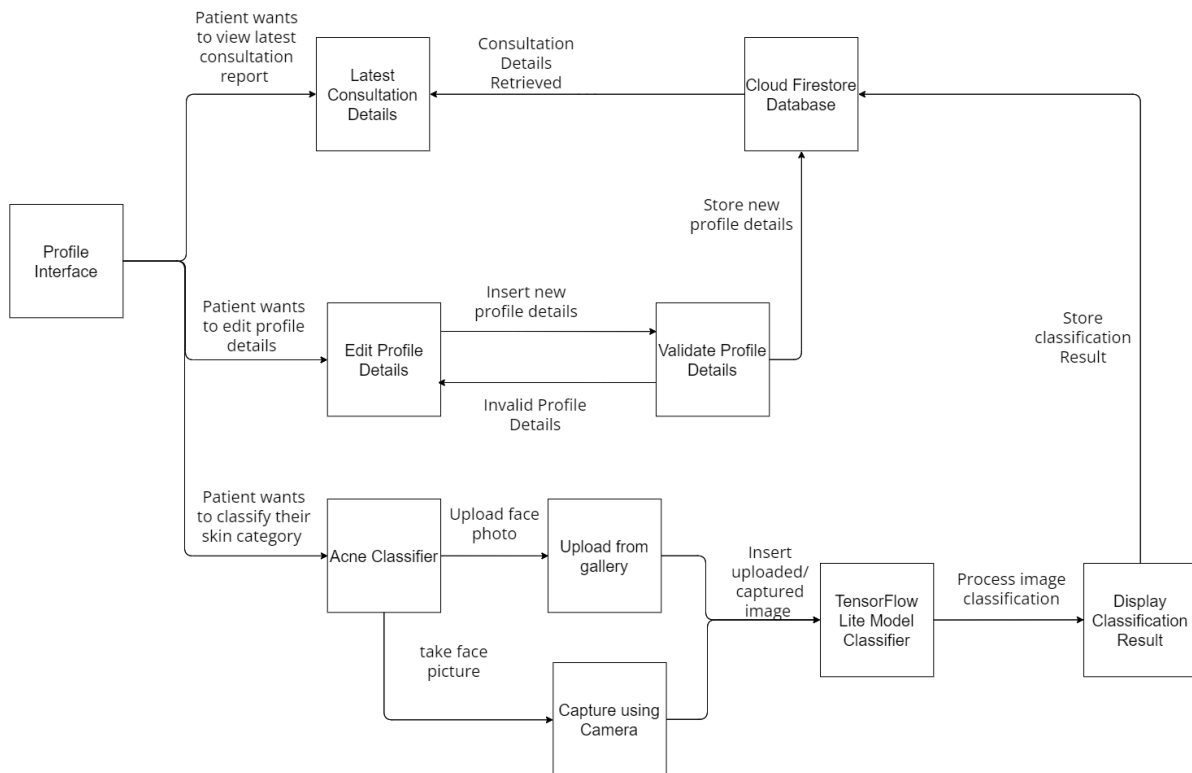
- **Schedule Interface (Patient Module)**

In the upcoming tab, the patient can click on “Reschedule” or “Cancel”. If the patient click “Cancel” the appointment status will be updated and save to the database, then the appointment will move to the cancel tab. If the patient click “Reschedule”, then the page will navigate to the reschedule interface. The patient can choose the new date, time and mode and click on the “Reschedule Appointment” to book the appointment. The patient will further navigate to the confirmation booking interface to further check the validation of the appointment details.

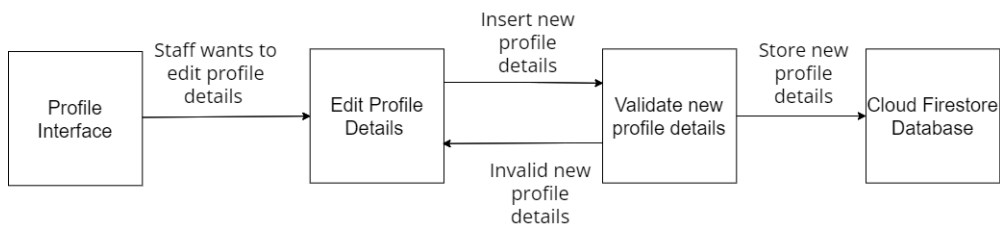
- **Schedule Interface (Staff Module)**

In the upcoming tab, the staff can click on the “View Consultation Details” on the selected appointment card, the staff will be redirect to the consultation details interface in view mode to view the previous consultation details before the consultation session is start so that it will reduce the time on identify the problem of the patient. If the staff is in the complete tab, the staff click on the “View Consultation Details”, the staff will be redirect to the consultation details interface in edit mode and the data displayed will be based on the appointment itself. The data retrieve from the database is according to the appointment ID to display on the interface. The staff can edit the consultation details while in edit mode and the changes will be update to the database when the staff save the changes made.

### 4.3.6 Profile Interface



**Figure 4.3.6.1: Profile Interface Block Diagram for Patient Module**



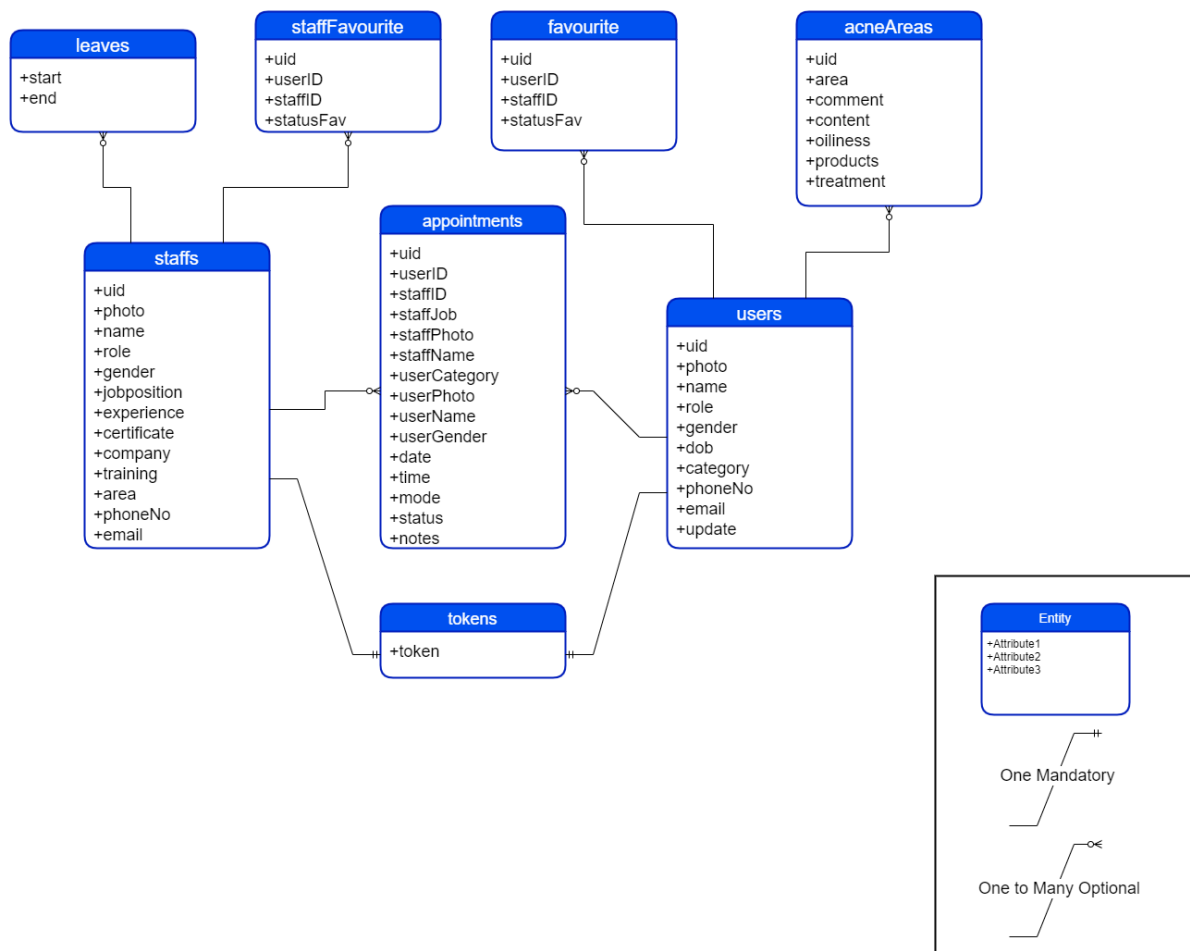
**Figure 4.3.6.2: Profile Interface Block Diagram for Staff Module**

Based on Figure 4.3.6.1 and Figure 4.3.6.2, the design of the profile interface involved the use of Cloud Firestore Database. Upon the user login, the data will be retrieved from the Cloud Firestore Database to display on the profile interface. The profile interface is different from the module because each role is required to perform different task. The common feature in both module is the user can edit their profile details by clicking the profile picture, the user will be redirect to the edit profile interface. The details of the profile will first retrieve from the database and display on the interface. The user can change the details accordingly and click save to save the changes that have been made. If the data is valid, then the data will be update to the database while if the data is not valid, the system will prompt the user to change the data to the valid format. The profile interface of the patient has addition features to be performed.

- **Profile Interface (Patient Module)**

The patient can click on the “AI Scan” button, the patient will be redirect to the classification interface. In the classification interface. The user can choose to capture using camera or upload image from gallery to perform classification and display the result. When the classifier finished the classification process, the interface will then display the result of the classification which is the acne type, and the result will directly save to the database and display on the profile interface with the last update date. The patient can also click on the “View Lates Report” to view the latest consultation details in the view mode. The latest consultation details are retrieved from the database where the details is obtain from the last appointment that have been made.

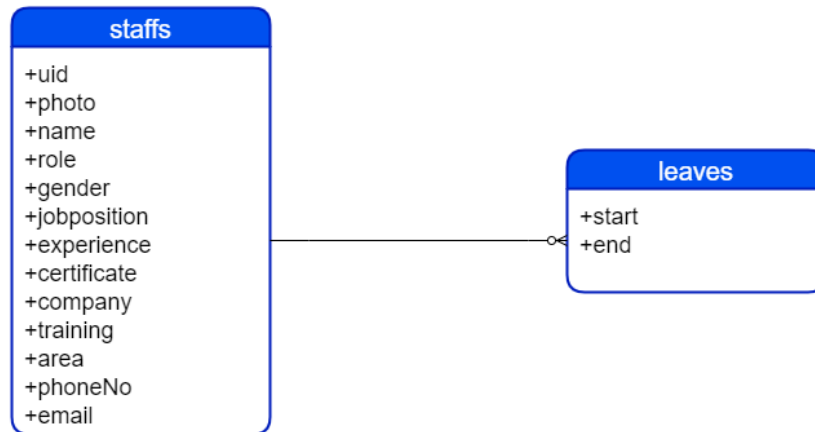
## 4.4 Database Design



**Figure 4.4.1: Database design for Facial and Skincare Service System**

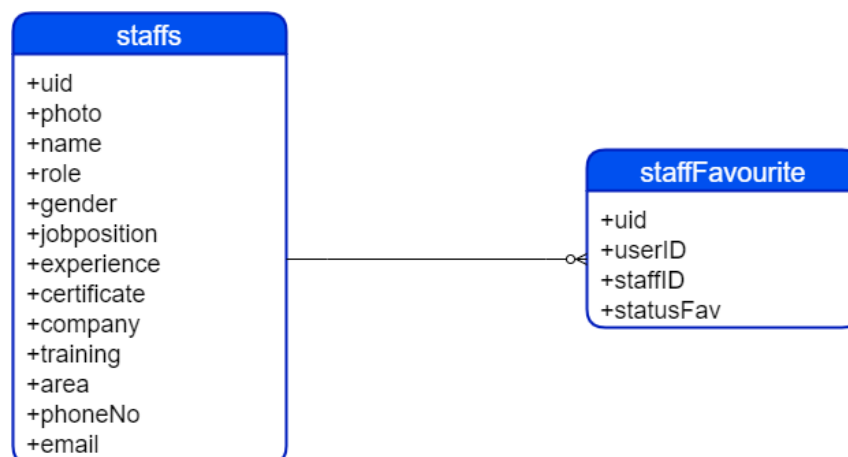
While developing a service system, it is important to have a database design to structure and store the important information in the database storage so that when the retrieval of information can be more efficient. Figure 4.4.1 has illustrated the overview of the database for this service system where it consists of 8 entities in with their respective attributes and relationship between the entities in this database design. The 8 entities in the database design consists of users, staffs, appointments, leaves, staffFavourite, favourite, acneAreas and tokens. The relationship between each entity is explain as follow:

- Staffs has many or optional leaves



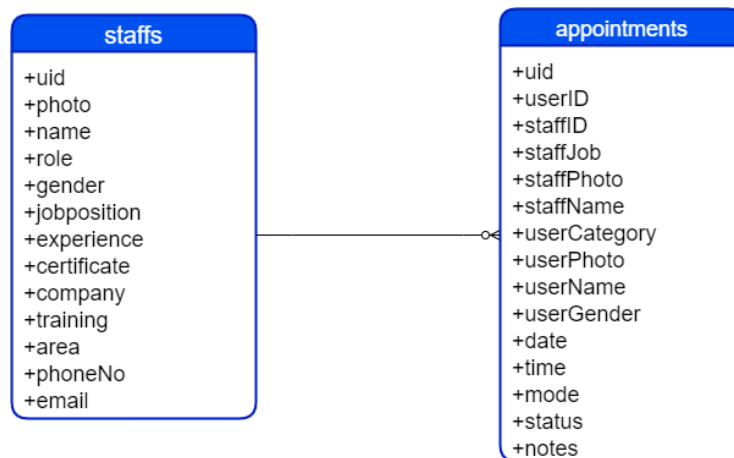
*Figure 4.4.2: Staffs Entity and Leaves Entity*

- Staffs has many or optional staff favourite



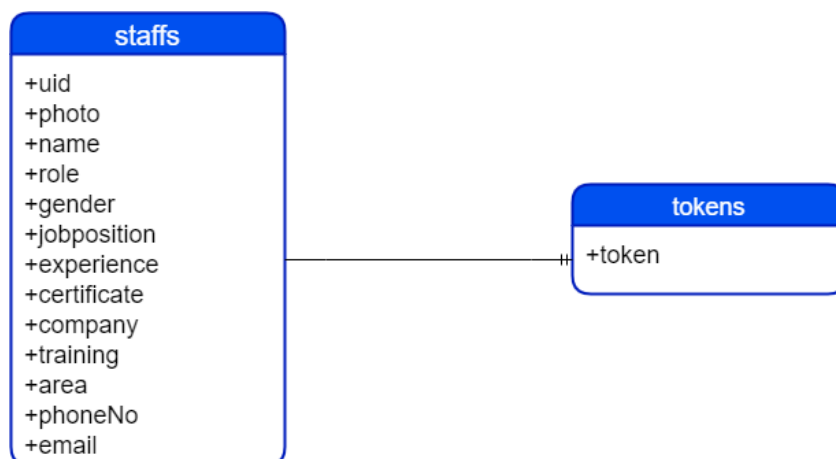
*Figure 4.4.3: Staffs Entity and staff favourite Entity*

- Staffs has many or optional appointments



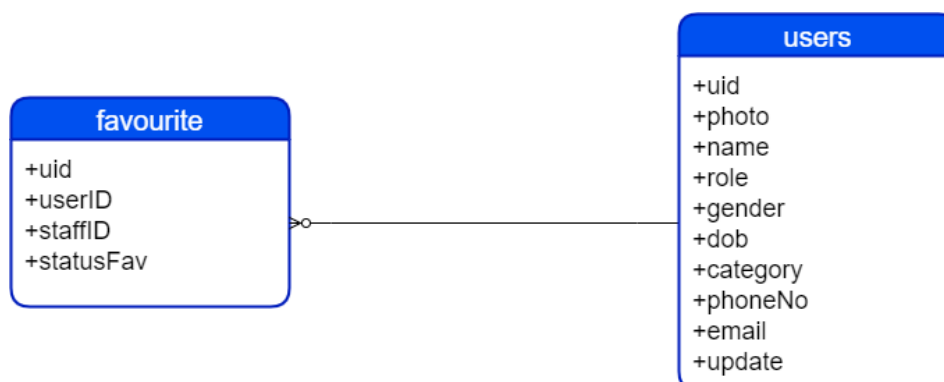
*Figure 4.4.4: Staffs Entity and appointments Entity*

- Staffs has one mandatory token



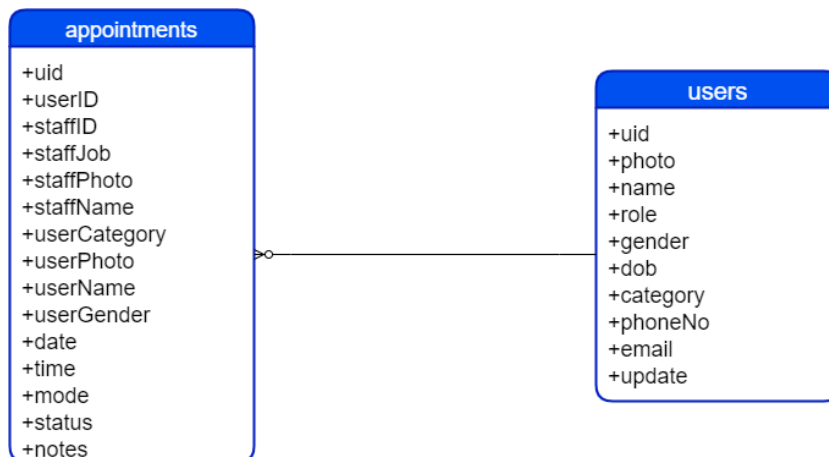
*Figure 4.4.5: Staffs Entity and tokens Entity*

- Users has many or optional favourite



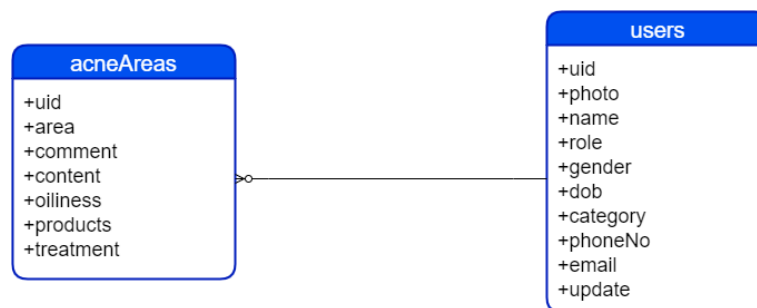
*Figure 4.4.6: Users Entity and favourite Entity*

- Users has many or optional appointments



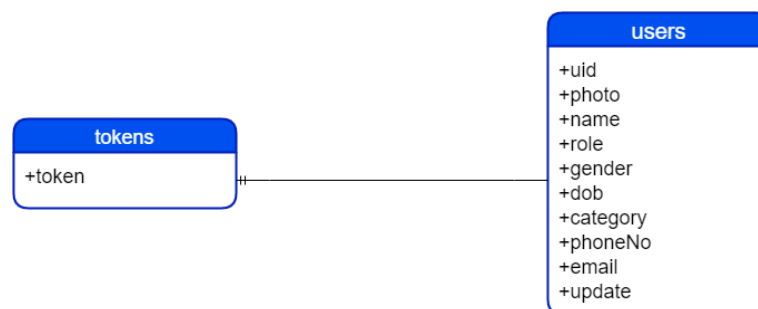
*Figure 4.4.7: Users Entity and Appointments Entity*

- Users has many or optional acne areas



*Figure 4.4.8: Users Entity and acne areas Entity*

- Users has one mandatory token

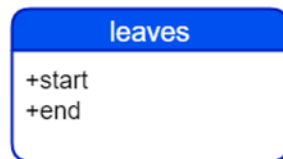


*Figure 4.4.9: Users Entity and tokens Entity*

The detailed explanation for each entity and the relationship between each entity will further elaborate as follow:

- **leaves**

Based on Figure 4.4.10, leaves entity is used to store the leave start date and leave end date from the staff. When the staff apply for leave, they are required to update the on-leave period. Therefore, the start is storing the start date of the leave and the end is storing the end date for the leave. Figure 4.4.11 show the example documents of leaves entity.



*Figure 4.4.10: leave Entity*

```
end: "2023-04-30 00:00:00.000"
start: "2023-04-27 00:00:00.000"
```

*Figure 4.4.11: Example documents for leave entity*

- **staffFavorite**

staffFavorite has 5 attributes which include uid, userID, staffID and statusFav as shown in Figure 4.4.12. This entity is used to store the information of the staff favorite profile where the uid is the unique id for the create favourite list, the userID is the id for the patient that retrieve from the users entity while for the staffID is the unique identity that store the information of the staff, while for statusFav is the status of the profile whether it is in favorite list. Figure 4.4.13 show the example documents of staffFavorite entity.



*Figure 4.4.12: staffFavourite Entity*

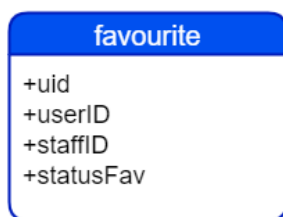
```
staffID: "HJvqZiT2S4XtsYtw98Zc0xKm3112"
statusFav: true
uid: "2CgPCVMA9JzaXw52xAVX"
userID: "O26OeB3wahYYpET5ZDxh0GCRjbx2"
```

*Figure 4.4.13: Example documents for staffFavourite entity*



- **favourite**

favourite is an entity that has the same attribute as the staffFavourite but this entity is storing the favourite list for the patients which is the users in the database. There are 5 attributes which is uid, userID, staffID and statusFav as shown in Figure 4.4.14. The uid is the unique id for the create favourite list that store the favorite profile status, the userID and staff ID is the unique identity that use to track the profile of the staff and patient. The statusFav is to store the boolean record for the status of the profile. while Figure 4.4.15 show the example documents of favourite entity.



*Figure 4.4.14: favourite Entity*

```

staffID: "p1okJqAYEOWrX2Yi0PWlwKvuHk22"
statusFav: true
uid: "lpFCNUewTEvkGVAecIHR"
userID: "026OeB3wahYYpET5ZDxh0GCRjbx2"
  
```

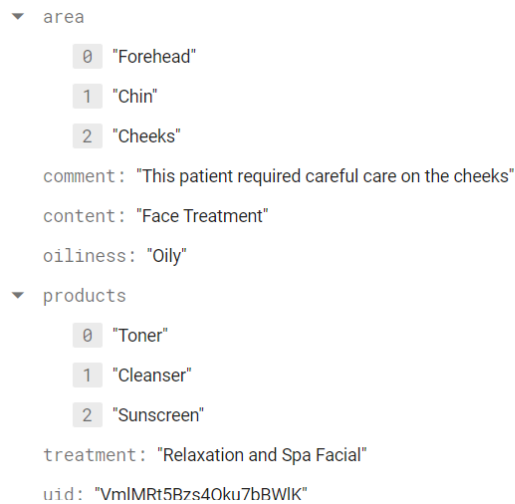
*Figure 4.4.15: Example documents for favourite entity*

- **acneAreas**

This entity is used to store the information for the consultation where the staff is required to complete the consultation details after the consultation session. There are 7 attributes in the entity table that is required to keep track the skin condition of the patient as shown in Figure 4.4.16. The uid is the is the unique identity that referring to the document that store information of the acneAreas while the area is used to store the areas that the acne occurs on the face. The comment is the important note act as a remark for the staff to refer and take note before the consultation session or future use. The content is storing the data for the activities that has been done for that consultation session. Oiliness is the attributes that storing the data for the skin condition of the patient which contains oily, dry and combination. The products attribute is used to keep track the products that has been use by the patients. Lastly is the treatment this entity is used to store the activities that has been done by the patient when the consultation is physical. Figure 4.4.17 show the example documents of acneAreas entity.



**Figure 4.4.16: *acneAreas* Entity**



**Figure 4.4.17: Example documents for *acneAreas* entity**

- **staffs**

staffs entity is used to store the information for each staff. There are 13 attributes in this entity which include the uid, photo, name, role, gender, jobposition, experience, certificate, company, training, area, phoneNo and email as shown in Figure 4.4.18. The uid is a string that record the unique id for the specific staff. The photo is the profile photo that upload by the staff, and it store the image path. Name is the staff name which store as a string in the database. Role attribute is storing the role of the user in the database to determine which interface they will interact with once they enter the system. The attribute gender is storing the data of the “Female” or “Male”. jobposition is an attribute that storing the job position of the staff in a company. The attribute of experience is used to store the information of the staff based on their career experience while certificate is the name of the university or college that they graduated. The company attribute is the name of the company, and the training is storing the information of the place name where the staff carry out the training. Area is the specialization of the staffs while phonenNo is the phone number and email is referred to email address. Figure 4.4.19 show the example documents of staffs’ entity.



**Figure 4.4.18: staffs Entity**

```

area: "Severe Acne"
certificate: "UCSI Diplama in Cosmetic Science "
company: "Face X Skin Beauty Center"
email: "abc122@gmail.com"
experience: "10"
gender: "Male"
jobposition: "Beauty Consultant"
name: "Richard Yan"
phoneNo: "010-578 4646"
photo: "https://firebasestorage.googleapis.com/v0/b/service-system-80467.appspot.com/o/GRUR8YigjVZizfNKUFuvxE1YoMn1%2Fphoto?alt=media&token=02729dba-ac00-4624-b191-a056f81201be"
role: "Staff"
training: "L'Oréal Ignite "
uid: "GRUR8YigjVZizfNKUFuvxE1YoMn1"
workID: "W3005748"

```

**Figure 4.4.19: Example documents for staffs entity**

- **appointments**

Based on Figure 4.4.20, there are 15 attributes in the appointment entity. Uid is refer to the unique identity for the document that storing the information of the appointment. userID is the unique identity from the users entity while staff ID is from the staffs entity. staffPhoto is the image path that retrieve form the staffs entity while staffName is the name for the staff. userCategory is the acne category for the patient. userPhoto is the image path that retrieve from the users entity while username is the name of the user also known as patient. userGender is the gender for the user that retrieve from the users entity. date is referred to the date of the appointment booked while the time is the booking time, and the mode is either physical or online. The status is used to store the status of the appointment whether it is an upcoming appointment, completed appointment or cancelled appointment, the notes is storing the unique id that refer to the document in the acneAreas entity. Figure 4.4.21 show the example documents of appointment entity.



**Figure 4.4.20:**  
**appointments Entity**

```

date: "4/17/2023"
day: "Monday"
mode: "Physical"
notes: "y2rxEl1ZlRtzGrNk6ns1"
staffID: "HJvqZiT2S4XtsYtw98Zc0xKm3112"
staffJob: "Lead Beauty Consultant"
staffName: "Natalie Yeong"
staffPhoto: "https://firebasestorage.googleapis.com/v0/b/service-system-80467.appspot.com/o/HJvqZiT2S4XtsYtw98Zc0xKm3112%2Fphoto?alt=media&token=8d7a6ae4-2e31-4618-8d49-312627b2890d"
status: "complete"
time: "9:00 AM"
uid: "VayJcaPoDONQsnQjsyKF"
userCategory: "category"
userGender: "Female"
userID: "O26OeB3wahYYpET5ZDxh0GCRjbx2"
userName: "Camile Jerry"
userPhoto: "https://firebasestorage.googleapis.com/v0/b/service-system-80467.appspot.com/o/O26OeB3wahYYpET5ZDxh0GCRjbx2%2Fphoto?alt=media&token=66715fb5-1984-43fd-b6a0-da449e8f220c"

```

**Figure 4.4.21: Example documents for appointments entity**

- **users**

Figure 4.4.22 shows the users entity that contains 10 attributes where the uid is the unique id for the document that store the users information while the photo is the image path that the patient uploaded the profile photo, the name is referred to the users' name and the role is use to identify the interface of the system for the user to perform specific task. The gender is determined whether the users is "Female" or "Male" and dob is the date of birth for the users. Acne category is store in the attribute of category, while the phoneNo is the phone number for the users and email is referred to the email address. Lastly, the update is the date store when the users update the category which is the acne category using the acne classification feature. Figure 4.4.23 show the example documents of users entity.



**Figure 4.4.22: users Entity**

```

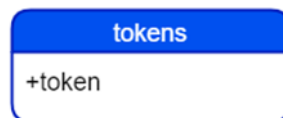
category: "Moderate Acne"
dob: "February 12, 2002"
email: "abc@gmail.com"
gender: "Female"
name: "Camile Jerry"
phoneNo: "014- 973 4460"
photo: "https://firebasestorage.googleapis.com/v0/b/service-system-80467.appspot.com/o/O26OeB3wahYYpET5ZDxh0GCRjbx2%2Fphoto?alt=media&token=6141c7c4-b0fd-43b6-a0ac-5199def3385a"
role: "User"
uid: "O26OeB3wahYYpET5ZDxh0GCRjbx2"
update: "4/24/2023"

```

**Figure 4.4.23: Example documents for users entity**

- **token**

Based on Figure 4.4.24, there is only one attribute in the tokens entity. the token attribute is used to store the unique token from the user device. This token is important to use in the Firebase Cloud Messaging to specify the receiver of the notification. Figure 4.4.25 show the example documents of tokens entity.



**Figure 4.4.24: tokens Entity**

```

token: "ciJyd96xSUEXV88a-MDG-R:APA91bFbwjTArd174_zx2pOZnL57zoVYc85gtZKyfVWbMgw4_U9FaNmlMQ1j11T7OGsl8kMNMm2d8jHq5g-6YR-9VCbdhPaVag_1S4DyhSq187C-QKAXCrTXE8Kmop_FEKO2w3mOMV2L"

```

**Figure 4.4.25: Example documents for tokens entity**

# Chapter 5

## System Implementation

### 5.1 Hardware Setup

- **Laptop**

Description	Specifications
Model	ASUS TUF Gaming A15
Processor	AMD Ryzen 7 4800H
Operating System	Windows 11
Graphic	NVIDIA GeForce GTX 1650 Ti
Memory	24GB DDR4 RAM
Storage	512GB SATA SSD

*Table 5.1.1: Laptop used for system development*

- **Mobile Device**

Description	Specifications
Model	Huawei Mate 20
Processor	Huawei Kirin 980
Operating System	Android 12 with EMUI 12.0.0
RAM	6.0GB
Internal Storage	128 GB

*Table 5.1.2: Smartphone used for system development*

## 5.2 Software Setup

Before the development for the system, there are some software needed to be installed and set up:



*Figure 5.2.1: Visual Studio Code Logo*

### 1. Visual Studio Code v1.73.1 [20]

Visual Studio Code which is also known as VS code is a code editor that developed by Microsoft with the Electron Framework for building and debugging modern web and cloud application. I will create the facial and skincare service system using Visual Studio Code to run the Dart programming to visualize the output.



*Figure 5.2.2: Flutter Logo*

### 2. Flutter v3.3.8 [21]

Flutter is the new platform that develop to build specially for the building of mobile application which can support both iOS and android operating system. I will use the Flutter as the platform to build the overall UI design in the Visual Studio Code.



*Figure 5.2.3: Dart Logo*

### **3. Dart v2.18.4 [22]**

Dart is a widely used programming language created by Google in 2011 that is mainly used for developing web, mobile, and server-side applications. The language is statically typed but also provides support for dynamic typing. It can be compiled to native code for different platforms, making it highly versatile.



*Figure 5.2.4: TensorFlow Logo*

### **4. TensorFlow v2.12.0 [23]**

Google's TensorFlow is a potent open-source library for artificial intelligence that utilizes data flow graphs to create models. Developers can use it to build multi-layered neural networks, which can perform tasks such as classification, perception, understanding, discovering, prediction, and creation. TensorFlow has a wealth of tools and functions to create intricate machine learning models and analyse data at scale. It is used extensively in various industries and academic fields to develop sophisticated AI applications, such as natural language processing, computer vision, and deep learning.





*Figure 5.2.5: Keras Logo*

## **5. Keras v2.12.0 [24]**

Keras is a user-friendly deep learning tool that simplifies the process of building complex models. It is built using TensorFlow 2, which is a robust machine learning platform. Keras is easy for beginners to understand and navigate, allowing them to develop models for tasks such as image classification, object detection, and natural language processing. Furthermore, it is highly adaptable, meaning developers can personalize their models to fit their specific requirements. This flexibility and its simple interface have made Keras a favourite among those working in artificial intelligence.

## 5.3 Setting and Configuration

### 5.3.1 Flutter Android Configuration

#### Configuration and Setting for Build Gradle Project File

- **Dependencies**

The dependencies of the Build Gradle project file needs to be updated to integrate the service from the Firebase platform as shown in Figure 5.3.1.1.

```
buildscript {
    ext.kotlin_version = '1.6.10'
    repositories {
        google()
        mavenCentral()
    }

    dependencies {
        classpath 'com.android.tools.build:gradle:7.3.1'
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"
        classpath 'com.google.gms:google-services:4.3.13'
    }
}
```

*Figure 5.3.1.1: Update the Dependencies in Build Gradle*

- **SDK Version**

The minimum version of SDK requirement to run the mobile application is 21 to maximum 31 SDK version.

```
android {
    compileSdkVersion 33
    ndkVersion flutter.ndkVersion

    defaultConfig {
        applicationId "com.example.servicesystem"
        minSdkVersion 21
        targetSdkVersion 31
        versionCode flutterVersionCode.toInteger()
        versionName flutterVersionName
        multiDexEnabled true
    }
}
```

*Figure 5.3.1.2: Update the SDK Version in Build Gradle*

- **AaptOptions**

AaptOptions is a class in the Android Gradle Plugin (AGP) that is added into the build gradle file to integrate the TensorFlow Lite in the Flutter app.

```
aaptOptions {  
    noCompress 'tflite'  
    noCompress 'lite'  
}
```

*Figure 5.3.1.3: Add In the aaptOptions in Build Gradle*

### **Permission Access in Android Manifest File**

The permission access for device is required to develop this project to request the access to internet support, device storage access, camera access and so forth as shown in

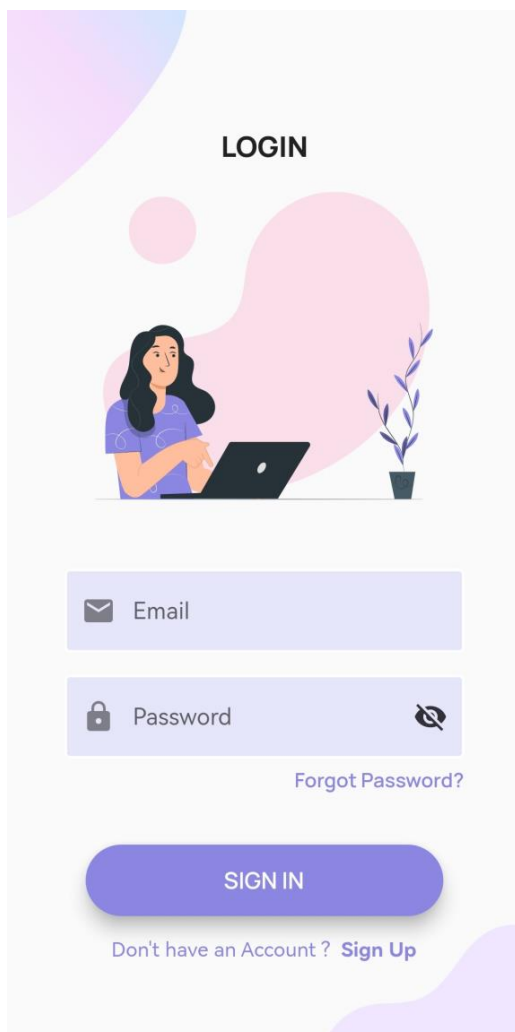
```
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />  
<uses-permission android:name="android.permission.RECORD_AUDIO" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.CAMERA" />  
<uses-permission android:name="android.permission.BLUETOOTH" />  
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />  
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />  
<uses-permission android:name="android.permission.WAKE_LOCK" />  
<uses-permission android:name="android.permission.VIBRATE"/>
```

*Figure 5.3.1.4: Permission Access Used*

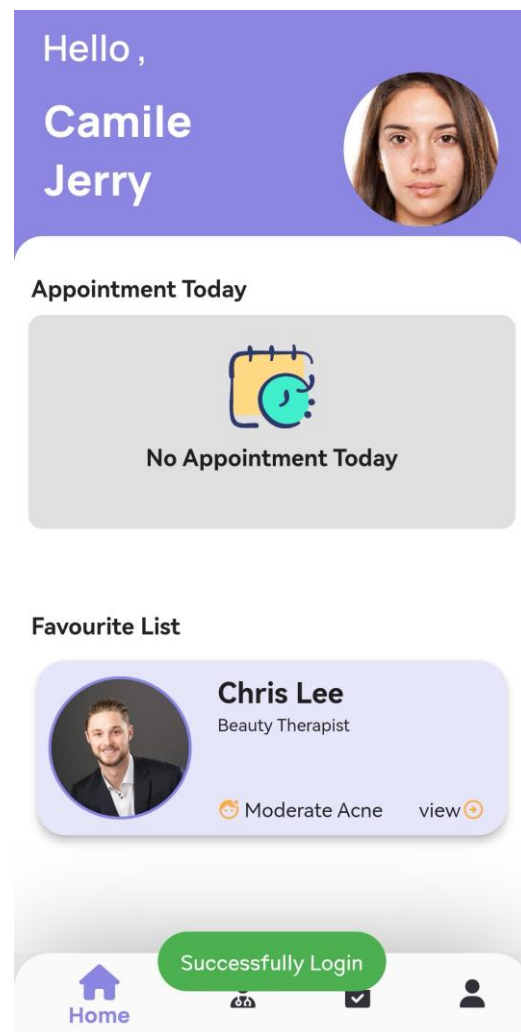
## 5.4 System Operation

### 5.4.1 Account Login

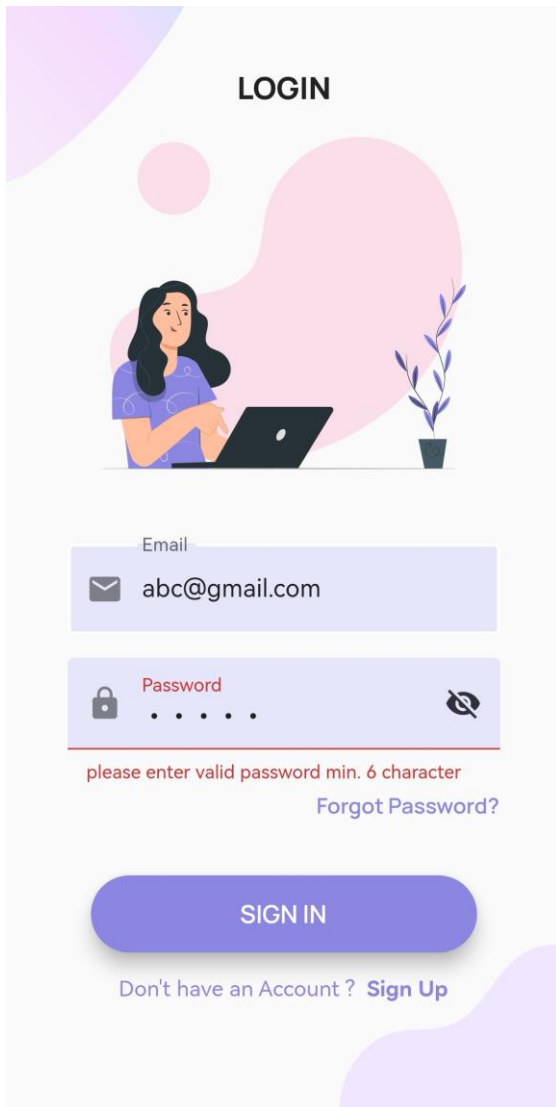
Figure 5.4.1.1 When the users launch the application the first page displayed on the screen is Login page if they have logout their account and wants to login again. The users need to enter their login credentials such as email and password into the text fields correctly. Once the user finished filling in the required fields, they need to click on the “Sign In”. After pressing the button, the system will verify it and display a toast message to inform the user whether they successfully login to the system as shown in Figure 5.4.1.2. An error message will be shows if the user provides the wrong login credentials and required them to re-enter the email address and password to proceed the login procedure as shown in Figure 5.4.1.3 and Figure 5.4.1.4 for different error message that display on the screen.



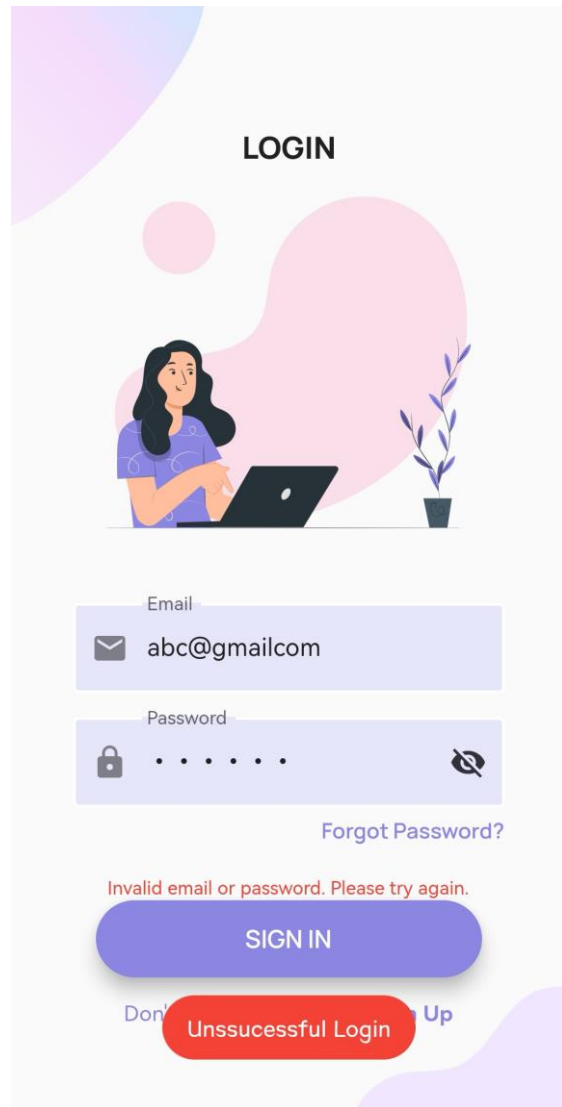
**Figure 5.4.1.1: Login Interface**



**Figure 5.4.1.2: Account Login verification  
- Successful**



**Figure 5.4.1.3: Account Login verification for password - Unsuccessful**

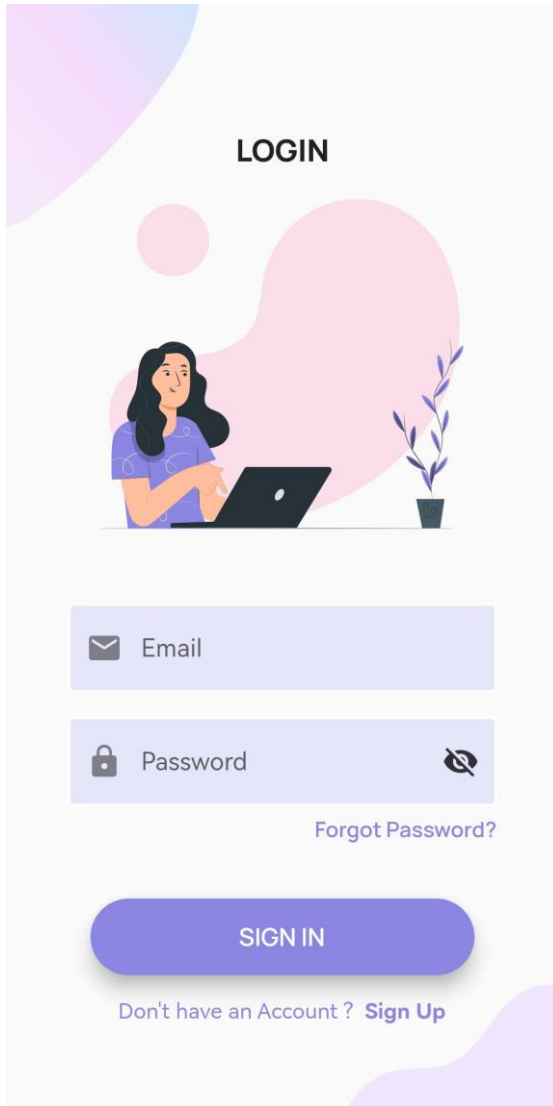


**Figure 5.4.1.4: Account Login verification for email address - Unsuccessful**

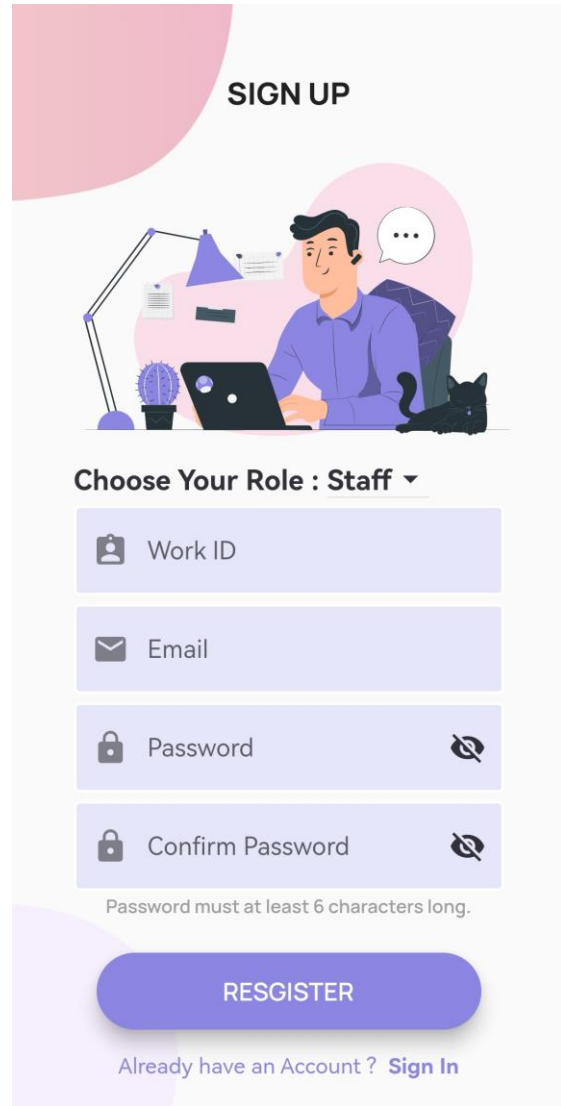
## 5.4.2 Staff Module

- **Account Registration**

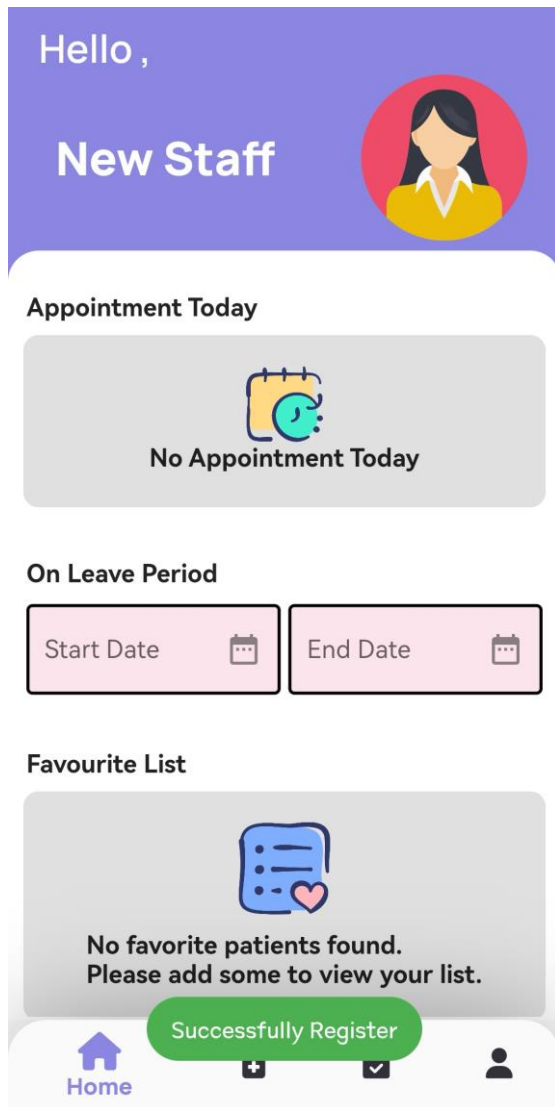
When the staff launches the application, the staff will first interact with the Login interface as shown in Figure 5.4.2.1. The staffs need to click on the “Sign Up” at the bottom of the “Sign In” button of the mobile application if they want to register for an account. When the staffs entered the Sign-Up interface, the staffs need to select their role to register to the system. For the staff they need to select their role as staff as shown in Figure 5.4.2.2. Once the staffs have entered the work id, email address, password and confirm password, the staffs are needed to press on the “Register” button to complete the registration. After pressing the button, the system will verify it and display a toast message to inform the staff whether the registration has been recorded as shown in Figure 5.4.2.3 and Figure 5.4.2.4. If the registration successful, then it will display “Successfully Register” and redirect the staff to the home interface of the system as shown in Figure 5.4.2.3 while unsuccessful of registration will be showing “Unsuccessful Registration” and an error message which requires the staff to entered again the details as shown in Figure 5.4.2.4.



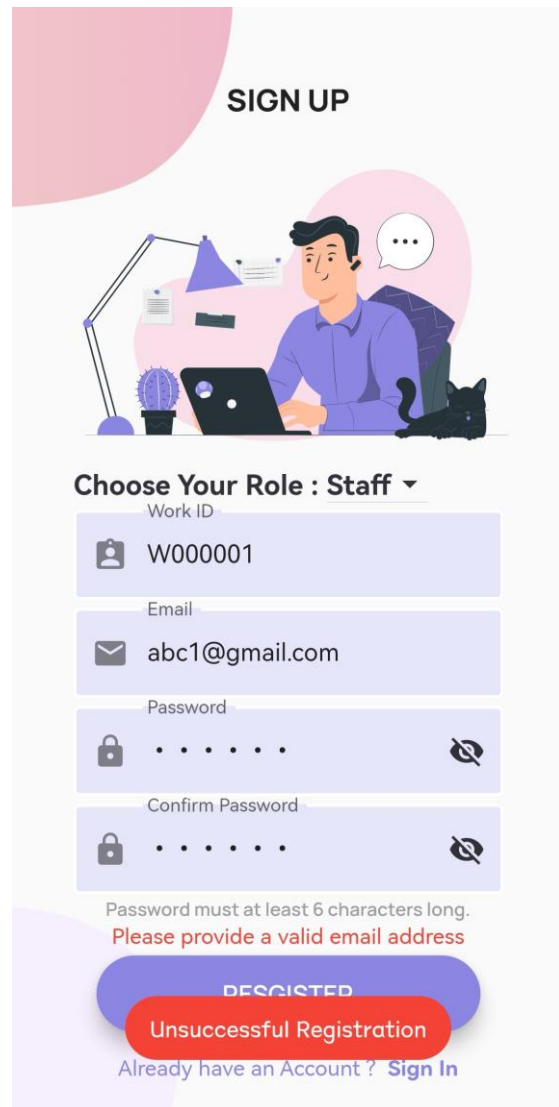
**Figure 5.4.2.1: Login Interface**



**Figure 5.4.2.2: Sign Up Form for Staff**



**Figure 5.4.2.3: Staff Account registration verification - Successful**



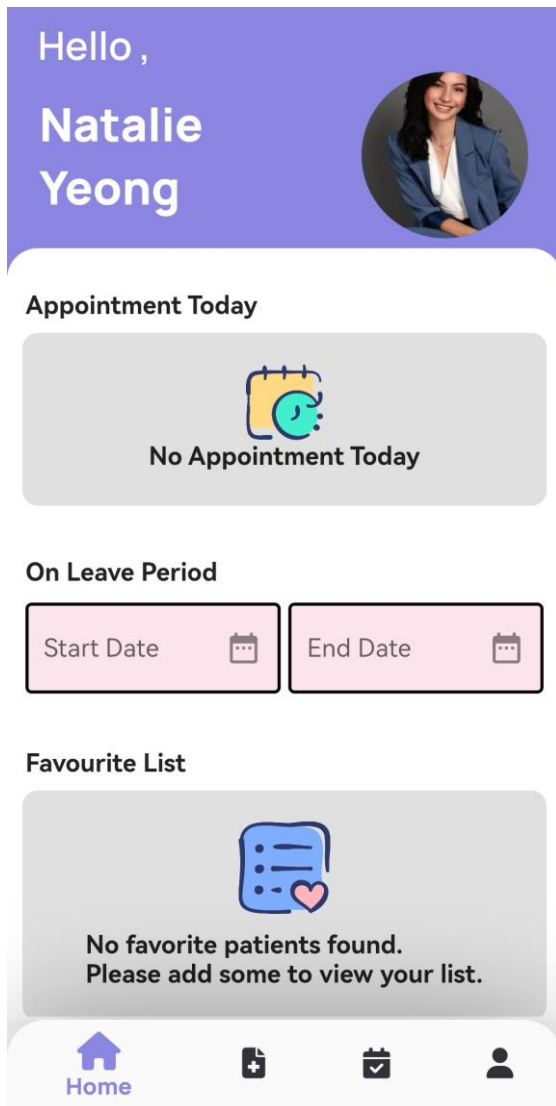
**Figure 5.4.2.4: Staff Account registration verification - Unsuccessful**

- **On Leave Period**

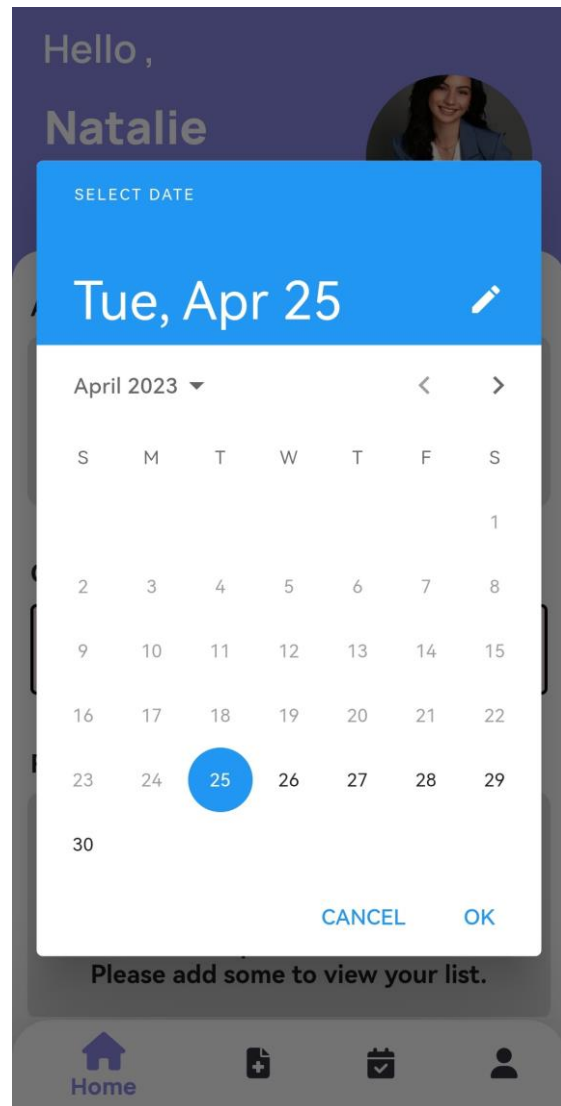
On Leave Period is specially design to the staff to manage the booking calendar by inserting the start date and end date that available in the homepage as shown in Figure 5.4.2.5. The staff can press on the calendar icon at the text fields of the with the hint text label as start date. A calendar will show up and the staff can pick a date from the calendar by pressing the day on the calendar as shown in Figure 5.4.2.6. After picking a day, the staff is required to press on “OK” to complete the process of choosing start date. A toast message will be shows when the staff press “OK” to alert the staff to proceed to end date to complete the on-leave process as shown in Figure 5.4.2.7. Next, the staff is required to pick a date from the end date calendar,



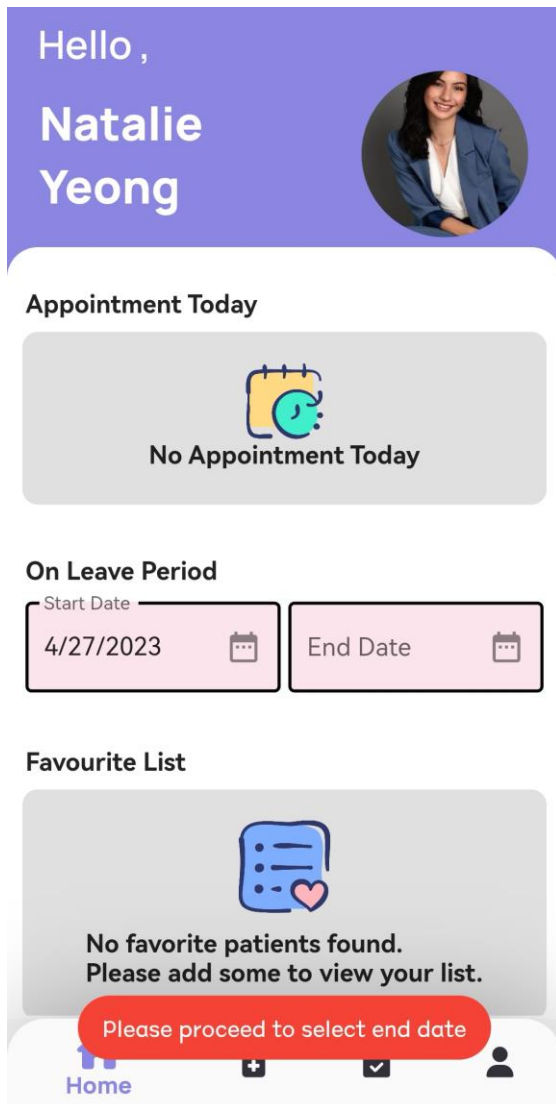
then press on “OK” to complete the process of on-leave period. The start date and end date will display in the text field when the date is picked as shown in Figure 5.4.2.8.



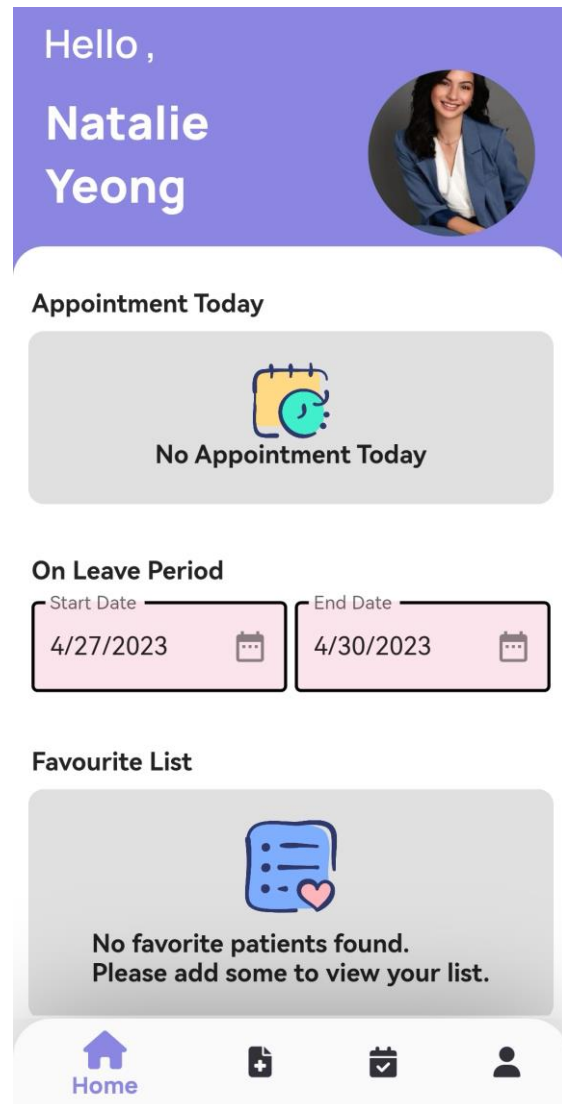
*Figure 5.4.2.5: On Leave Period in Homepage*



*Figure 5.4.2.6: Calendar Picker to pick start date and end date*




**Figure 5.4.2.7: Toast Message to alert user**

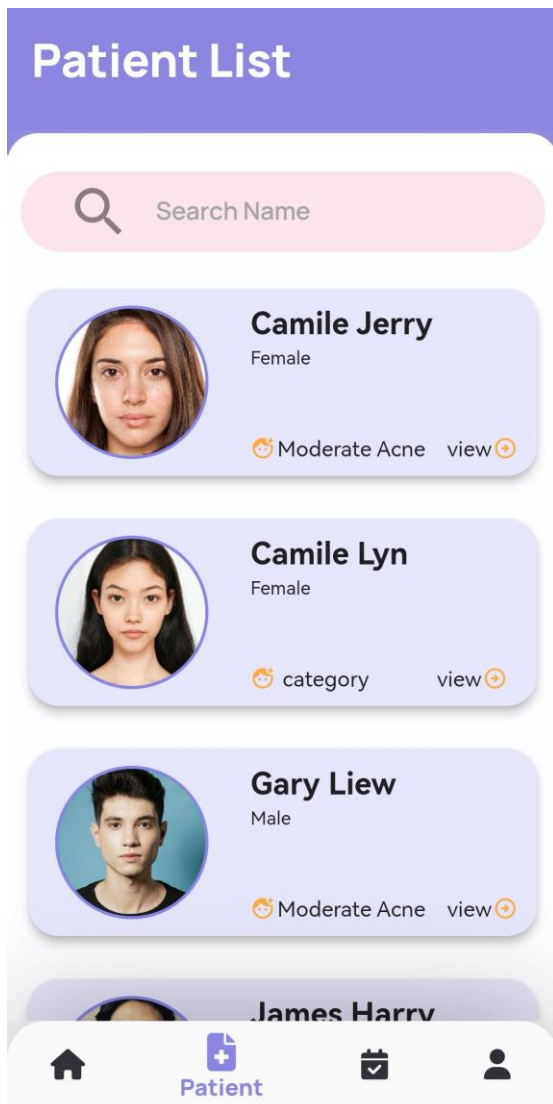


**Figure 5.4.2.8: Start Date and End Date shows in the text field**

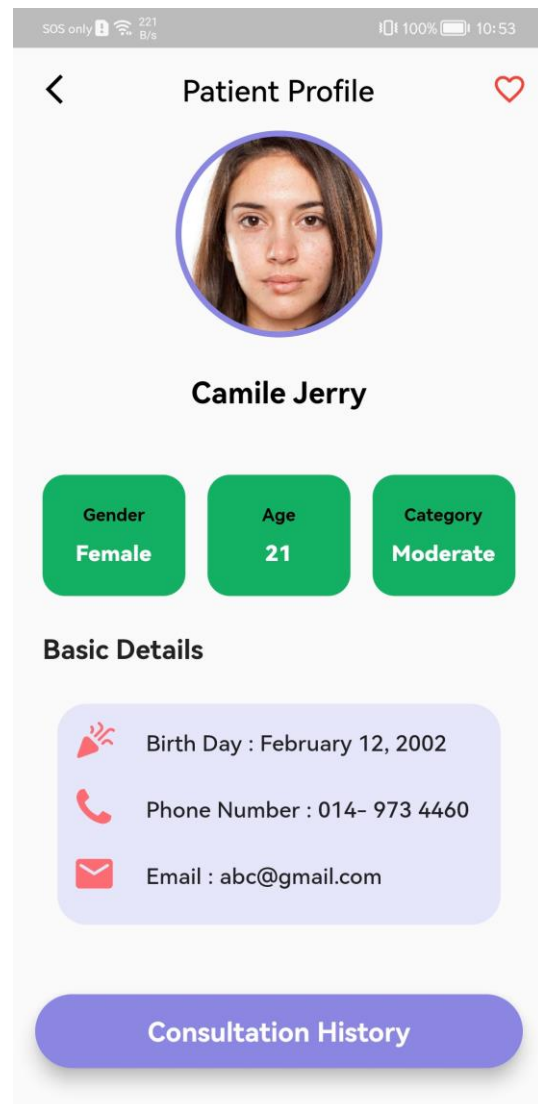
- **Patient List**

When the staff want to view the patient list that exist in the database, the staff can press on the “” icon. Once the staff is pressing the button, the page will navigate to patient list interface where the list of the patient is displayed as shown in Figure 5.4.2.9. The staff can scroll through the list to view the patient details. When the staff select on a patient to view the profile, the staff will be redirect to the patient profile details page where the detail of the patient is displayed as shown in Figure 5.4.2.10. The staff can also press on the “Consultation History” button to view the consultation history of a patient on consultation history page as shown in Figure 5.4.2.11. If the patient does not have any consultation history, the consultation history page

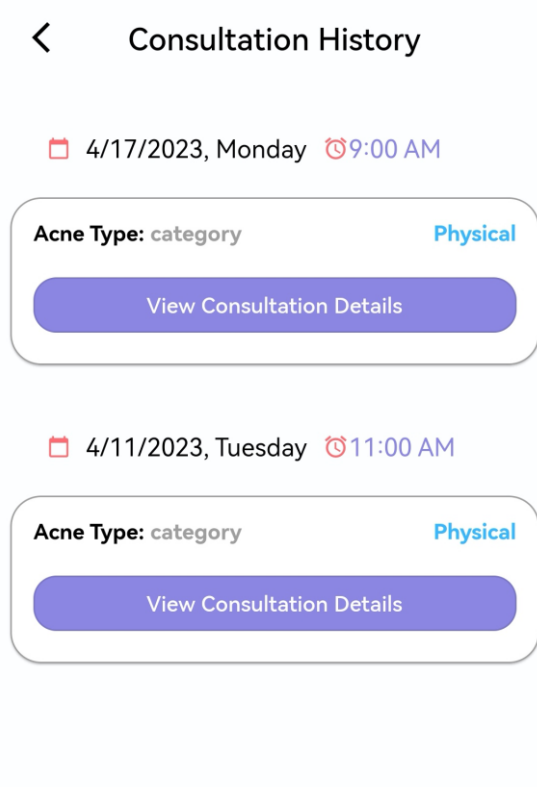
will show the message “No Consultation History Found” to indicates that there is no consultation history for that patient as shown in Figure 5.4.2.12.



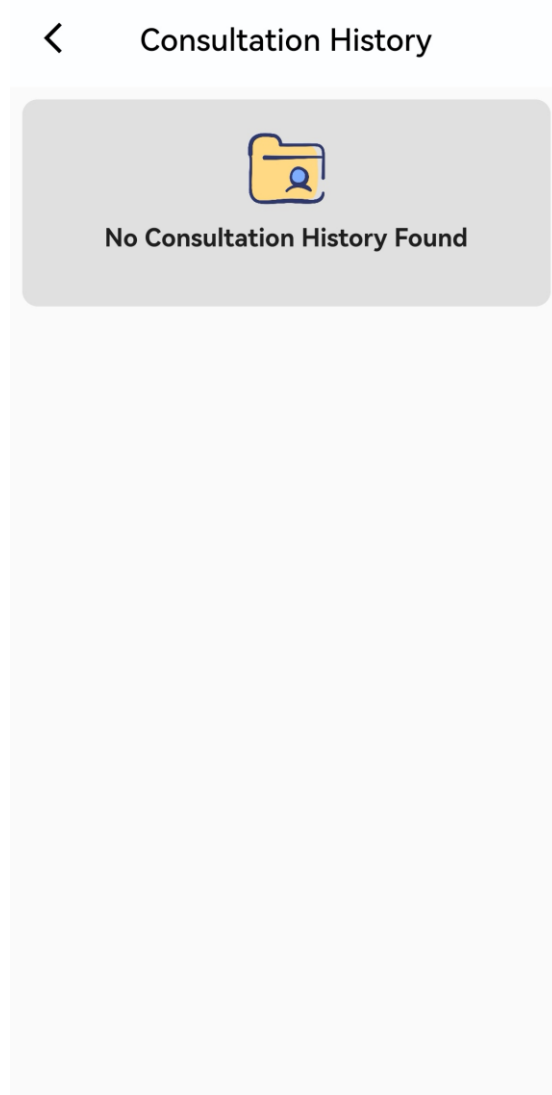
*Figure 5.4.2.9: Patient List View*



*Figure 5.4.2.10: Patient Profile Details*



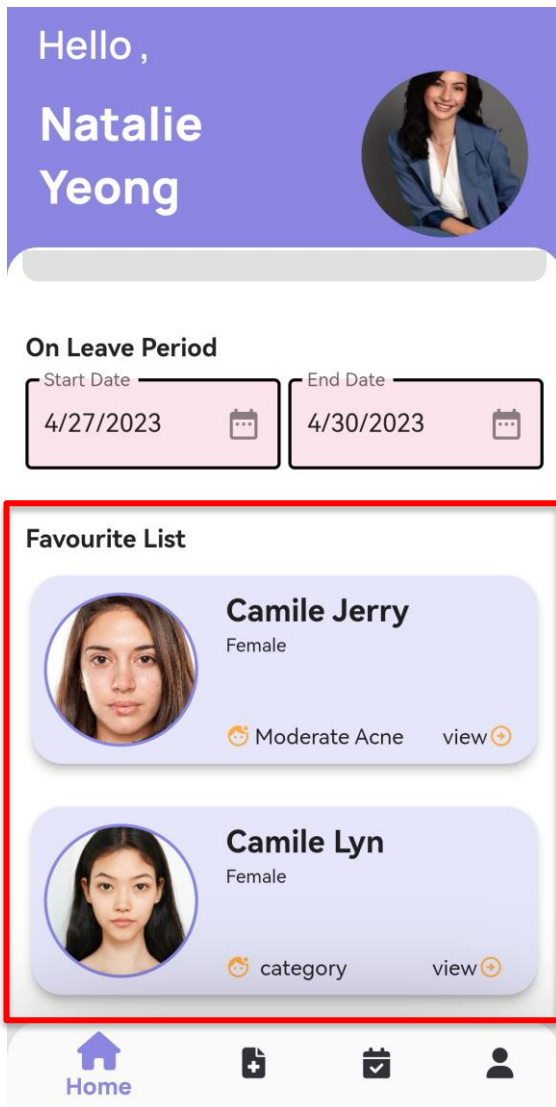
**Figure 5.4.2.11: List of Consultation History**



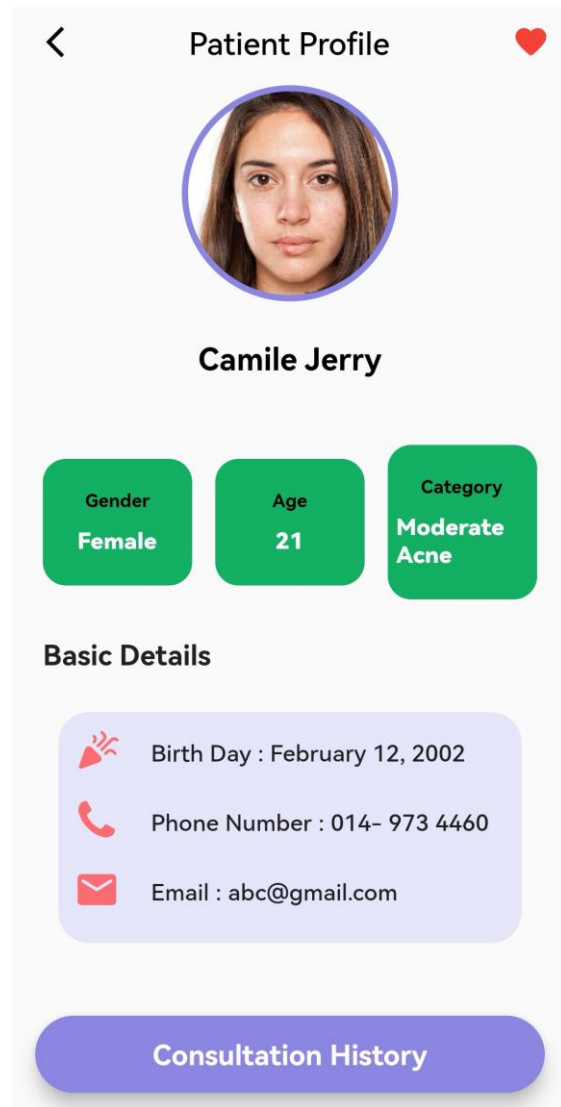
**Figure 5.4.2.12: No Consultation History Message**

- **Favorite List**

The staff can view the favorite list in the homepage which is below the “On Leave Period” section as shown in Figure 5.4.2.13. The staff can scroll the favorite list if there are more than two favorite profiles. The staff can click on the profile card, then the page will navigate to the patient profile details page. In the patient profile details page, the heart icon on the top right is filled with red color to indicate that the profile is in the favorite list as shown in Figure 5.4.2.14.



**Figure 5.4.2.13: Favorite List Section in Homepage**

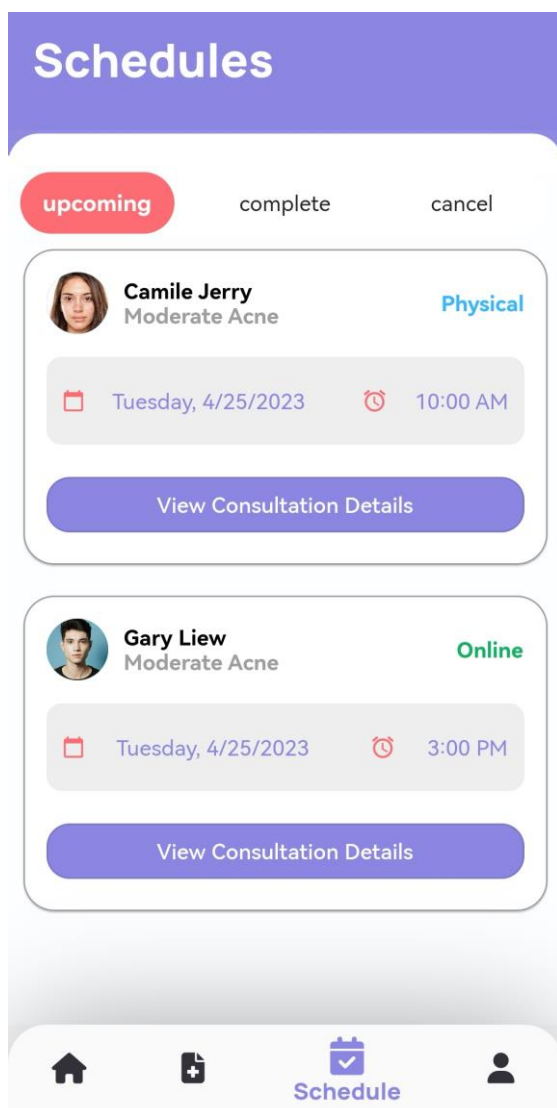


**Figure 5.4.2.14: Patient Profile Details for Favorite List**

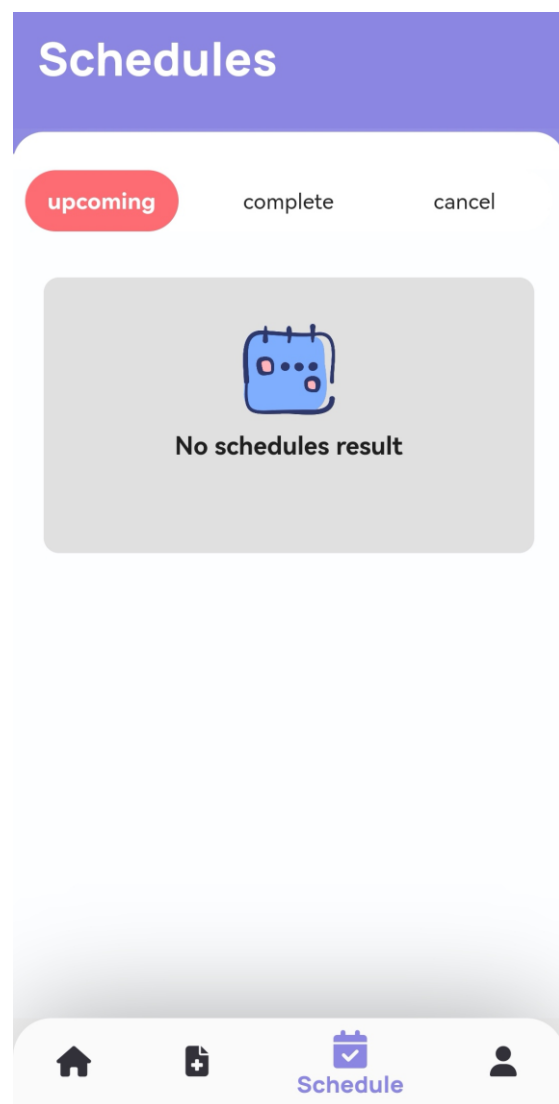
- **Schedules**

When the staff want to view the schedule list that exist in the database, the staff can press on the “📅” icon. . Once the staff is pressing the button, the page will navigate to schedule list interface where the list of the appointment from the patient is displayed when the list of schedules is not empty as shown in Figure 5.4.2.15. The staff will first see the upcoming schedule interface, and there are two other tab which is complete and cancel. When the schedule result is empty, a message will show “No schedules result” to indicates that no appointment from the patient as shown in Figure 5.4.2.16. The list of the schedule is different in each tab where the list will have “View Consultation Details” for the list in upcoming tab to view the last consultation details as shown in Figure 5.4.2.17, while for the list in complete tab,

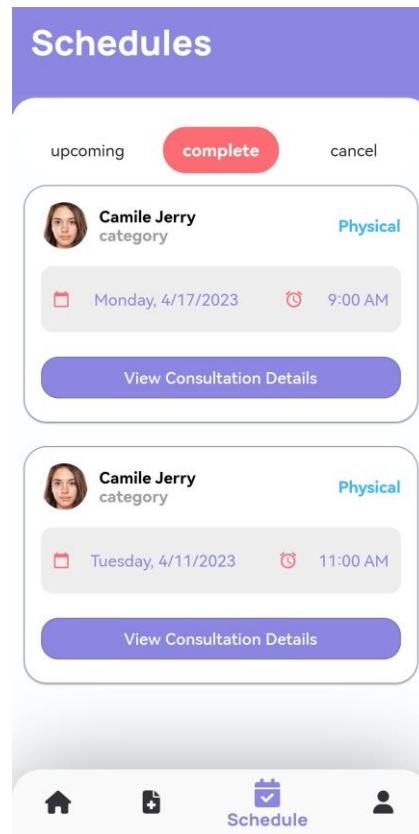
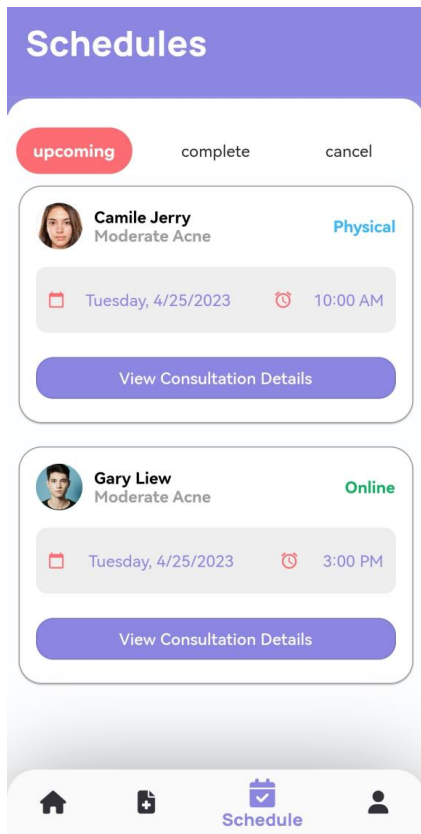
the list will have “View Consultation Details” button which allow the staff to view the consultation history based on the appointment as shown in Figure 5.4.2.18. In the cancel tab, the staff can only view the list of the canceled appointment as shown in Figure 5.4.2.19. When the staff press on the “View Consultation Details” in upcoming page, the staff will redirect to the consultation details for the last appointment as shown in Figure 5.4.2.20. If the staff click on the “View Consultation Details” in complete tab, the staff will be redirect to the consultation details for the selected appointment as shown in Figure 5.4.2.21.



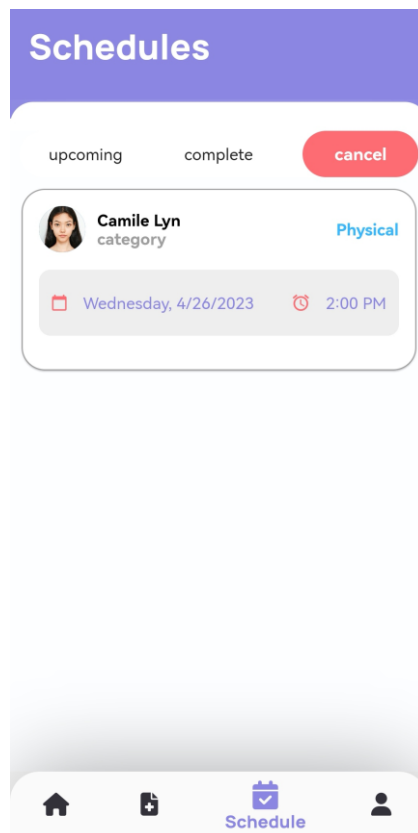
*Figure 5.4.2.15: List of schedules - Staff*



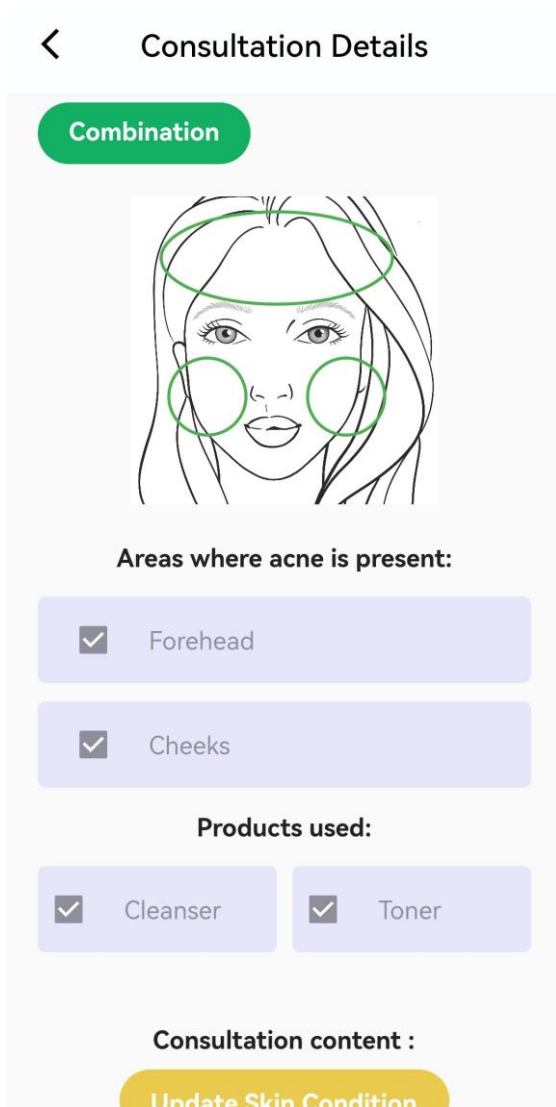
*Figure 5.4.2.16: List of schedules is empty - Staff*



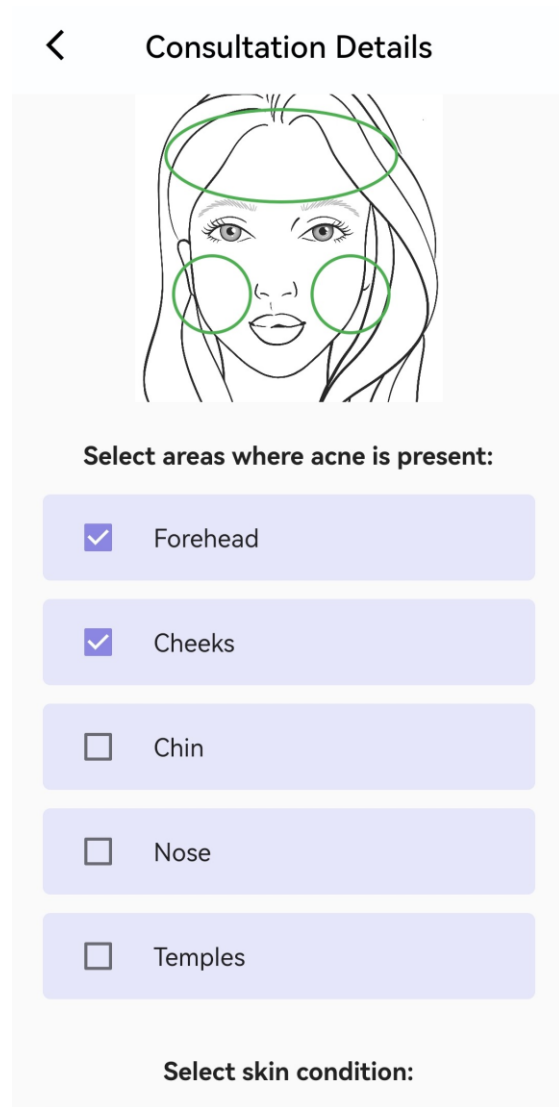
**Figure 5.4.2.17: Upcoming Interface -Staff**    **Figure 5.4.2.18: Complete Interface - Staff**



**Figure 5.4.2.19: Cancel Interface - Staff**



*Figure 5.4.2.20: Consultation Details for last appointment*

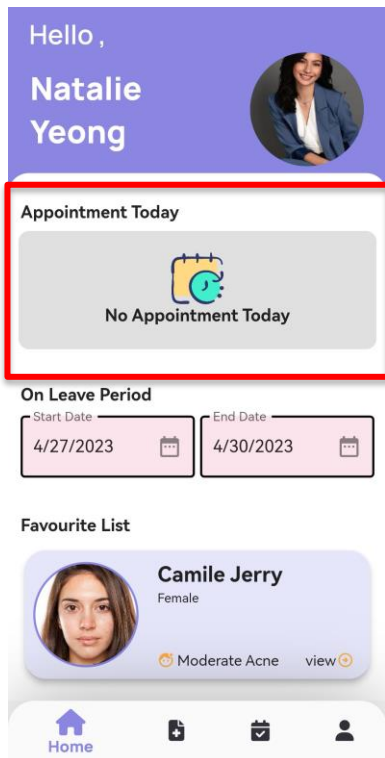


*Figure 5.4.2.21: Consultation Details for selected appointment*

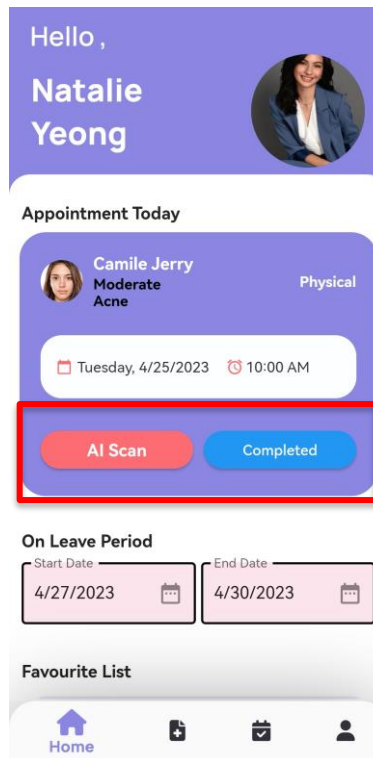
- **Appointment Today**

Appointment Today is a section of the features that available in the homepage. When there is not appointment for the current date, then the message will display “No appointment Today” to inform the staff that he/she does not have any consultation session need to conduct as shown in Figure 5.4.2.22. If there is an appointment on current date, the appointment card will be displayed. There are two types of appointment card based on the appointment mode. When the appointment mode is physical the button shows in the appointment card is “AI Scan” and “Completed” as shown in Figure 5.4.2.23. If the appointment mode is online, then the button will change to “Join Call” and “Completed” as shown in Figure 5.4.2.24.

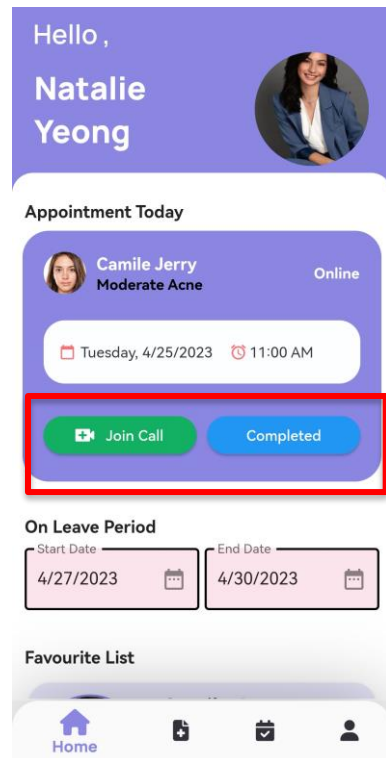




**Figure 5.4.2.22: No Appointment Today**




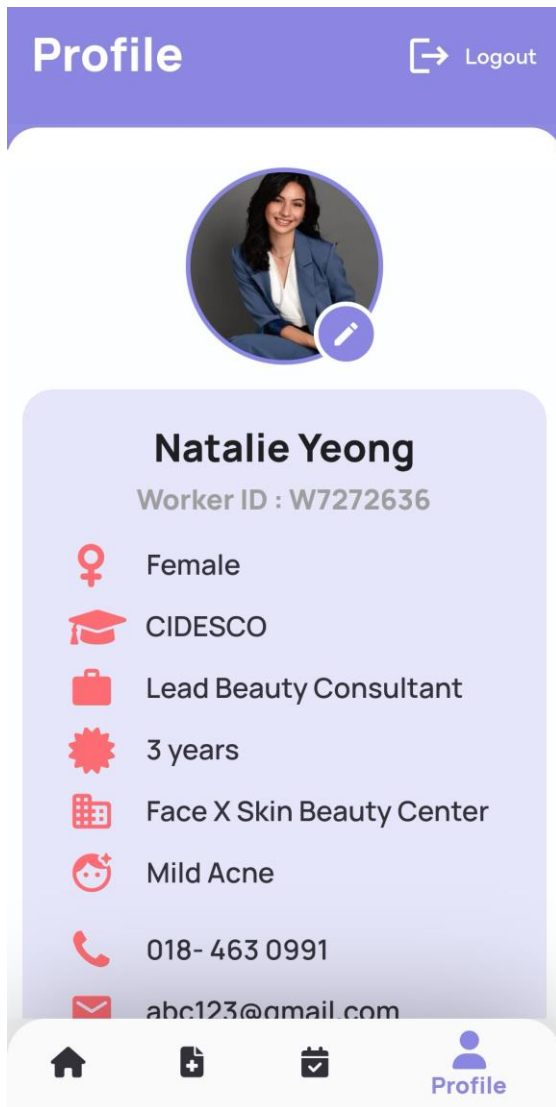
**Figure 5.4.2.23: Appointment card for physical mode**



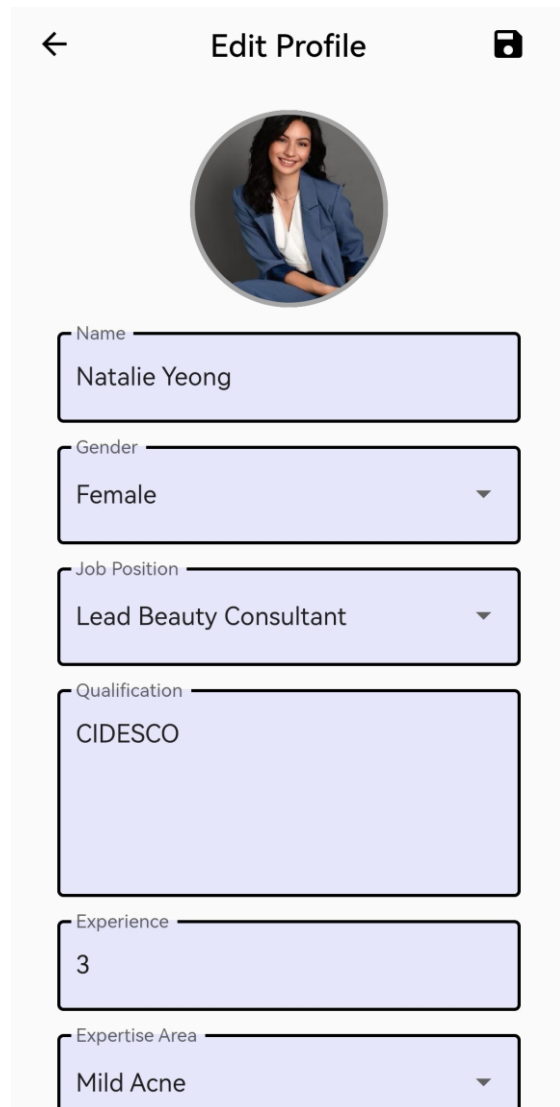
**Figure 5.4.2.24: Appointment card for online mode**

- **Profile**

When the staff want to view their profile details, the staff can press on the “  ” icon. . Once the staff is pressing the button, the page will navigate to profile details interface where the details of the staff will be displayed as shown in Figure 5.4.2.25. The staff can scroll down the page to view all the details of the profile. If the staff wants to edit their profile the staff can press on the profile picture, then the page will navigate to the edit profile details page as shown in Figure 5.4.2.26.



**Figure 5.4.2.25: Profile Details Interface**



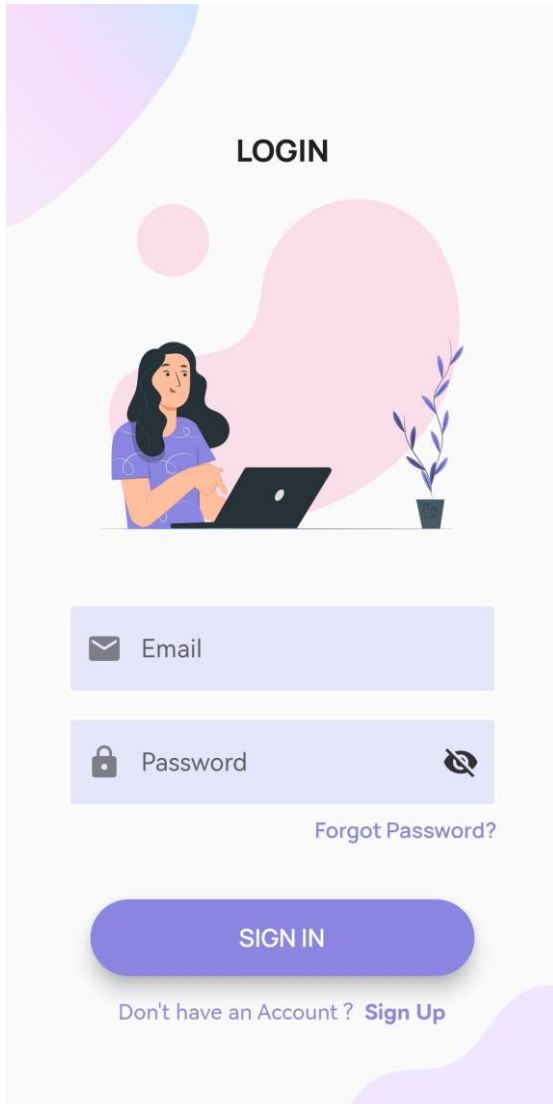
**Figure 5.4.2.26: Edit Profile Details Interface**

### 5.4.3 Patient Module

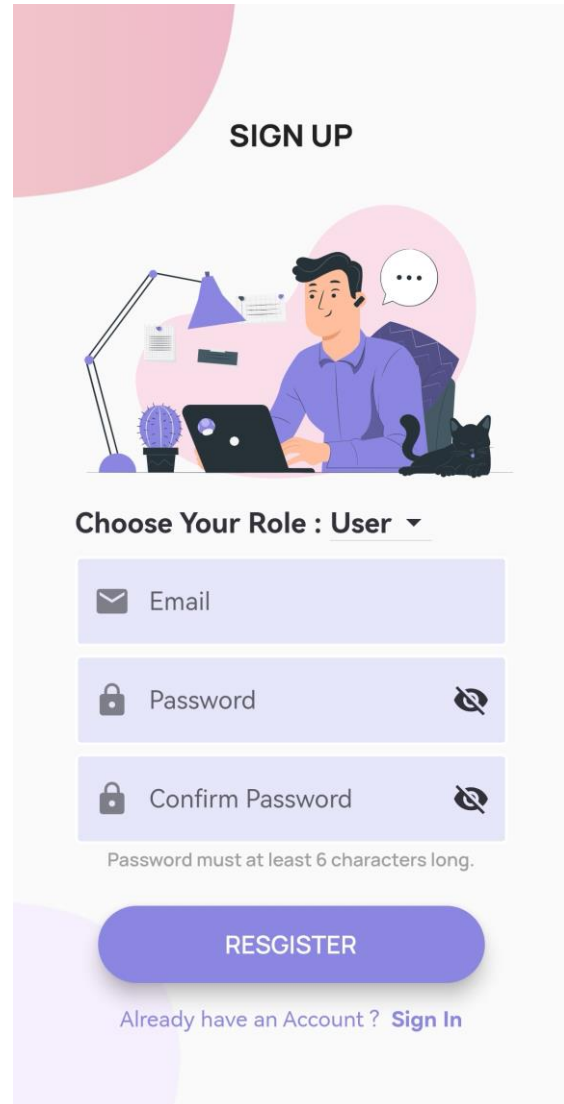
- **Account Registration**

When the patients launch the application, the patients will first interact with the Login interface as shown in Figure 5.4.3.1. The patients need to click on the “Sign Up” at the bottom of the “Sign In” button of the mobile application if they want to register for an account. When the patients entered the Sign-Up interface, the patients need to select their role to register to the system. For the patients they need to select their role as user as shown in Figure 5.4.3.2. Once the patients have entered the email address, password and confirm password, the patient is needed to press on the “Register” button to complete the registration. After pressing the button, the system will verify it and display a toast message to inform the patient whether the

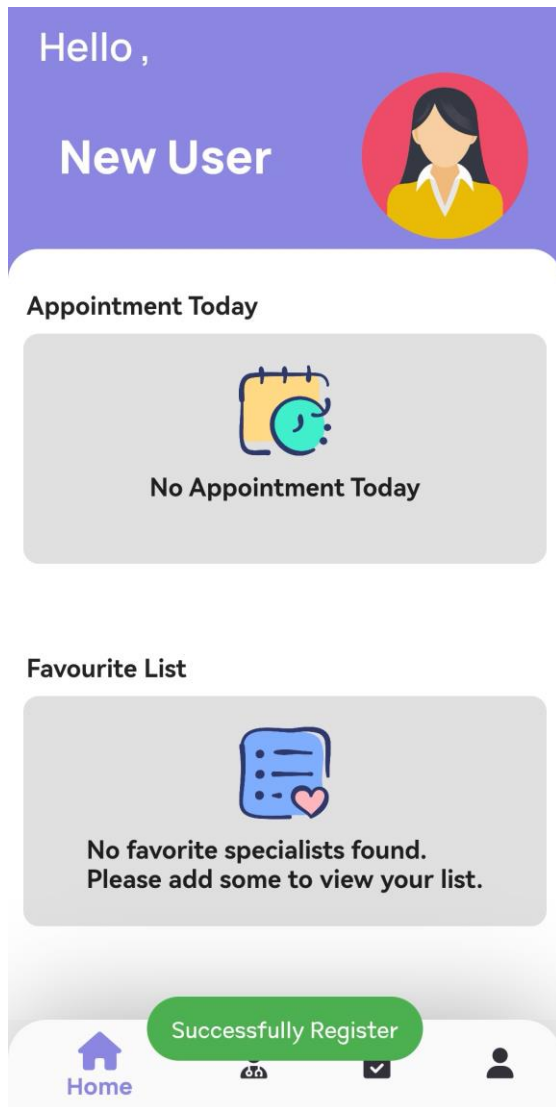
registration has been recorded as shown in Figure 5.4.3.3 and Figure 5.4.3.4. If the registration successful, then it will display “Successfully Register” and redirect the patient to the home interface of the system as shown in Figure 5.4.3.3 while unsuccessful of registration will be showing “Unsuccessful Registration” and an error message which requires the patient to entered again the details as shown in Figure 5.4.3.4.



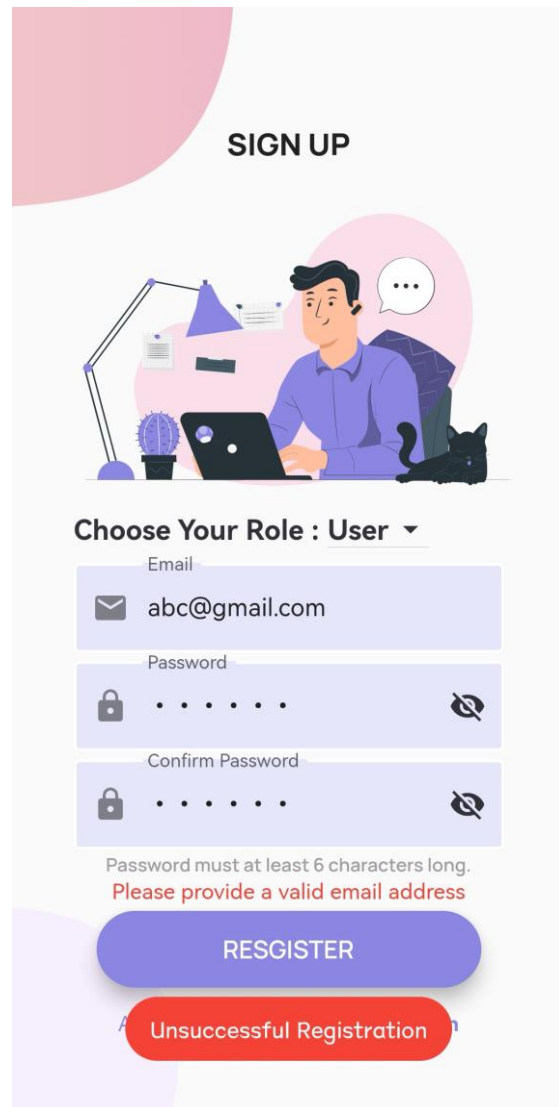
**Figure 5.4.3.1: Login Interface**



**Figure 5.4.2.2: Sign Up Form for Patient**



**Figure 5.4.2.3: Patient Account registration verification - Successful**

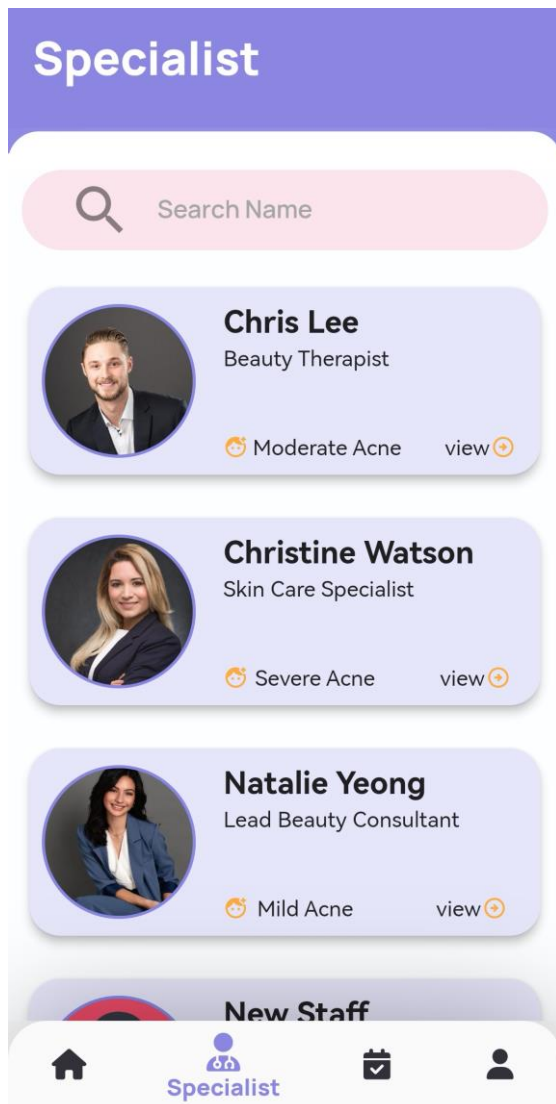


**Figure 5.4.2.4: Patient Account registration verification - Unsuccessful**

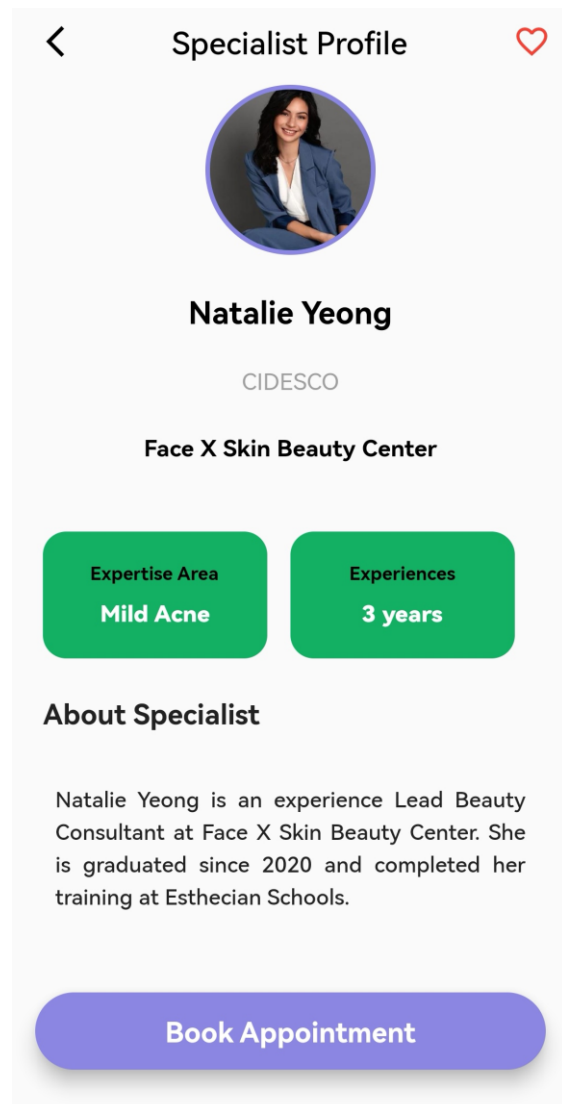
- **Specialist List**

When the patient wants to view the specialist’s list that exist in the database, the patient can press on the “ 👤 ” icon. Once the patient is pressing the button, the page will navigate to specialist list interface where the list of the specialist is displayed as shown in Figure 5.4.3.5. The patient can scroll through the list to view the specialist details. When the patient selects on a specialist to view the profile, the patient will be redirect to the specialist profile details page where the detail of the specialist is displayed as shown in Figure 5.4.3.6. The patient can also press on the “Book Appointment” button to book an appointment with the selected Specialist on booking appointment page as shown in Figure 5.4.3.7. Once the patient booked an appointment, the patient can press the “Make Appointment” button, then the page will navigate

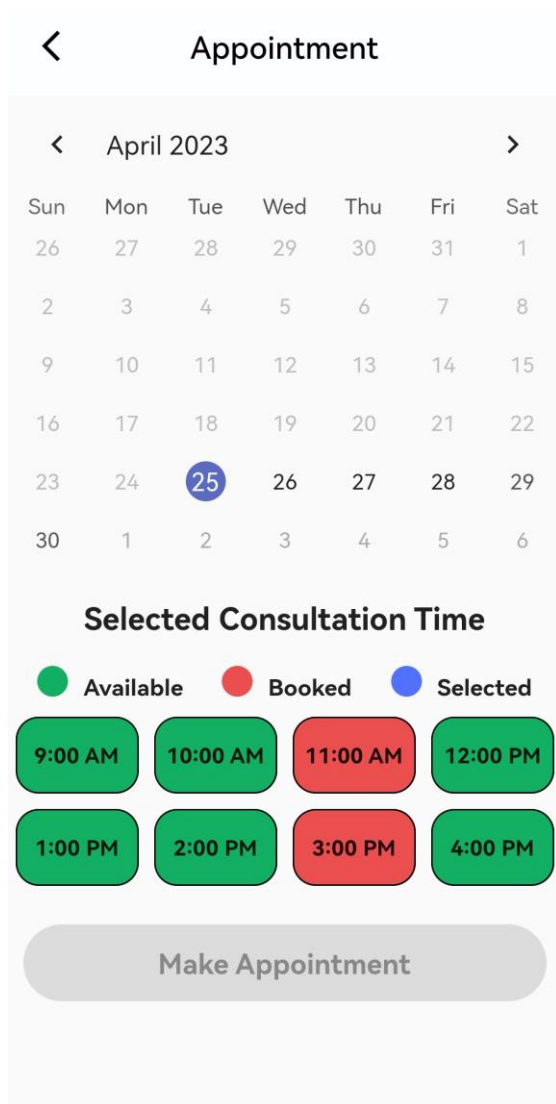
to the confirmation booking page as shown in Figure 5.4.3.8. The patient needs to press on the “Back to Home Page” button to exit the confirmation booking page.



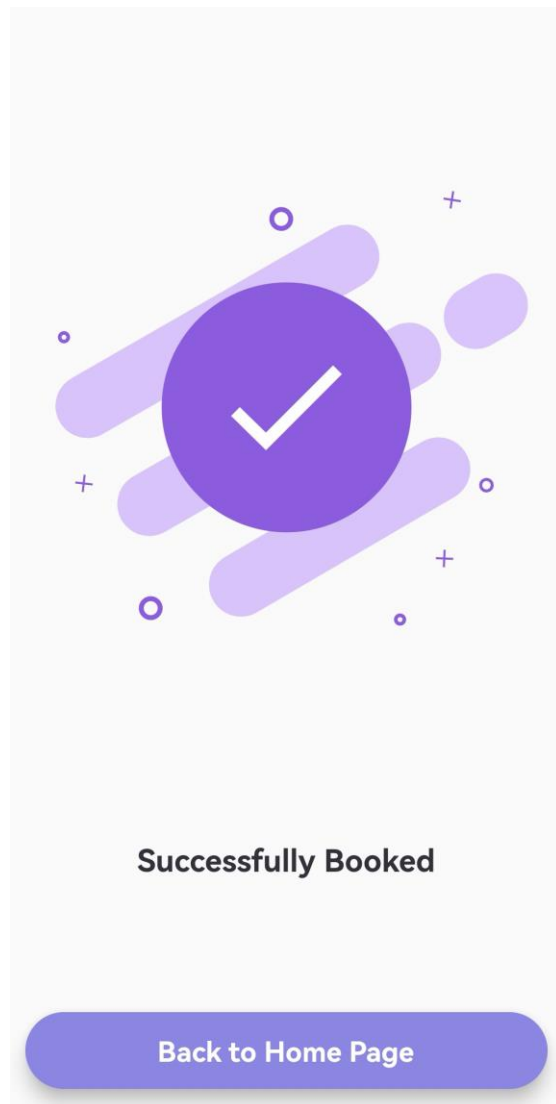
*Figure 5.4.3.5: Specialist's List View*



*Figure 5.4.3.6: Specialist Profile Details*




**Figure 5.4.3.7: Appointment Booking Interface**

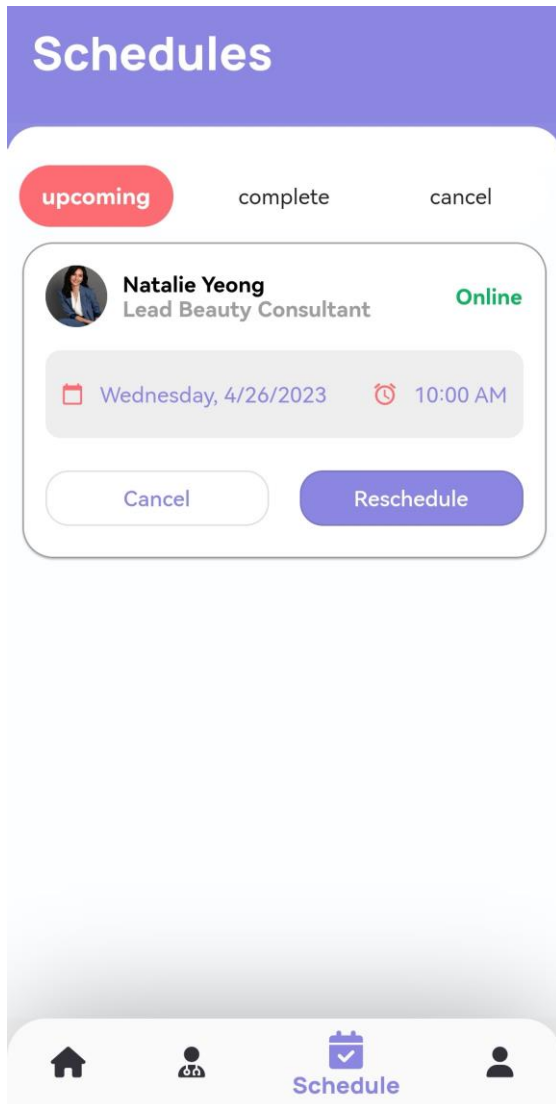


**Figure 5.4.3.8: Booking Confirmation Interface**

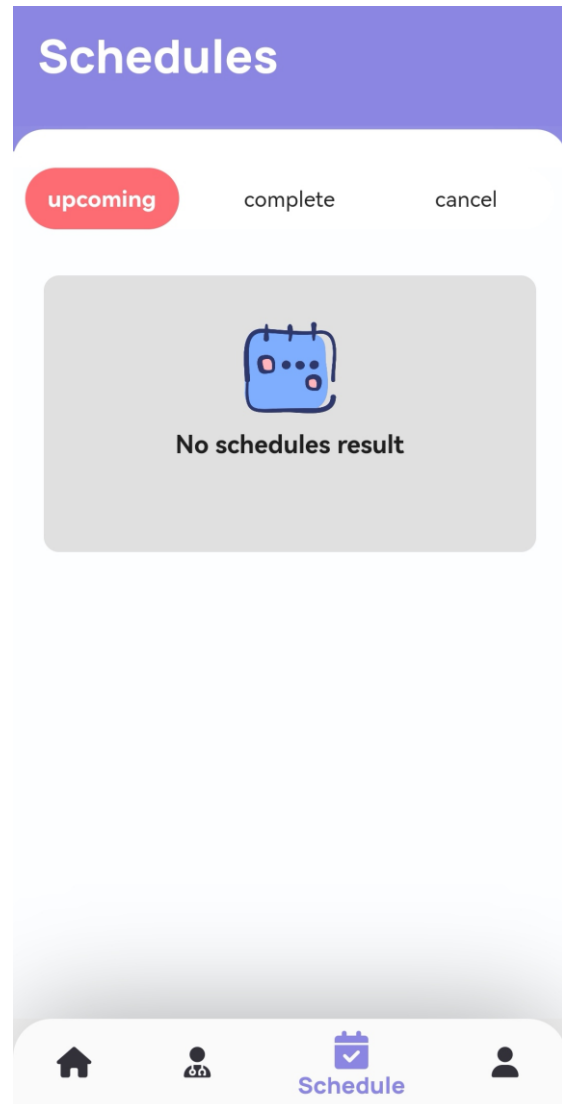
- **Schedules**

When the patient wants to view the schedule list that exist in the database, the patient can press on the “” icon. . Once the patient is pressing the button, the page will navigate to schedule list interface where the list of the appointment that have been booked is displayed when the list of schedules is not empty as shown in Figure 5.4.3.9. The patient will first see the upcoming schedule interface, and there are two other tab which is complete and cancel. When the schedule result is empty, a message will show “No schedules result” to indicates that no appointment has been booked as shown in Figure 5.4.3.10. The list of the schedule is different in each tab where the list will have “Cancel” and reschedule” button as shown in Figure 5.4.3.9, while for the list in complete tab, the list will have “View Consultation Details” button which

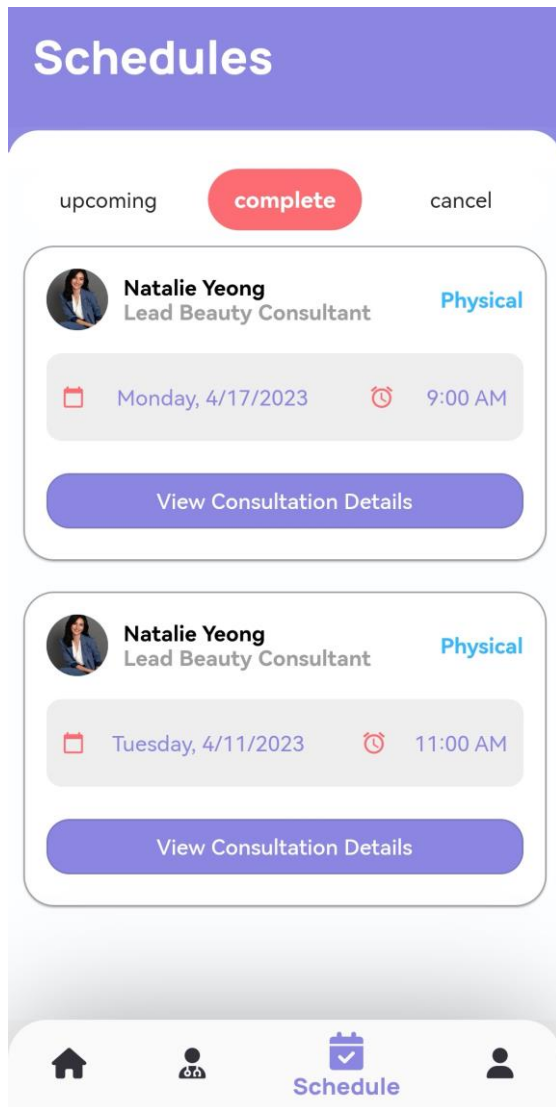
allow the patient to view the consultation history based on the selected appointment as shown in Figure 5.4.3.11. In the cancel tab, the patient can only reschedule the canceled appointment as shown in Figure 5.4.3.12. When the patient press on the “Reschedule” in upcoming page or cancel page, the patient will redirect to the reschedule booking page as shown in Figure 5.4.3.13. If the patient clicks on the “View Consultation Details” in complete tab, the patient will be redirect to the consultation details for the selected appointment as shown in Figure 5.4.3.14.



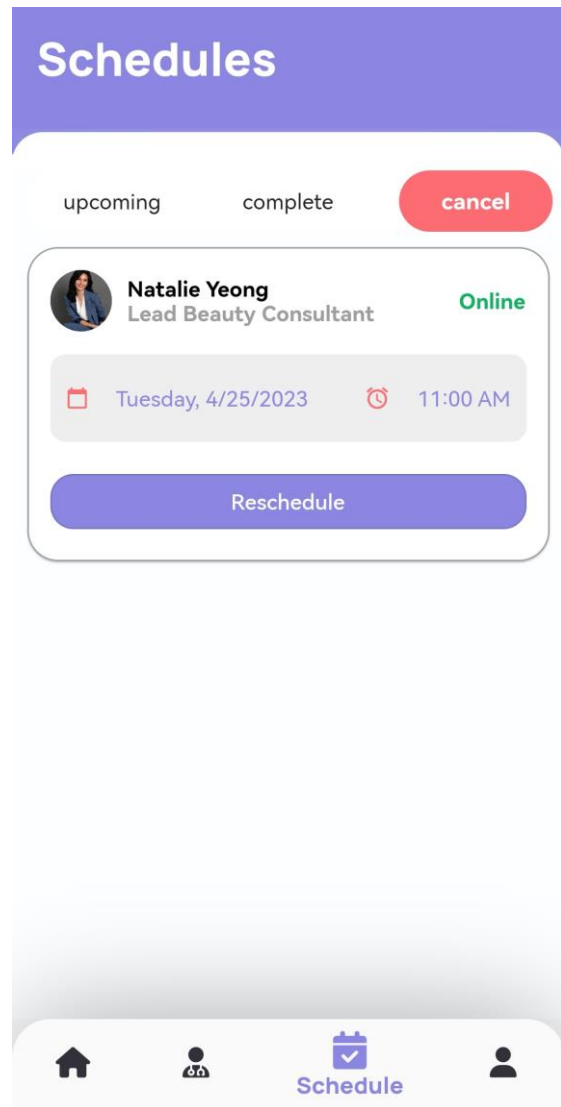
**Figure 5.4.3.9: Upcoming Interface- Patient**



**Figure 5.4.3.10: List of Schedule is empty**

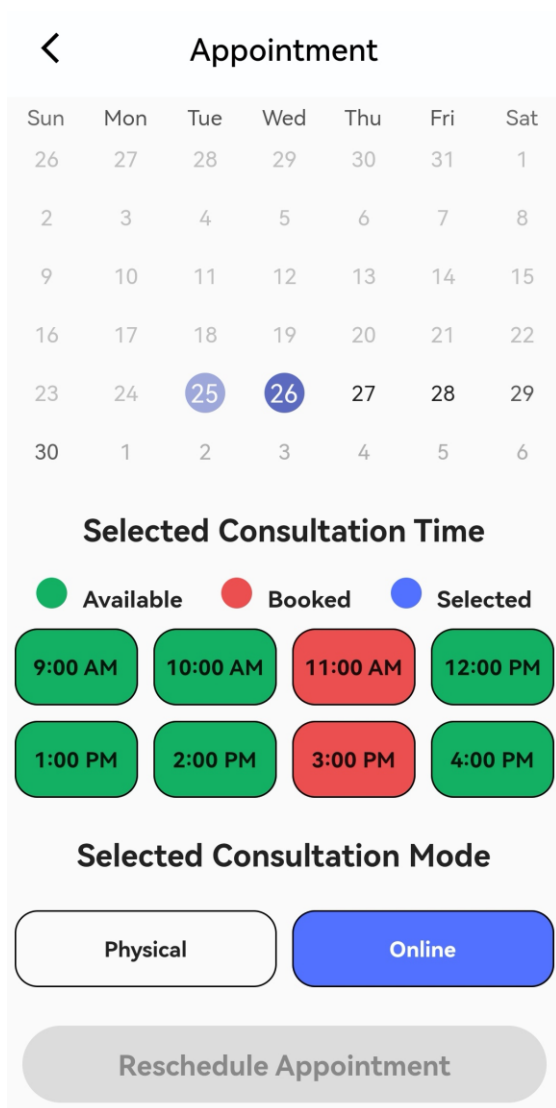


**Figure 5.4.3.11: Complete Interface - Patient**

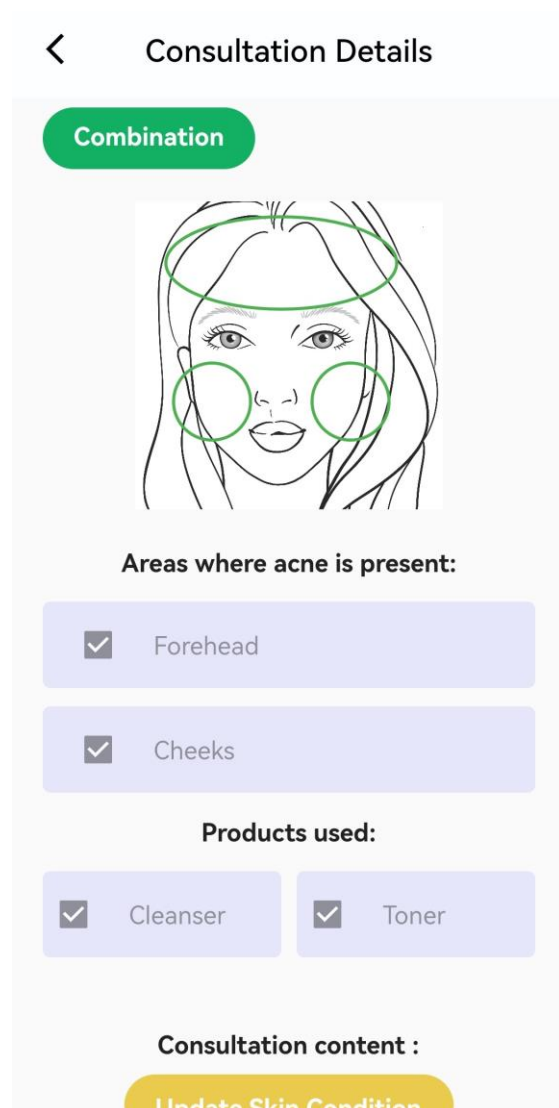


**Figure 5.4.3.12: Cancel Interface - Patient**





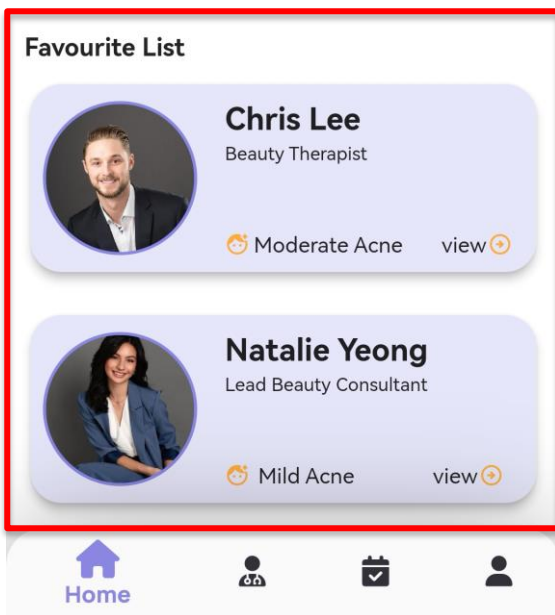
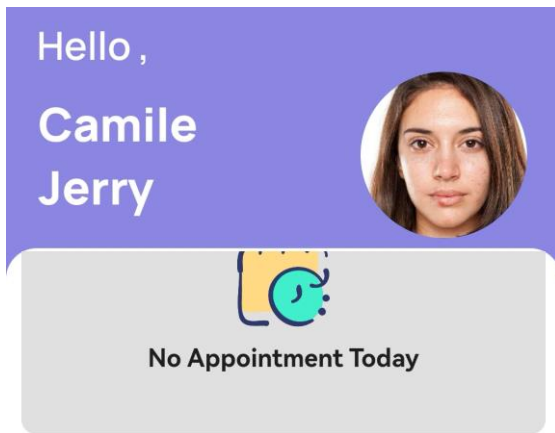
**Figure 5.4.3.13: Reschedule Appointment Interface- Patient**



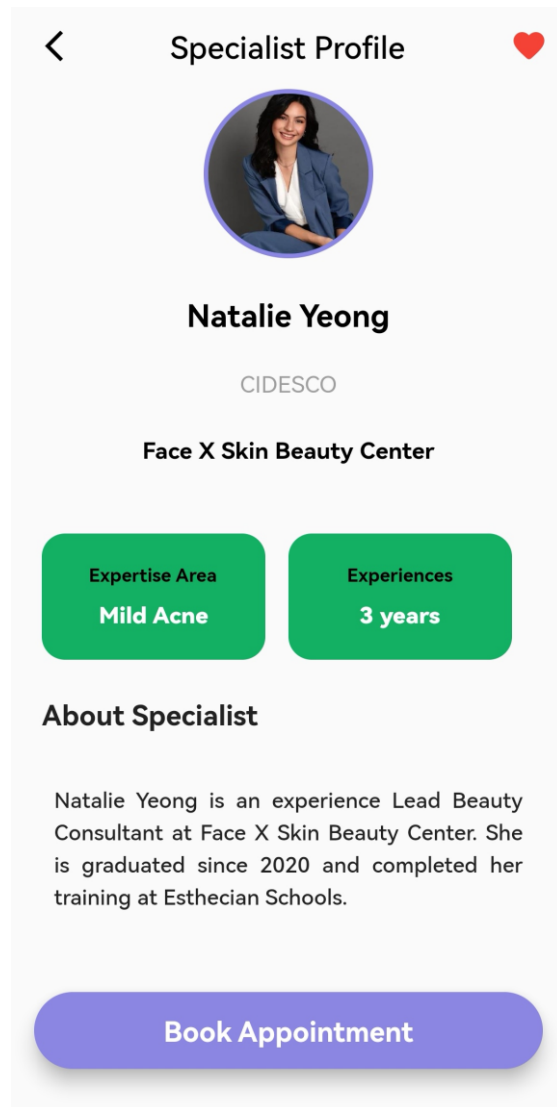
**Figure 5.4.3.14: Consultation Details Interface- Patient**

- **Favorite List**

The patient can view the favorite list in the homepage which is below the “Appointment Today” section as shown in Figure 5.4.3.15. The patient can scroll the favorite list if there are more than two favorite profiles. The patient can click on the profile card, then the page will navigate to the specialist profile details page. In the specialist profile details page, the heart icon on the top right is filled with red color to indicate that the profile is in the favorite list as shown in Figure 5.4.3.16.




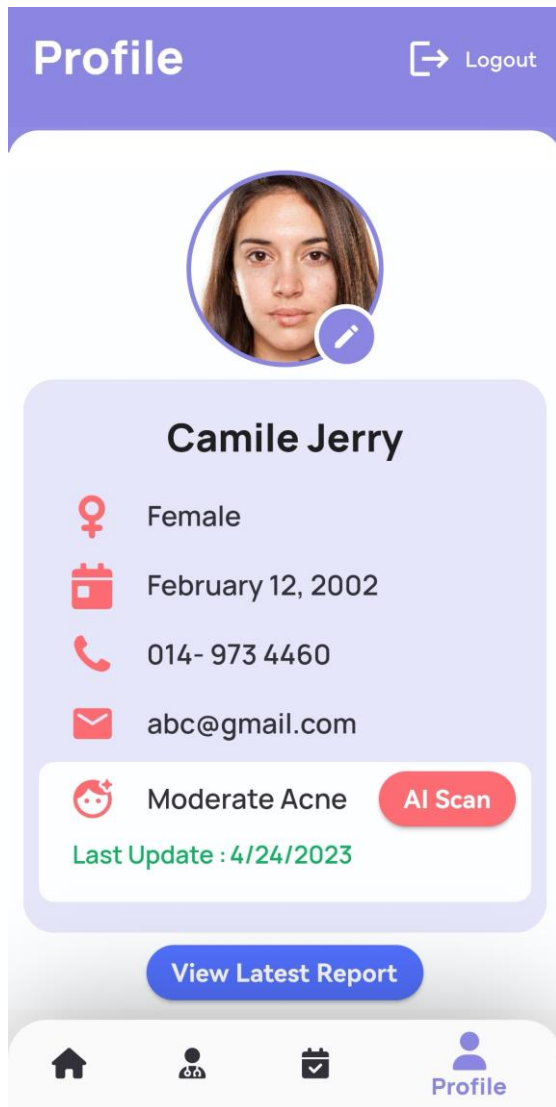
*Figure 5.4.3.15: Favorite List Section in Homepage - Patient*



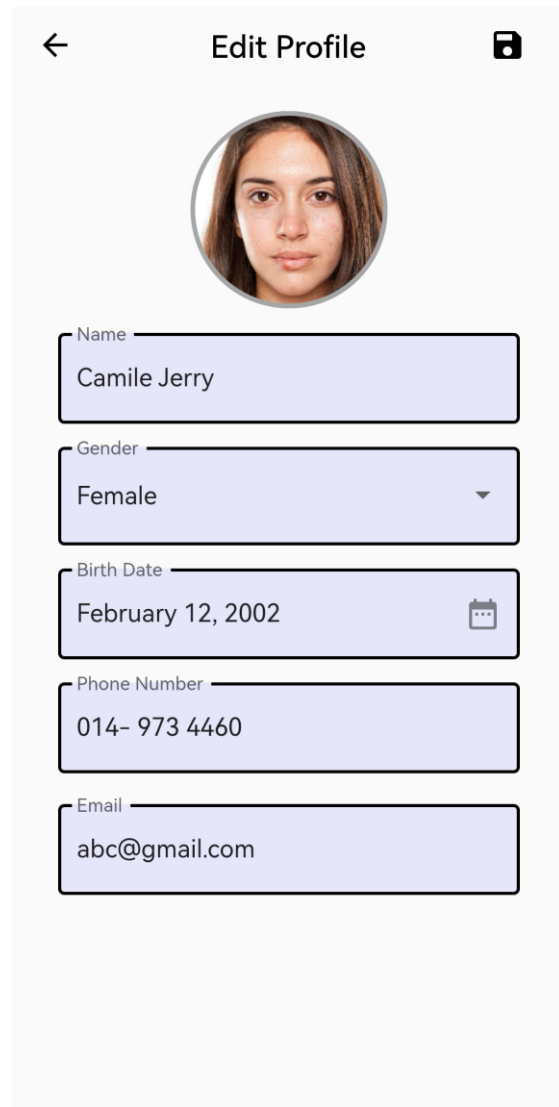
*Figure 5.4.3.16: Specialist Profile Details for Favorite List*

- **Profile**

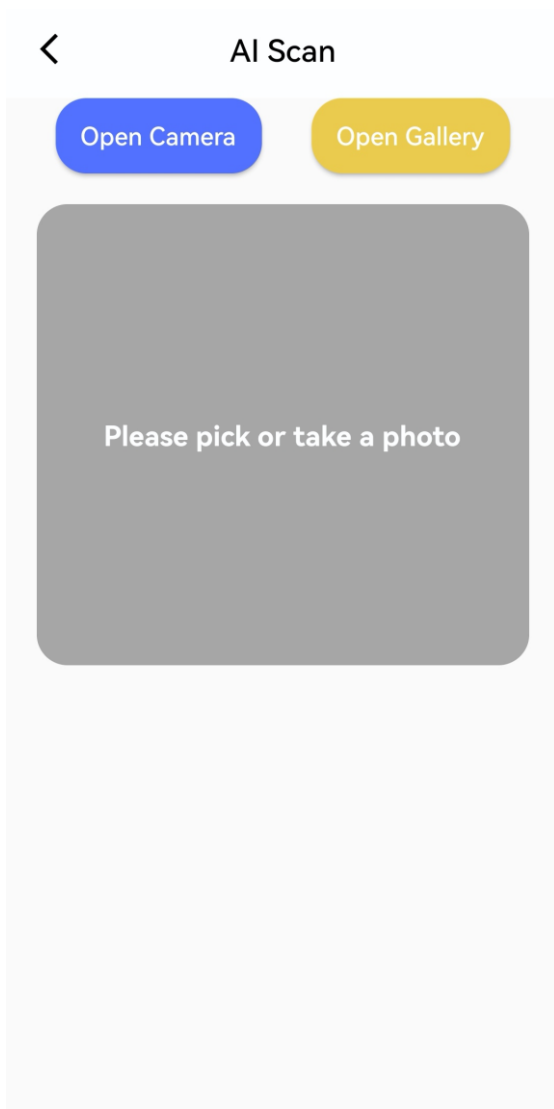
When the patient wants to view their profile details, the staff can press on the “  ” icon. . Once the patient is pressing the button, the page will navigate to profile details interface where the details of the patient will be displayed as shown in Figure 5.4.3.17. If the patients want to edit their profile the patients can press on the profile picture, then the page will navigate to the edit profile details page as shown in Figure 5.4.3.18. The patient can press on the “AI Scan” button, then the page will navigate to the classification page where the patient can update their skin category as shown in Figure 5.4.3.19. If the patient wants to view the latest consultation report, the patient can click on the “View Latest Report” and the patient will be redirect to the latest consultation details as shown in Figure 5.4.3.20.



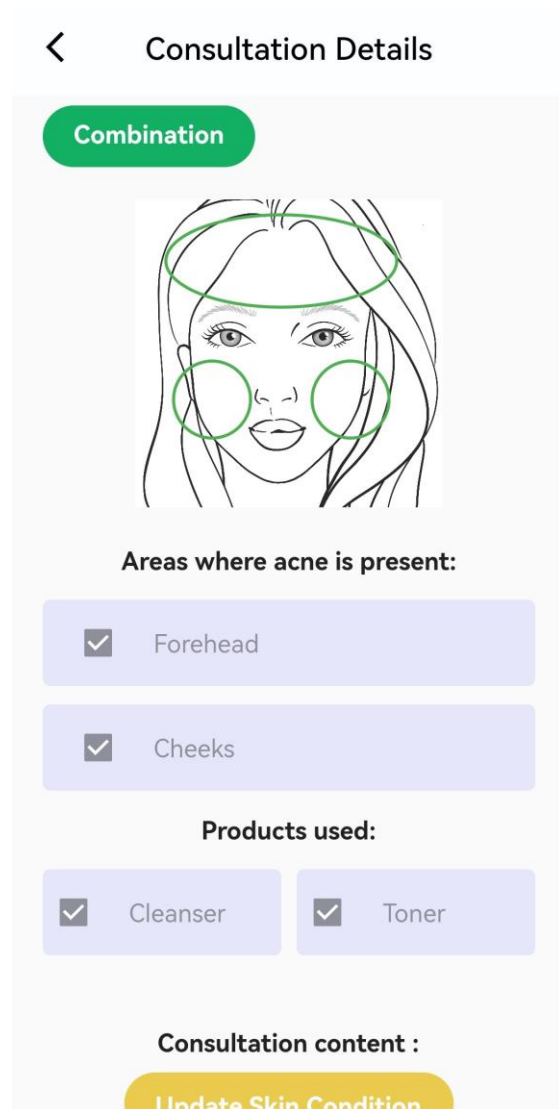
**Figure 5.4.3.17: Profile Details Interface - Patient**



**Figure 5.4.3.18: Edit Profile Details Interface - Patient**



**Figure 5.4.3.19: Acne Classification Interface - Patient**



**Figure 5.4.3.20: Consultation Details for last appointment - Patient**

## **5.5 Implementation Issues and Challenges**

The issues and challenges are inevitable throughout this project. Firstly, the installation and setup of the TensorFlow environment is challenging because it required to use the pip run to install the TensorFlow. Other than that, integration of TensorFlow into the Flutter is also challenging as the dependencies is having the incompatible issue and it still remain unresolved because it required the author to migrate the plugin.

Another issues that I have face during the progress of the project is that the integration of the plugin for ZEGOCLOUD because the version has been updated and there is lesser resource to be refer which make it harder for me during the implementation process of the video call. Therefore, the video call is functionable, but the calling notification is lacking due to the lack of documentation guide and the resources online.

Furthermore, there are some issues that have been faced during the building of the APK file. The use of ZEGOCLOUD and Firebase Cloud Messaging is not functionable when the application is run using APK file. Therefore, an extra file which is proguard-fules.pro to specify the task that allow the ZEGOCLOUD and Firebase Cloud Messaging to be able to function when using the APK to launch the system.

In this project, I have done my self-learning process as the TensorFlow is a new machine learning platform that I have not learn. Hence, I am required to pick up the progress to learn the basic function and understand the TensorFlow technique and some machine learning technique. Fortunately, there are many resources that can be refer and some full course tutorial that teach the theory and the basic flow of the image processing using TensorFlow. Even though the material that I have learn are the basic criteria, but I manage to deploy the TensorFlow Lite model into the Flutter mobile application.

## **5.6 Concluding Remark**

The system is completed with the core features that allow the users to interact with the system easily. The feature is link together and perform well which enable the users to navigate from different interface in the system to perform different task.

# Chapter 6

## System Evaluation and Discussion

### 6.1 TensorFlow Model Performance Evaluation

#### 6.1.1 Accuracy and Loss

The `itertools` module's `product()` function to generate combinations of hyperparameters such as `learning_rates`, `dropout_rates`, `initial_epochs`, and `fine_tune_epochs`. For each combination of hyperparameters, the script loads a pre-trained EfficientNetV2B0 model and appends a customized output layer. The compiled model uses the specified learning rate, loss function, and evaluation metric. The model is then trained on the training dataset for the prescribed initial epoch count and validated on the validation dataset. The base model is made trainable after validation, and the entire model is recompiled with a lower learning rate. The fine-tune epochs are also specified, and the model is fine-tuned on the training dataset with the weights learned during the initial training. List of hyperparameters use is shown in Figure 6.1.1.1.

```
learning_rates = [0.001, 0.0001]
dropout_rates = [0.2, 0.3]
initial_epochs = [30, 50]
fine_tune_epochs = [5, 10]
```

Figure 6.1.1.1: Snippet code for lists of hyperparameters

Finally, the model is evaluated on the validation dataset, and the accuracy along with the hyperparameter combination is printed. The result obtained is shown in Table 6.1.

Learning Rate	Dropout Rate	Initial epochs	Fine-tune epochs	Validation accuracy
0.001	0.2	30	5	0.37
0.001	0.2	30	10	0.39
0.001	0.2	50	5	0.37
0.001	0.2	50	10	0.39
0.001	0.3	30	5	0.39
0.001	0.3	30	10	0.39
0.001	0.3	50	5	0.39
0.001	0.3	50	10	0.39
0.0001	0.2	30	5	0.66

0.0001	0.2	30	10	0.63
0.0001	0.2	50	5	0.62
0.0001	0.2	50	10	0.53
0.0001	0.3	30	5	0.62
0.0001	0.3	30	10	0.66
0.0001	0.3	50	5	0.65
0.0001	0.3	50	10	0.63

**Table 6.1: Accuracy and Hyperparameters**

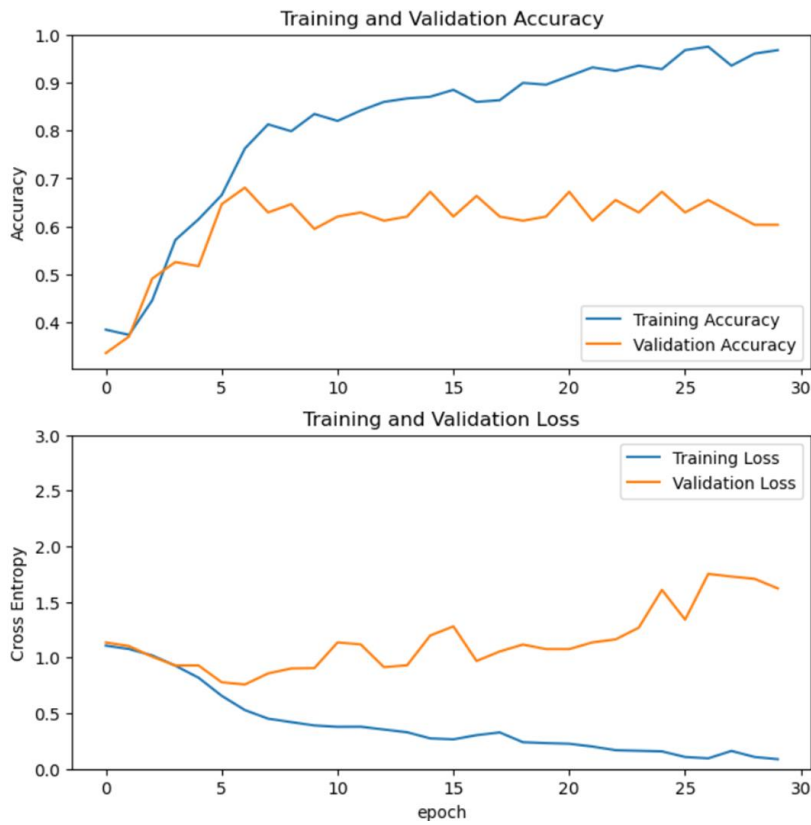
Based on the result obtained in Table 6.1, the dataset works well when the learning rate is 0.0001 and the dropout rate at 0.2 if the initial epochs are 30 and the fine-tune epochs is 5. However, the dataset also performs well when the dropout rate is 0.3 and the learning rate at 0.0001 when the initial epochs are 30 and the fine-tune-epochs is 10. Both hyperparameters obtained the accuracy at 0.66.

<b>Hyperparameter 1</b>	Learning Rate = 0.0001 Dropout Rate = 0.2 Initial epochs = 30 Fine-tune epochs = 5
<b>Hyperparameter 2</b>	Learning Rate = 0.0001 Dropout Rate = 0.3 Initial epochs = 30 Fine-tune epochs = 10

**Table 6.2: Hyperparameters with Highest Accuracy**



**Figure 6.1.1.2: Accuracy and Loss for Hyperparameter 1**



**Figure 6.1.1.3: Accuracy and Loss for Hyperparameter 2**

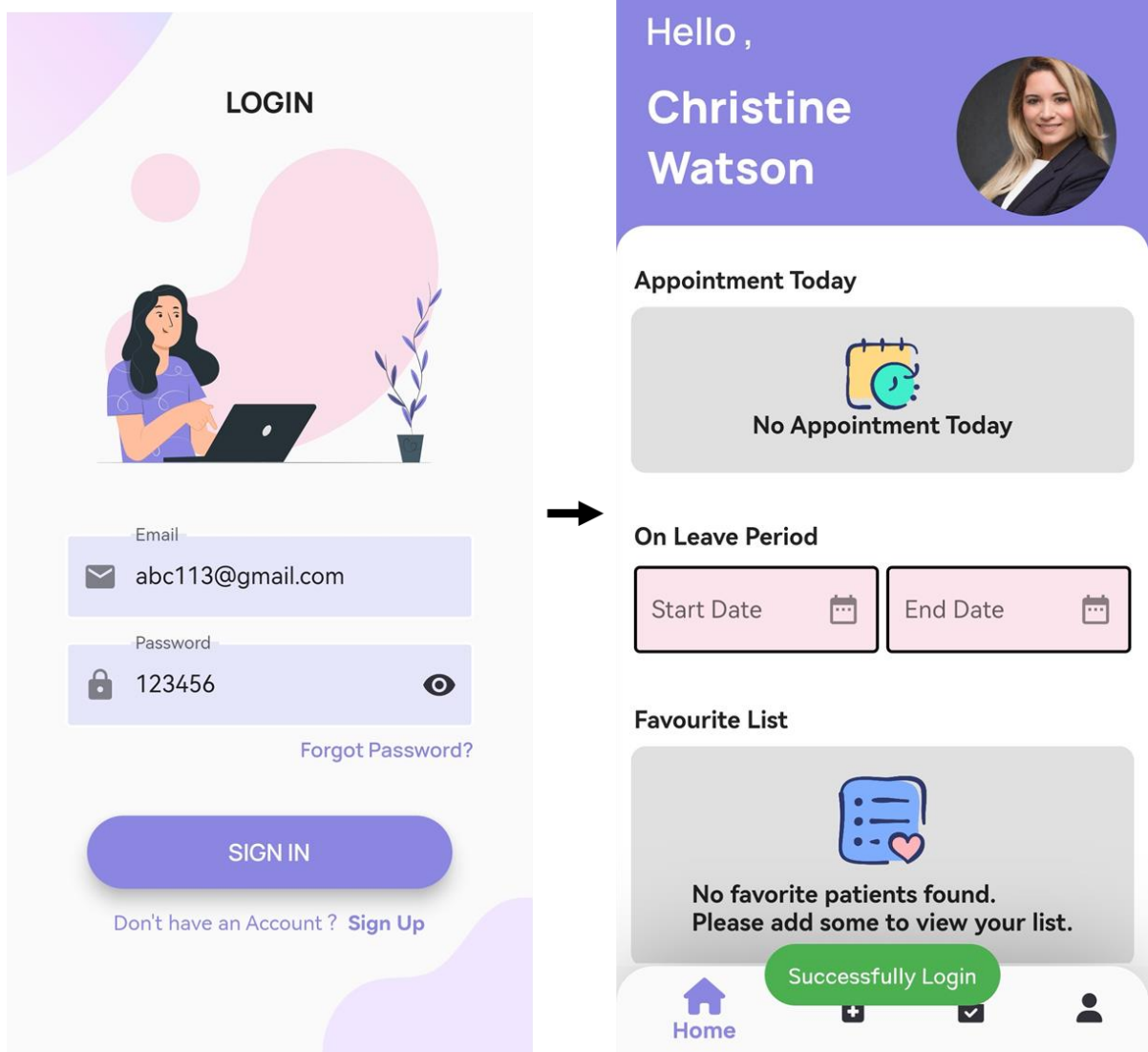


## 6.2 Testing Setup and Result

### 6.2.1 Login Module

- Login User Account

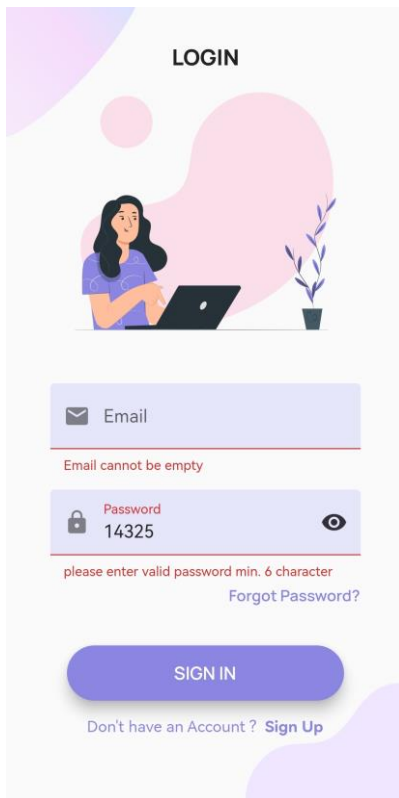
<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Login User Account	
<b>Reference Document</b>	authenticate.dart, background_signin.dart, sign_in.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_LUA_01	
<b>Test Scenario (Purpose)</b>	To verify the functionality of login a user account for staffs and patients.	
<b>Test Case Description</b>	Test the functionality of inserting login credentials for staffs and patient using an existing account.	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	1. Staff/ patients launch the application.	
<b>Expected Result</b>	Staff should be able to login an account and redirect to the homepage.	
<b>Actual Result</b>	Staff able to login existing user account successfully and has been redirect to the homepage with the existing user data displayed on the screen.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Input email address	abc113@gmail.com	
2. Input password	123456	
3. Click “Sign In” button once finish filled up all the login credentials	Sign In	
4. Redirect to the Homepage	Registration Successful	



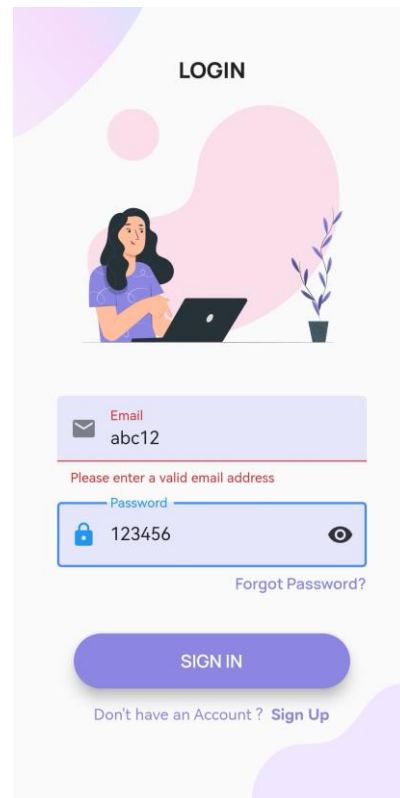
**Figure 6.2.1.1: Test Result for test case TC\_LUA\_01**

<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Login User Account	
<b>Reference Document</b>	authenticate.dart, background_signin.dart, sign_in.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_LUA_02	
<b>Test Scenario (Purpose)</b>	To verify the functionality of login a user account which does not allow the non-existing user to login .	
<b>Test Case Description</b>	<p>Test the functionality of login user account.</p> <ul style="list-style-type: none"> <li>- Unable to login the account with wrong login credentials and password.</li> <li>- Unable to login the account with wrong format of email address and password.</li> </ul>	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	1. Staff/ patients launch the application.	
<b>Expected Result</b>	Staff should be unable to login an account and error message will be displayed.	
<b>Actual Result</b>	Staff unable to login existing user account successfully and error message is displayed.	
<b>Test Result</b>	PASSED	
<b>Test Step (Invalid email and password)</b>	<b>Test Data</b>	
1. Input email address	(blank)	
2. Input password	14325	
3. Click “Sign In” button once finish filled up all the login credentials	Sign In	
4. Redirect to the Homepage	Unsuccessful Login	
<b>Test Step (Invalid email)</b>	<b>Test Data</b>	
1. Input email address	abc12	
2. Input password	123456	
3. Click “Sign In” button once finish filled up all the login credentials	Sign In	
4. Redirect to the Homepage	Unsuccessful Login	

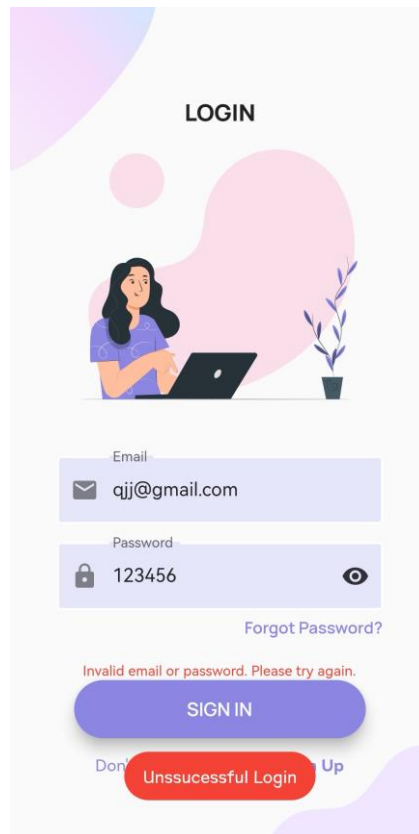
Test Step (Non-existing user account)	Test Data
1. Input email address	qjj@gmail.com
2. Input password	123456
3. Click “Sign In” button once finish filled up all the login credentials	Sign In
4. Redirect to the Homepage	Unsuccessful Login



**Figure 6.2.1.2: Test Result for invalid email and password (TC\_LUA\_02)**



**Figure 6.2.1.3: Test Result for invalid email (TC\_LUA\_02)**



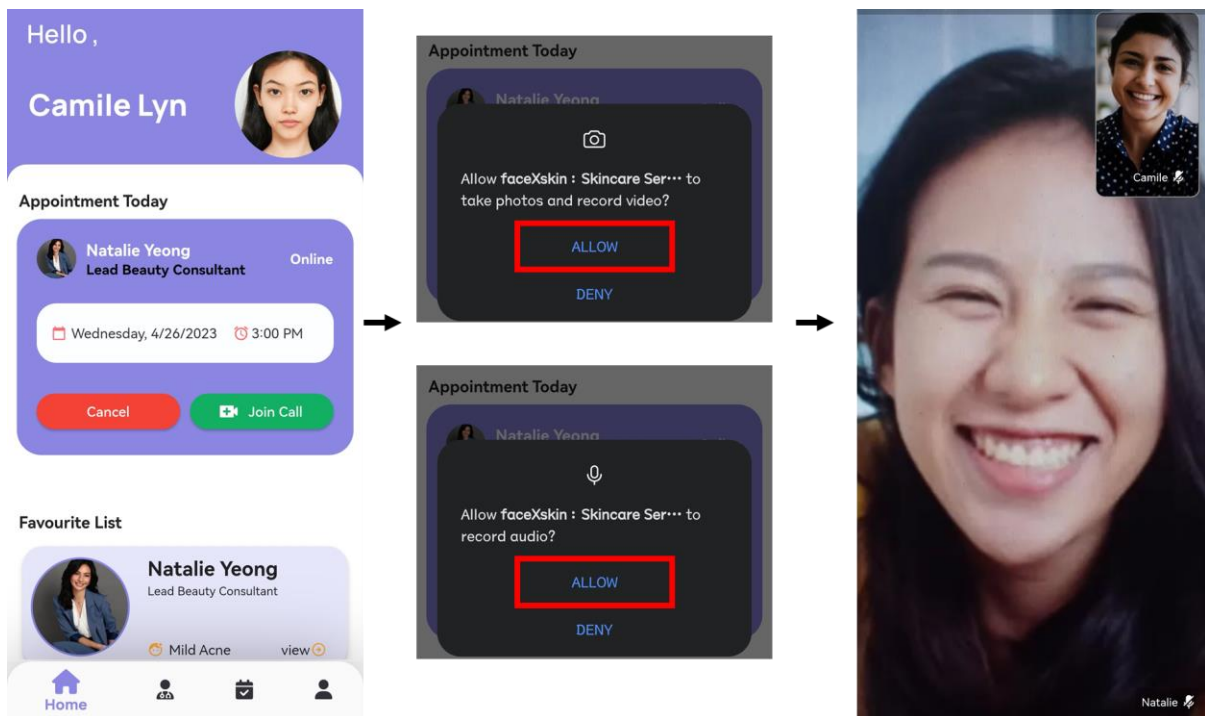
**Figure 6.2.1.4: Test Result for non-existing user account (TC\_LUA\_02)**

## 6.2.2 Staff & Patient Module

- Video Call

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Video Call
<b>Reference Document</b>	call_invitation.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_VC_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of video call which allow the staff and patient to join the call.
<b>Test Case Description</b>	Test the functionality of joining the video call. <ul style="list-style-type: none"> <li>- Able to join the video call room.</li> </ul>

<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff/ Patient is login to the system successfully.</li> <li>2. Staff/ Patient has appointment today and the appointment mode is online.</li> </ol>
<b>Expected Result</b>	Staff/ Patient should be able to join the video call consultation room with microphone and camera.
<b>Actual Result</b>	Staff? Patient able to join the video call room with microphone and camera.
<b>Test Result</b>	PASSED
<b>Test Step</b>	<b>Test Data</b>
1. Click “Join Call” button	Join Call
2. Permission Access Device for microphone and camera	Allow
3. Redirect to the video call room	Join Call Successful



**Figure 6.2.2.1: Test Result for test case TC\_VC\_01**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Video Call
<b>Reference Document</b>	call_invitation.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_VC_02
<b>Test Scenario (Purpose)</b>	To verify the functionality of video call which the staff and patient to join the call.
<b>Test Case Description</b>	Test the functionality of joining the video call. - Unable to join the video call room.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	1. Staff/ Patient is login to the system successfully. 2. Staff/ Patient has appointment today and the appointment mode is online.
<b>Expected Result</b>	Staff/ Patient unable to join the video call room and redirect to the homepage with error message displayed.
<b>Actual Result</b>	Staff/ Patient unable to join the video call room and redirect to the homepage with error message displayed.
<b>Test Result</b>	PASSED
<b>Test Step</b>	<b>Test Data</b>
1. Click "Join Call" button	Join Call
2. Allow application to microphone and camera	Deny
3. Redirect to the video call room	Join Call Unsuccessful

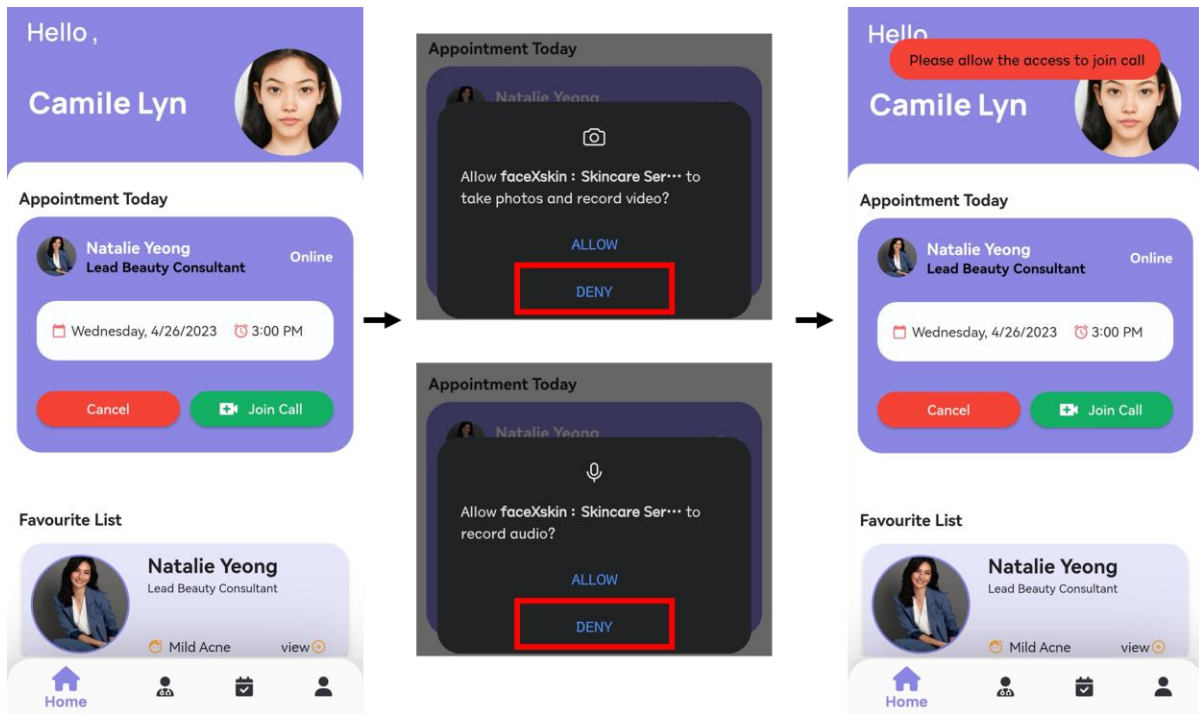


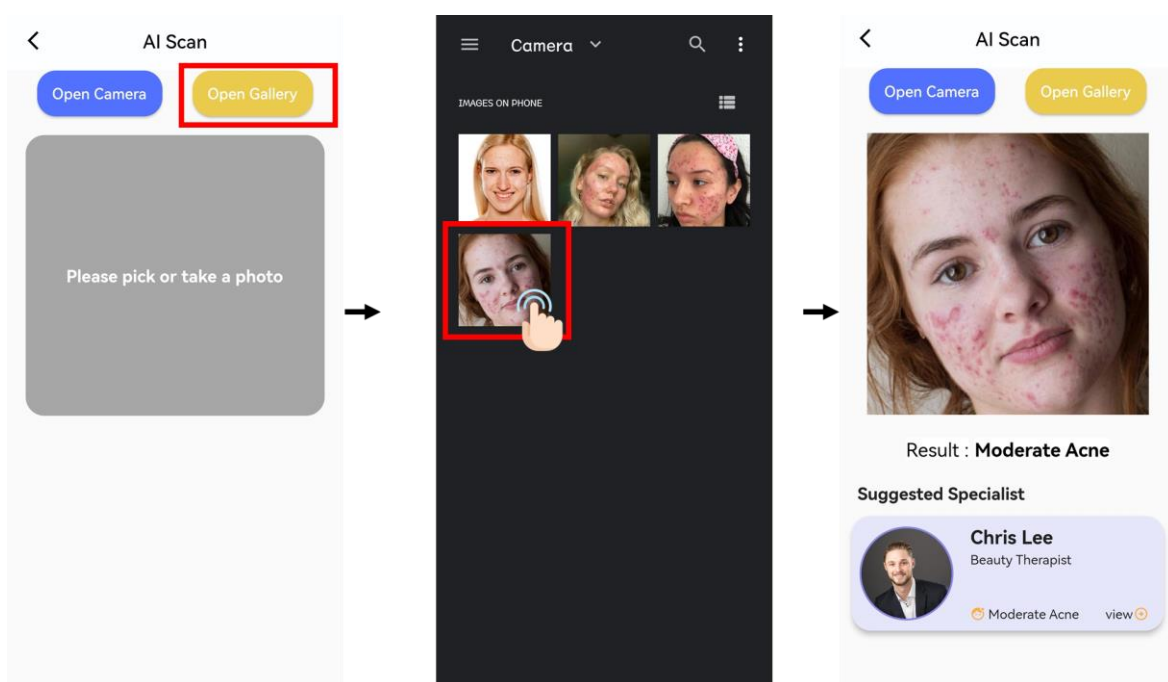
Figure 6.2.2.2: Test Result for test case TC\_VC\_02

- AI Scan ( Upload Image)

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	AI Scan
<b>Reference Document</b>	aiscan.dart, scan_detail.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_AIS_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of classification for acne by uploading the image.
<b>Test Case Description</b>	Test the functionality of uploading image to use the acne classification.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	1. Staff/ Patient click “AI Scan” button. 2. Staff/ Patient is on the acne classification page.
<b>Expected Result</b>	Staff/ Patient should be able to view the classification result after uploading the image.



<b>Actual Result</b>	Staff/ Patient able to upload the image and view the classification result after uploading the image.	
<b>Test Result</b>	PASSED	
<b>Test Step (Staff Module)</b>		<b>Test Data</b>
1. Staff click “Open Gallery” to upload the image from the gallery		“Open Gallery”
2. Staff select image to upload		image selected
3. Staff view the classification result		-
<b>Test Step (Patient Module)</b>		<b>Test Data</b>
1. Staff click “Open Gallery” to upload the image from the gallery		“Open Gallery”
2. Staff select image to upload		image selected
3. Staff view the classification result and suggested specialist		-



**Figure 6.2.2.3: Test Result for Staff Module (TC\_AIS\_01)**

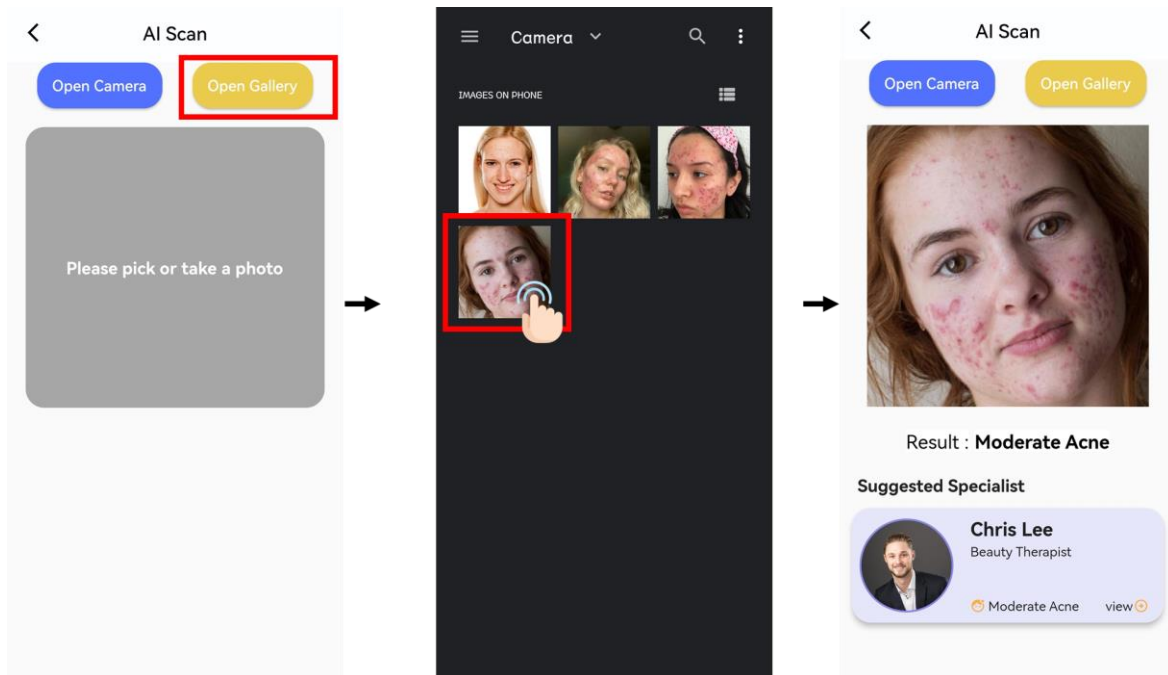
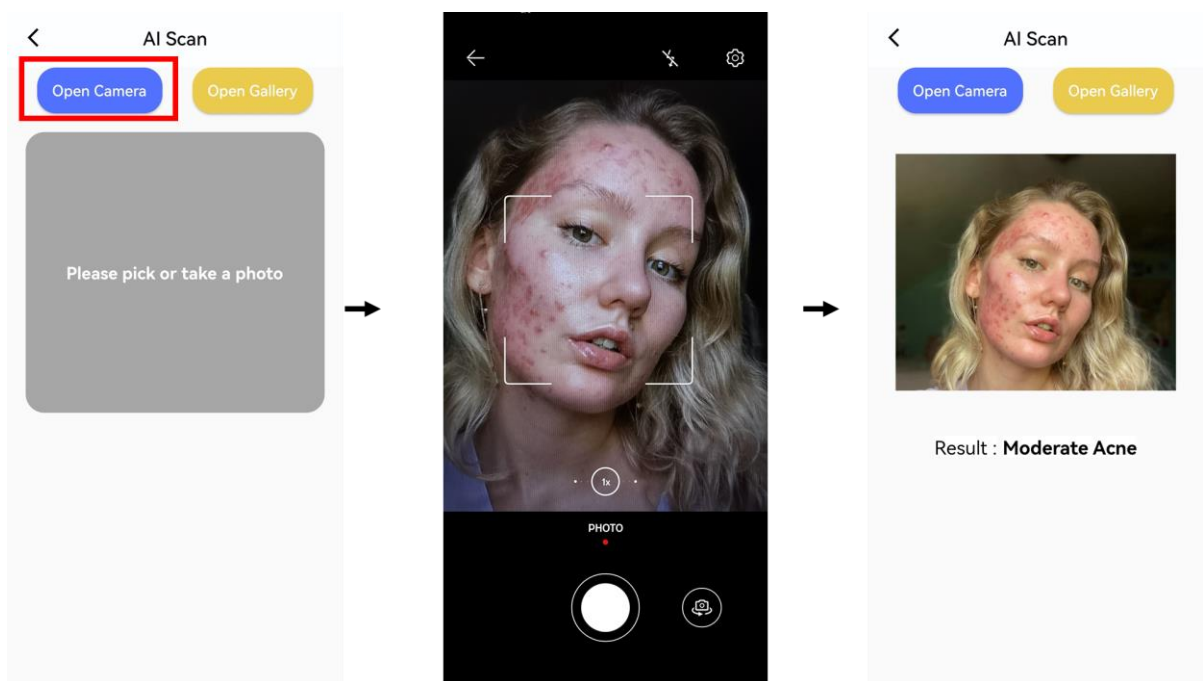


Figure 6.2.2.4: Test Result for Patient Module (TC\_AIS\_01)

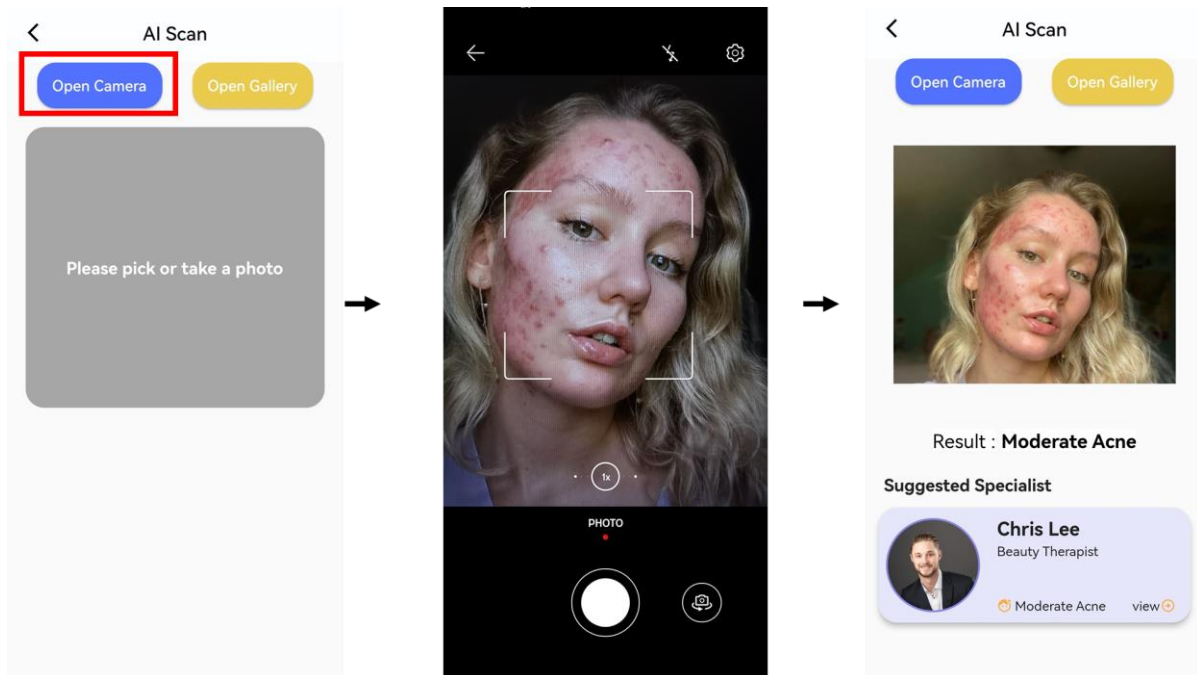
- AI Scan ( Capture Image)

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	AI Scan
<b>Reference Document</b>	aiscan.dart, scan_detail.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_AIS_02
<b>Test Scenario (Purpose)</b>	To verify the functionality of classification for acne by capture an image.
<b>Test Case Description</b>	Test the functionality of capturing an image to use the acne classification.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff/ Patient click “AI Scan” button.</li> <li>2. Staff/ Patient is on the acne classification page.</li> </ol>
<b>Expected Result</b>	Staff/ Patient should be able to view the classification result after capturing the image.

<b>Actual Result</b>	Staff/ Patient able to capture the image and view the classification result after capturing the image.	
<b>Test Result</b>	PASSED	
<b>Test Step (Staff Module)</b>		<b>Test Data</b>
1. Staff click “Open Camera” to upload the image from the gallery		“Open Camera”
2. Staff capture image using camera		image captured
3. Staff view the classification result		-
<b>Test Step (Patient Module)</b>		<b>Test Data</b>
1. Patient clicks “Open Camera” to upload the image from the gallery		“Open Camera”
2. Patient captures image using camera		image captured
3. Patients view the classification result and suggested specialist		-



*Figure 6.2.2.5: Test Result for Staff Module (TC\_AIS\_02)*

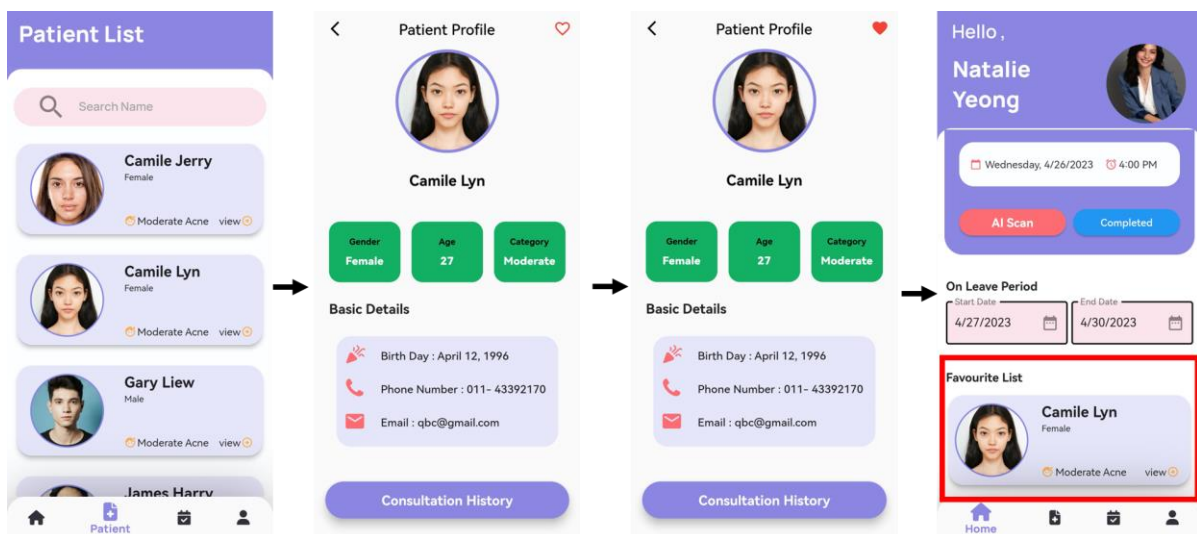


**Figure 6.2.2.6: Test Result for Patient Module (TC\_AIS\_02)**

- **Add Favourite List**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Add Favourite List
<b>Reference Document</b>	fav_card.dart, favlist.dart, body.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_AFL_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of favourite list which the profile is added to the favourite list
<b>Test Case Description</b>	Test the functionality of adding the profile to the favourite list.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff/ Patient login successfully login to the system.</li> <li>2. Staff/ Patient is on the homepage.</li> </ol>
<b>Expected Result</b>	Staff/ Patient should be able to add the profile to the favourite list and view the profile on the homepage.

<b>Actual Result</b>	Staff/ Patient is able to add the profile to the favourite list and view the profile on the homepage.	
<b>Test Result</b>	PASSED	
<b>Test Step (Staff Module)</b>		<b>Test Data</b>
1. Staff click “Patient” on the bottom navigation tab		“Patient”
2. Staff press on the selected patient profile		Patient profile selected
3. Staff is in the patient profile details page		-
4. Staff press on the heart icon to add the profile into favourite list		“Heart” icon
5. Staff exit the patient profile details page by pressing the back button		“ < ”
6. Staff return to the homepage by pressing “Home” on the bottom navigation tab		“Home”
7. Staff view the profile in the favourite list		-
<b>Test Step (Patient Module)</b>		<b>Test Data</b>
1. Patient clicks “Specialist” on the bottom navigation tab		“Specialist”
2. Patient presses on the selected specialist profile		Specialist profile selected
3. Patient is in the specialist profile details page		-
4. Patient presses on the heart icon to add the profile into favourite list		“Heart” icon
5. Patient exit the specialist profile details page by pressing the back button		“ < ”
6. Patient returns to the homepage by pressing “Home” on the bottom navigation tab		“Home”
7. Patient views the profile in the favourite list		-



**Figure 6.2.2.7: Test Result for Staff Module (TC\_AFL\_01)**

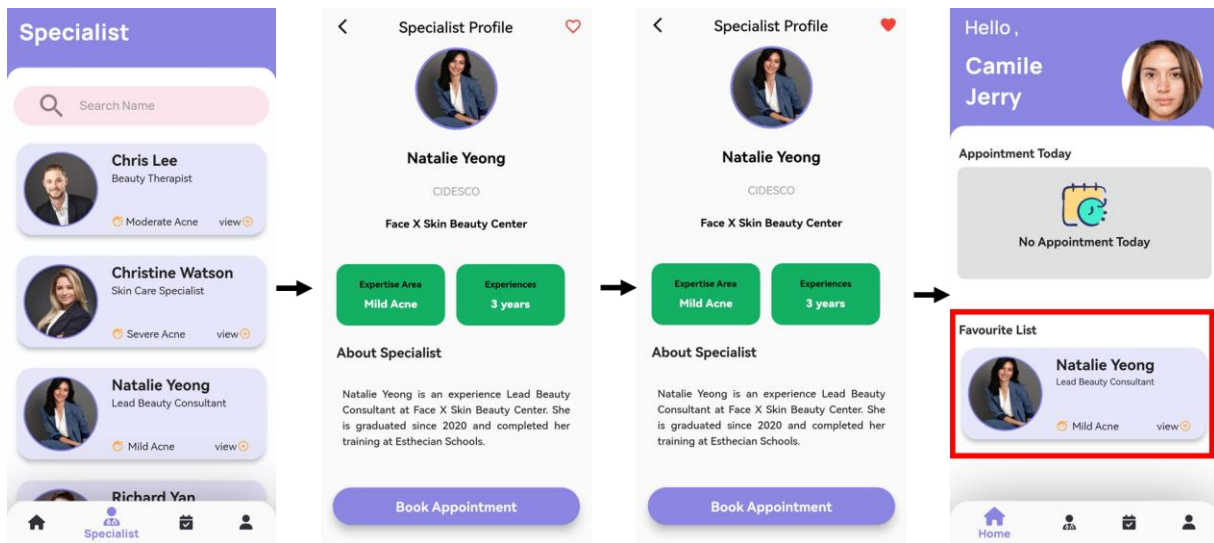
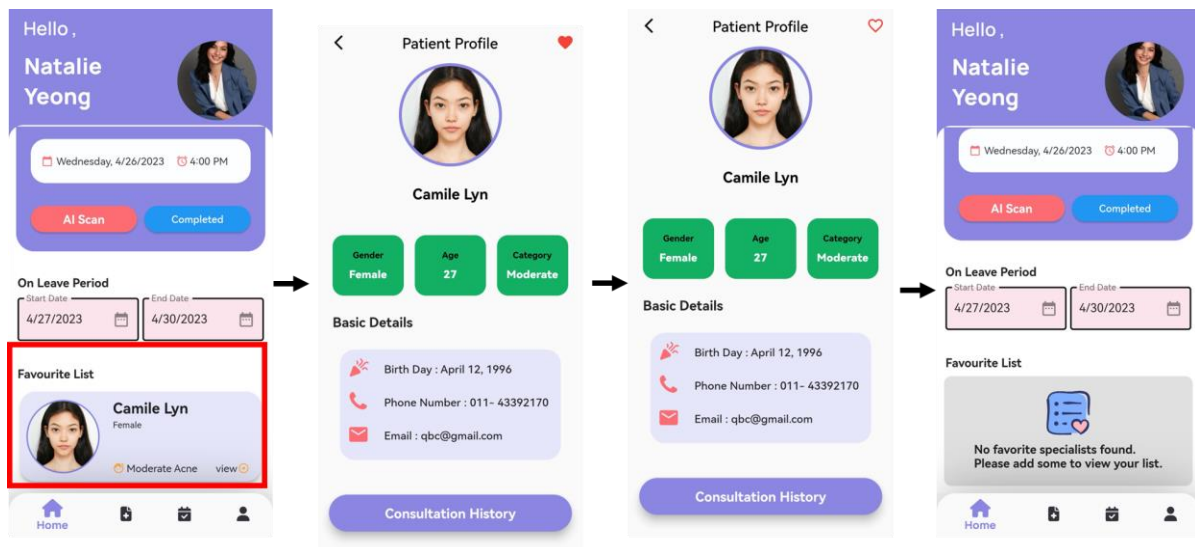


Figure 6.2.2.8: Test Result for Patient Module (TC\_AFL\_01)

- Remove Favourite List

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Remove Favourite List
<b>Reference Document</b>	fav_card.dart, favlist.dart, body.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_RFL_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of favourite list which the profile is removed from the favourite list
<b>Test Case Description</b>	Test the functionality of removing the profile to the favourite list.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>Staff/ Patient login successfully login to the system.</li> <li>Staff/ Patient is on the homepage.</li> </ol>
<b>Expected Result</b>	Staff/ Patient should be able to remove the profile to the favourite list and view the profile on the homepage.
<b>Actual Result</b>	Staff/ Patient is able to remove the profile to the favourite list and view the profile on the homepage.

<b>Test Result</b>	PASSED	
<b>Test Step (Staff Module)</b>		<b>Test Data</b>
1. Staff press on the selected patient profile		Patient profile selected
2. Staff is in the patient profile details page		-
3. Staff press on the heart icon to add the profile into favourite list		“Heart” icon
4. Staff exit the patient profile details page by pressing the back button		“ < ”
5. Profile removed from the favourite list		successfully removed
<b>Test Step (Patient Module)</b>		<b>Test Data</b>
1. Patient presses on the selected specialist profile		Specialist profile selected
2. Patient is in the specialist profile details page		-
3. Patient presses on the heart icon to add the profile into favourite list		“Heart” icon
4. Patient exit the specialist profile details page by pressing the back button		“ < ”
5. Profile removed from the favourite list		successfully removed



**Figure 6.2.2.9: Test Result for Patient Module (TC\_RFL\_01)**

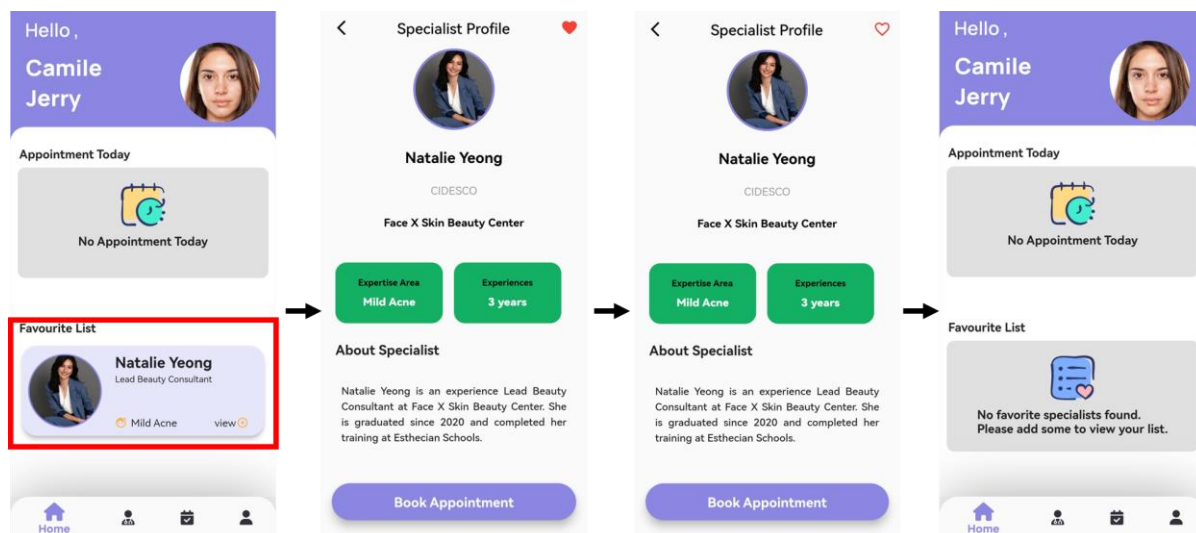


Figure 6.2.2.10: Test Result for Patient Module (TC\_RFL\_01)

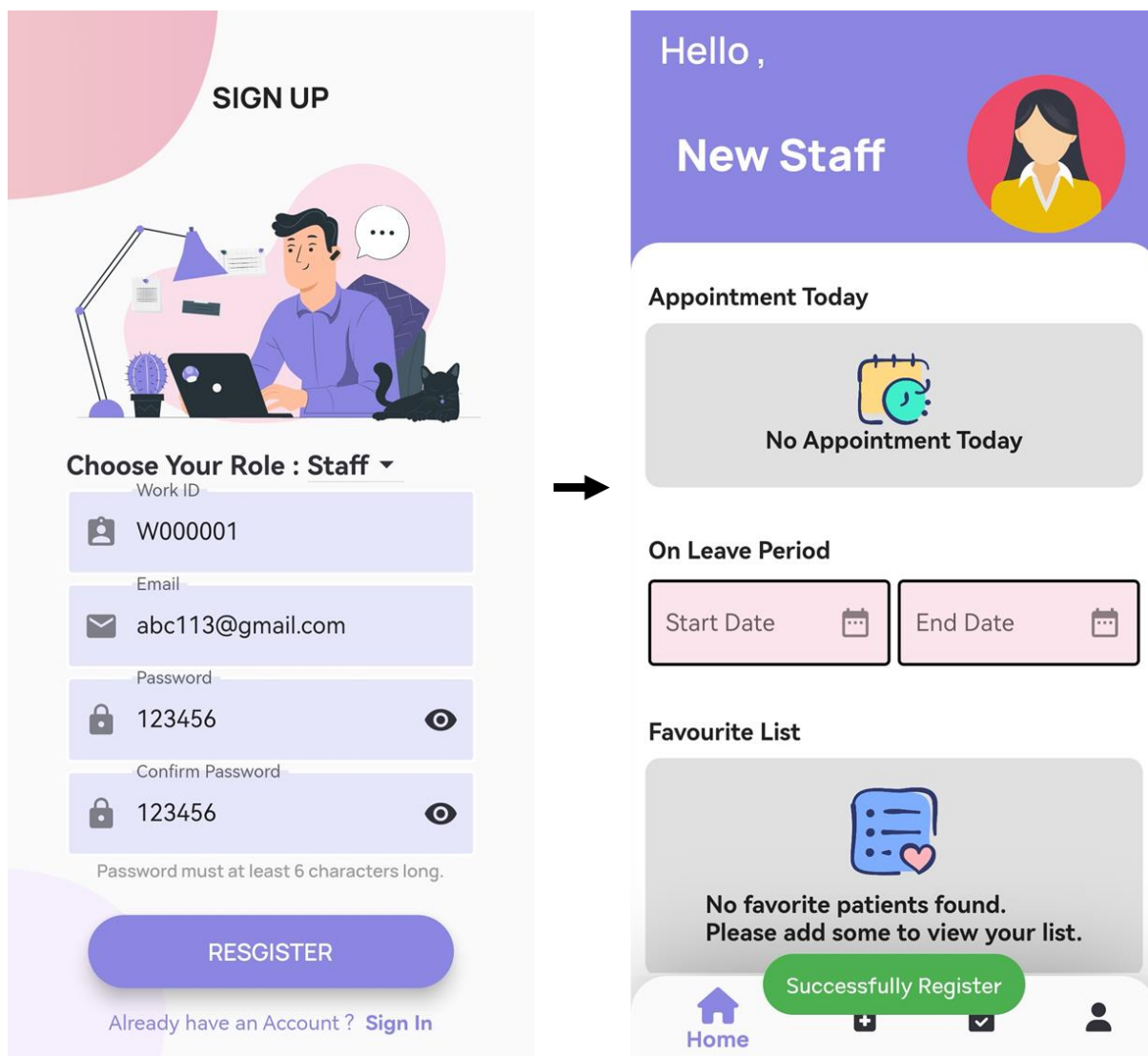
### 6.2.3 Staff Module

- Register User Account

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Staff Module – Register User Account
<b>Reference Document</b>	authenticate.dart, background_signup.dart, sign_up.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_RUA_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of register a user account which successfully register an account to the system.
<b>Test Case Description</b>	Test the functionality of inserting new staff details such as entering work id, email address, password and confirm password and register an account to enter the system.
<b>Pre- Requisite (Special or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff is in the Login page when starting the application.</li> <li>2. Staff must click on the “Sign Up” which is at the bottom if the Login page.</li> <li>3. Staff is in the Sign Up page after clicking the “Sign Up” from the Login page.</li> </ol>
<b>Expected Result</b>	Staff should be able to register an account and redirect to the homepage.



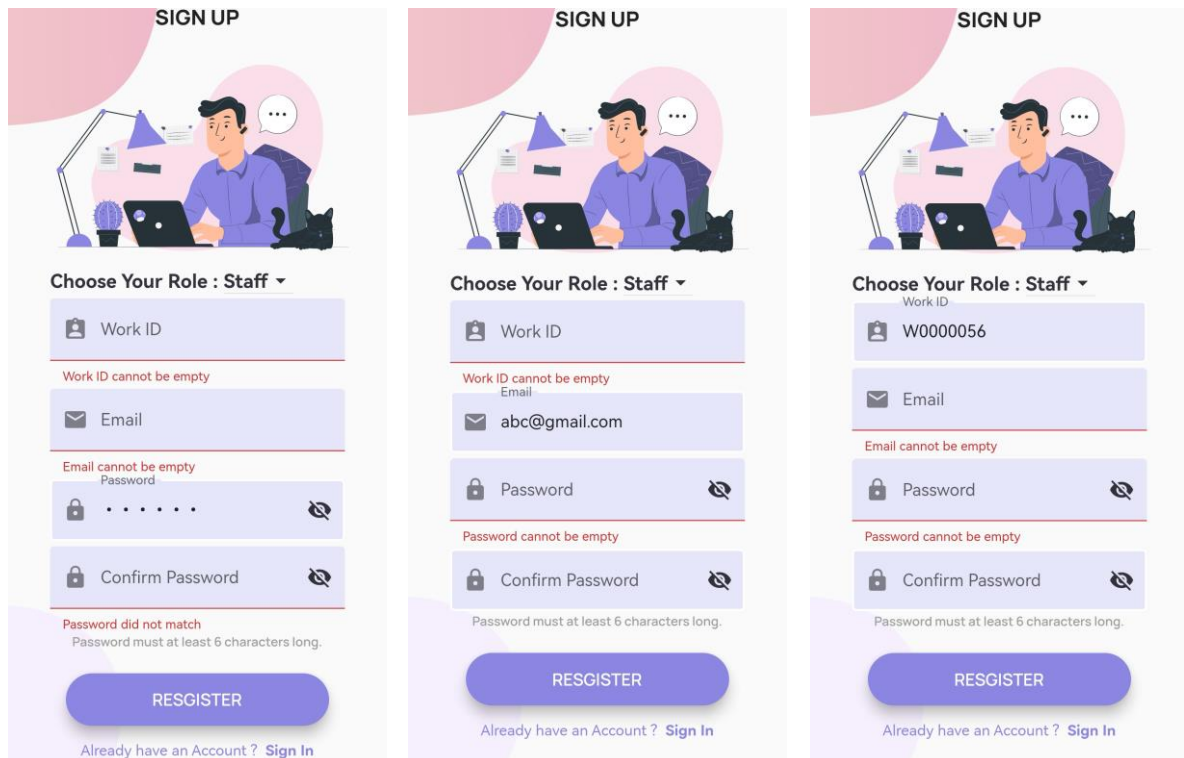
<b>Actual Result</b>	Staff able to register a user account successfully and has been redirect to the homepage with the new user data displayed on the screen.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Select role	Staff	
2. Input Work ID	W000001	
3. Input email address	abc113@gmail.com	
4. Input password	123456	
5. Input confirm password	123456	
6. Click “Register” button once finish filled up all the registration details	Register	
7. Redirect to the Homepage	Registration Successful	



**Figure 6.2.3.1: Test Result for test case TC\_RUA\_01**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Staff Module – Register User Account
<b>Reference Document</b>	authenticate.dart, background_signup.dart, sign_up.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_RUA_02
<b>Test Scenario (Purpose)</b>	To verify the functionality of register a user account which the registration is failed.
<b>Test Case Description</b>	Test the functionality of not filling in required details and unable to register an account.

<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff is in the Login page when starting the application.</li> <li>2. Staff must click on the “Sign Up” which is at the bottom if the Login page.</li> <li>3. Staff is in the Sign Up page after clicking the “Sign Up” from the Login page.</li> </ol>
<b>Expected Result</b>	Staff unable to register an account and will not be redirect to the homepage.
<b>Actual Result</b>	Staff unable to register a user account successfully and not redirect to the homepage.
<b>Test Result</b>	PASSED
Test Step	Test Data
1. Select role	Staff
2. Input is blank within 1 of the text fields	(blank)
3. Click “Register” button	Register
4. Redirect to the Homepage	Registration Successful



**Figure 6.2.3.2: Test Result for test case TC\_RUA\_02**

- **Edit Consultation Details**

<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Staff Module – Edit Consultation Details	
<b>Reference Document</b>	updateContent.dart, updateContent_screen.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_ECD_01	
<b>Test Scenario (Purpose)</b>	To verify the functionality of consultation details in the schedules list that is in the complete tab.	
<b>Test Case Description</b>	Test the functionality of editing the consultation details and save update the consultation details .	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff is in the Schedule page by clicking the “Schedule” on the bottom navigation tab.</li> <li>2. Staff must click on the “complete” from the horizontal navigation tab.</li> <li>3. Staff is in the complete tab of the schedule list.</li> </ol>	
<b>Expected Result</b>	Staff should be able to edit and save update the consultation details.	
<b>Actual Result</b>	Staff is able to make changes in the consultation details and save the changes of the consultation details.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Staff clicks on the “ View Consultation Details” to view the consultation details	“View Consultation Details”	
2. Staff click on the checkbox to select the areas of the acne present	Chin	
3. Staff click on the radio button to change the skin condition	Oily	
4. Staff click on the checkbox to select the product used by the patient	Sunscreen	
5. Staff select the consultation content from the dropdown list	Face Treatment	
6. Staff select the facial treatment from the dropdown list	Brightening Facial	
7. Staff input the important comment in the text field	skin oily on cheeks and forehead	

8. Staff clicks on the “Save” button

Successfully update

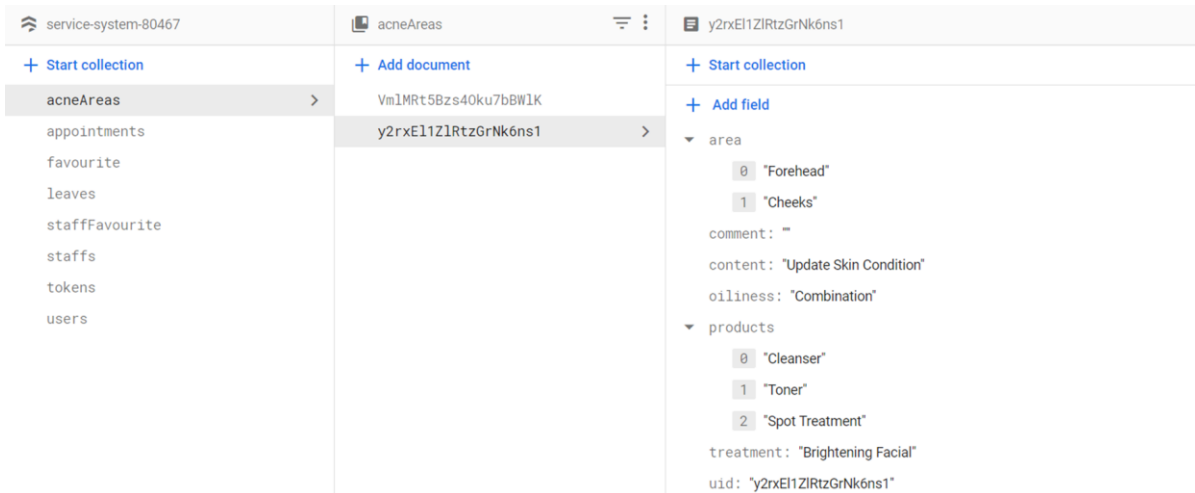
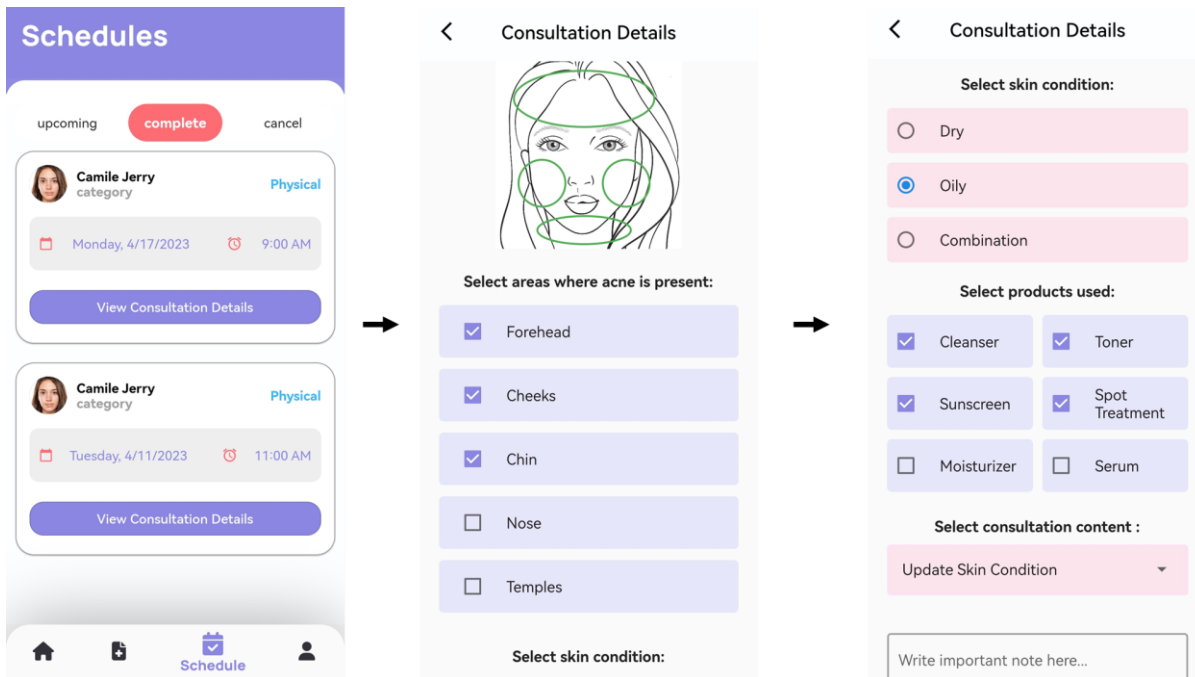
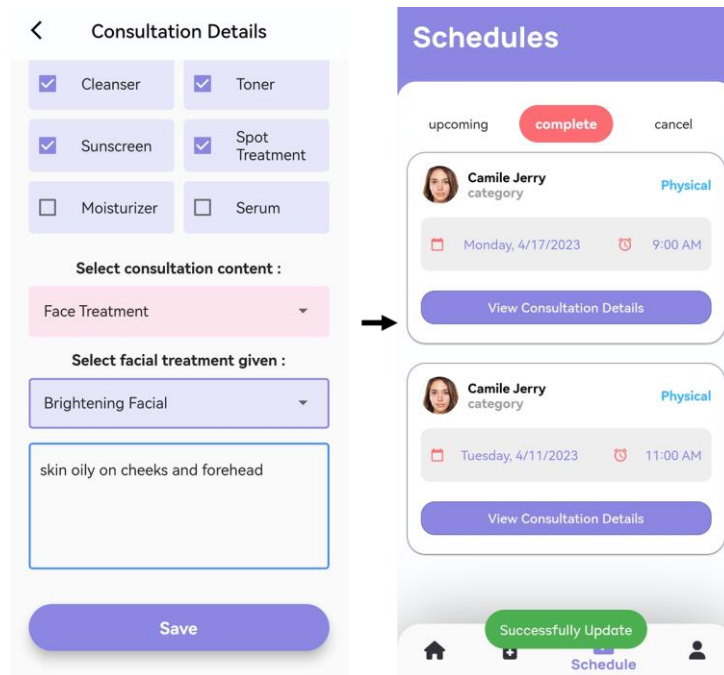
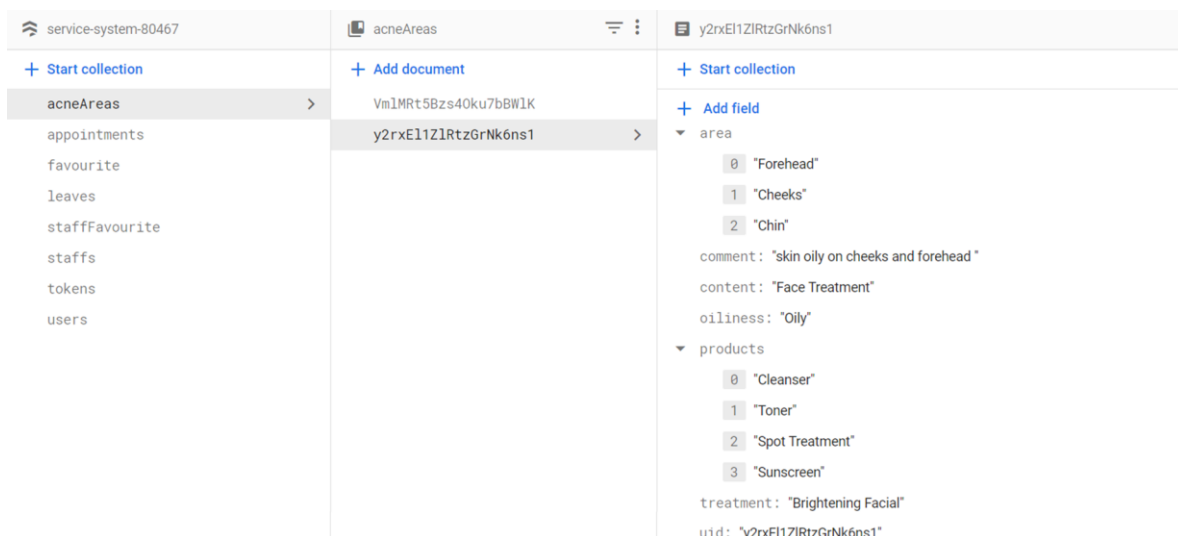


Figure 6.2.3.3: Initial Consultation Details in Cloud Firestore Database





**Figure 6.2.3.4: Test Result for test case TC\_ECD\_01**

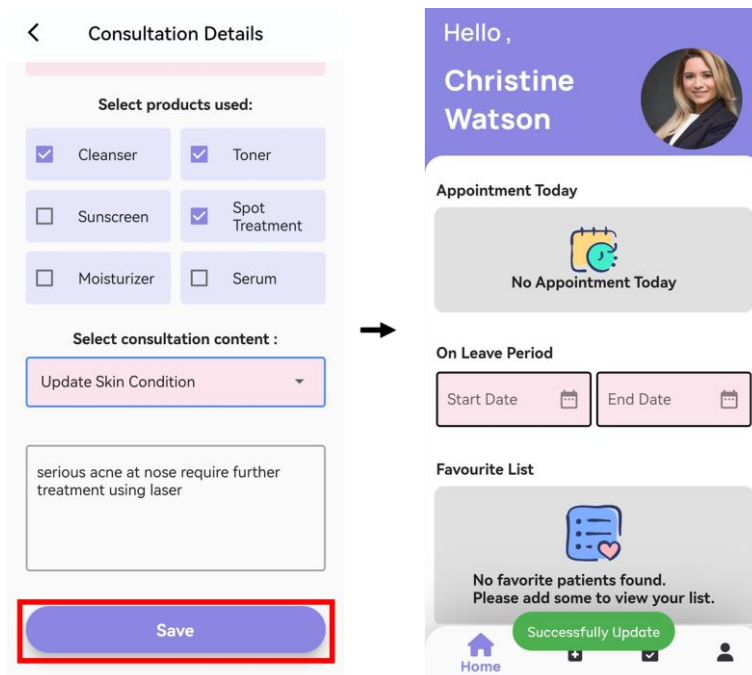
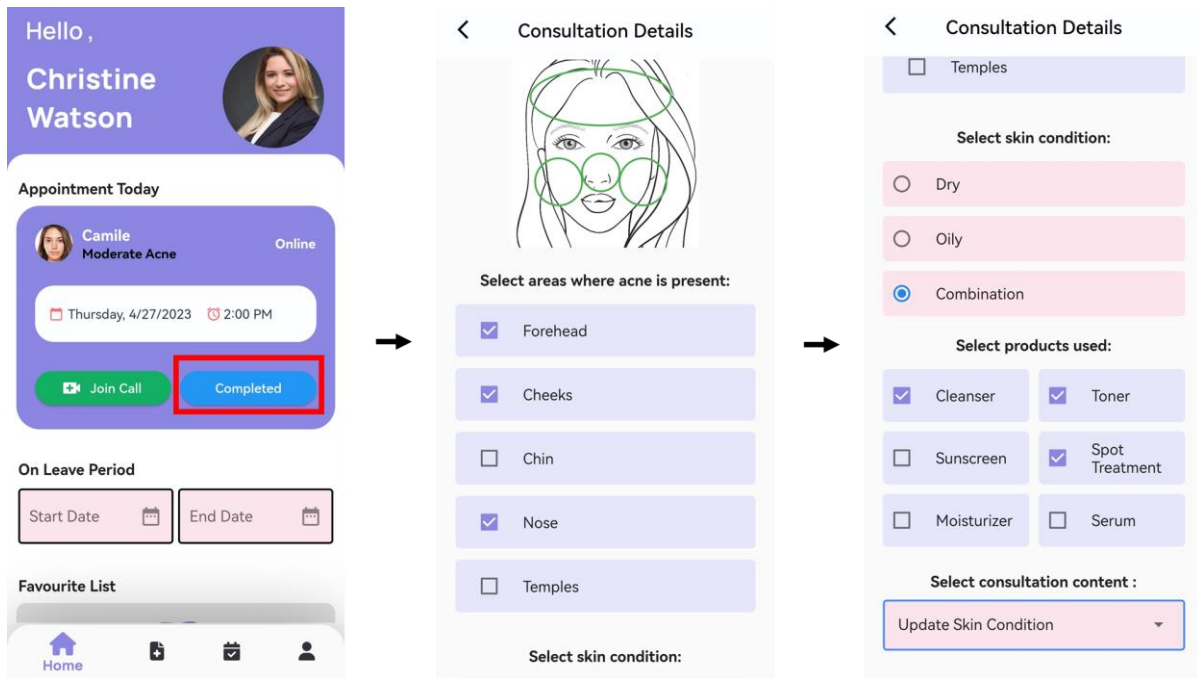


**Figure 6.2.3.5: Updated Consultation Details Store in Cloud Firestore Database**

- **Add Consultation Details**

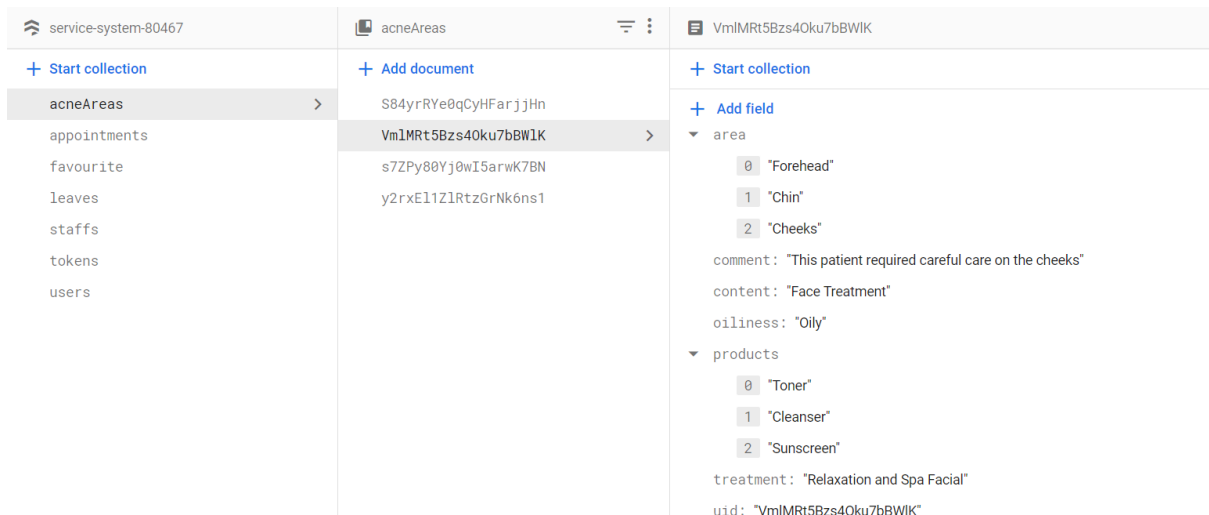
<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Staff Module – Add Consultation Details
<b>Reference Document</b>	viewdata_screen.dart, viewdata.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_ACD_01

<b>Test Scenario (Purpose)</b>	To verify the functionality of consultation details in the homepage when there is an appointment today.	
<b>Test Case Description</b>	Test the functionality of adding the new consultation details when the consultation session is completed.	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff is in the Homepage page and there is an appointment today.</li> <li>2. Staff has finished conducting the consultation session.</li> </ol>	
<b>Expected Result</b>	Staff should be able to add a new consultation detail.	
<b>Actual Result</b>	Staff is able to add a new consultation detail after completing the consultation details form.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Staff clicks on the “ Complete” to view the consultation details form	“ Complete”	
2. Staff click on the checkbox to select the areas of the acne present	Forehead, Cheeks, Nose	
3. Staff click on the radio button to change the skin condition	Combination	
4. Staff click on the checkbox to select the product used by the patient	Cleanser, Toner, Spot Treatment	
5. Staff select the consultation content from the dropdown list	Update Skin Condition	
6. Staff input the important comment in the text field		
7. Staff clicks on the “Save” button	Successfully update	



**Figure 6.2.3.6: Test Result for test case TC\_ACD\_01**





**Figure 6.2.3.7: Added Consultation Details in Cloud Firestore Database**

- **Edit Profile Details**

<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Staff Module – Edit Profile Details	
<b>Reference Document</b>	edit_staff_setting.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_EPD_01	
<b>Test Scenario (Purpose)</b>	To verify the functionality of profile details in the edit profile page.	
<b>Test Case Description</b>	Test the functionality of editing the details in the staff profile.	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Staff is in the Profile page by clicking the “Profile” on the bottom navigation tab.</li> <li>2. Staff must click on the profile picture which have the edit button attach to the profile picture in profile page.</li> <li>3. Staff is in the edit profile page after clicking the profile picture in the profile page.</li> </ol>	
<b>Expected Result</b>	Staff should be able to edit the profile details and submit save changes.	
<b>Actual Result</b>	Staff is able to edit the profile details and successfully save the changes make.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Input name	Christine Witson	
2. Input gender	Female	
3. Input job position	Facialist	
4. Input qualification	University of Cosmetic	
5. Input experience	4	
6. Input expertise area	Mild Acne	
7. Input training place	Perth	

8. Input phone number	012-4782456
9. Staff clicks on the “Save” button	Successfully update

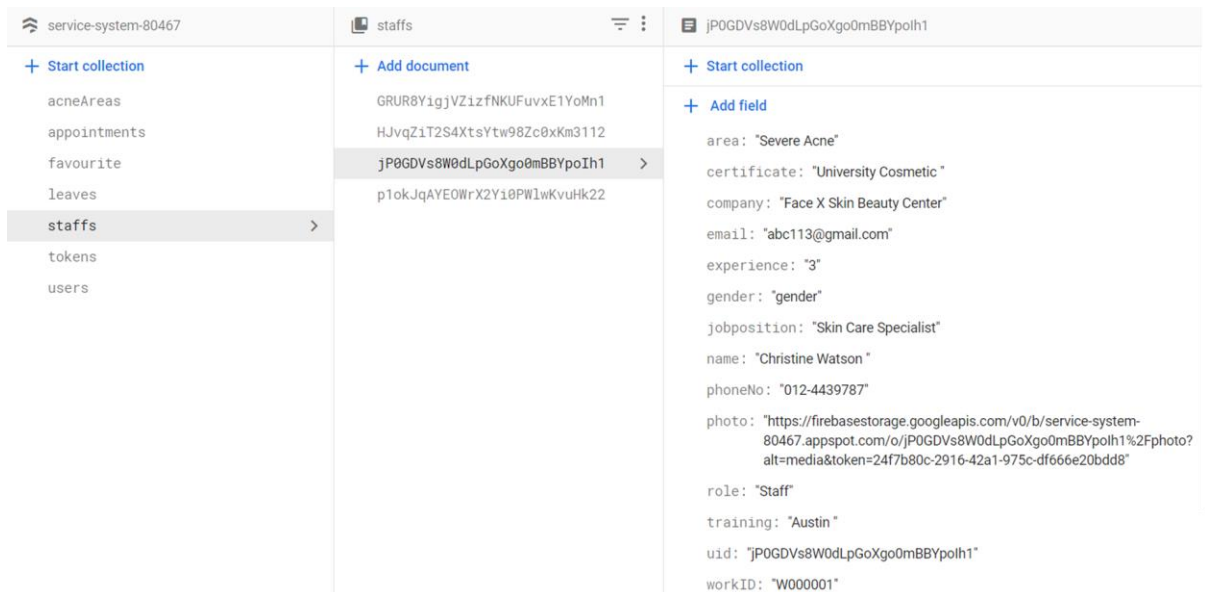


Figure 6.2.3.8: Initial Staff Profile Details in Cloud Firestore Database

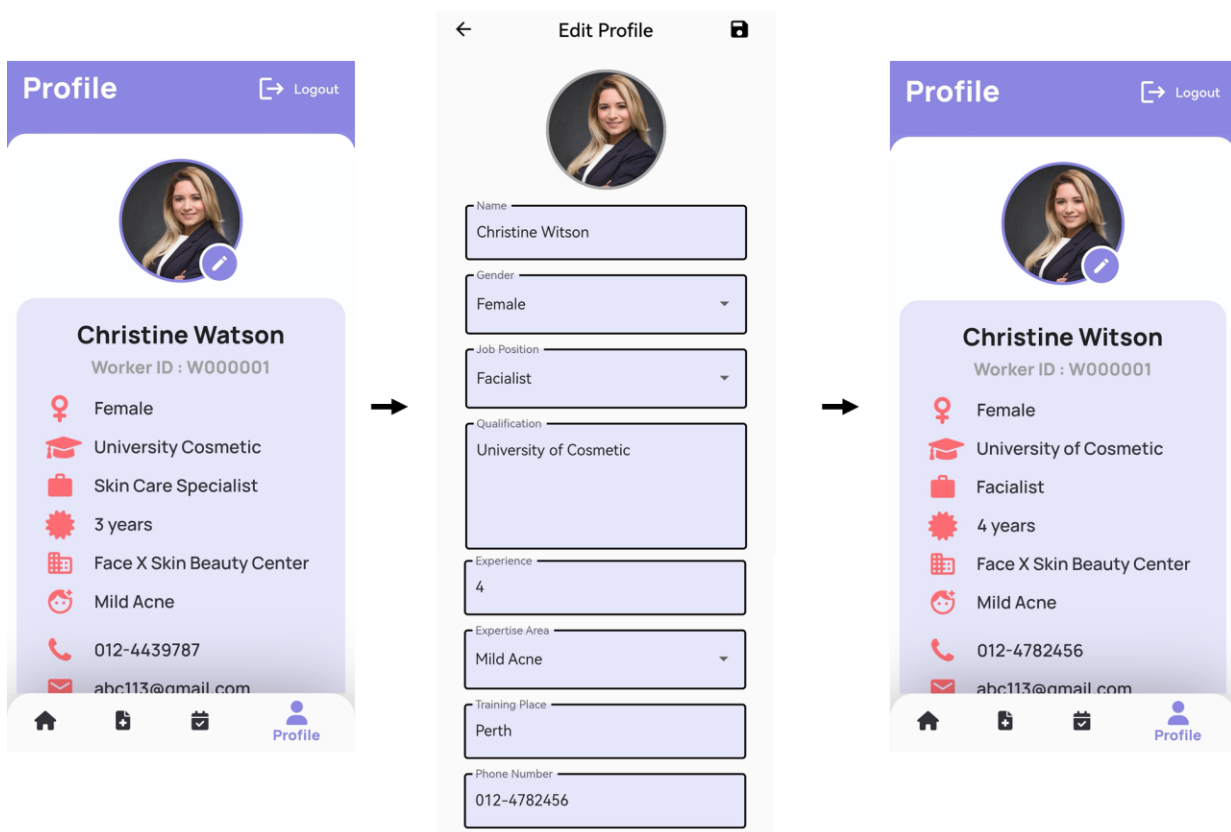
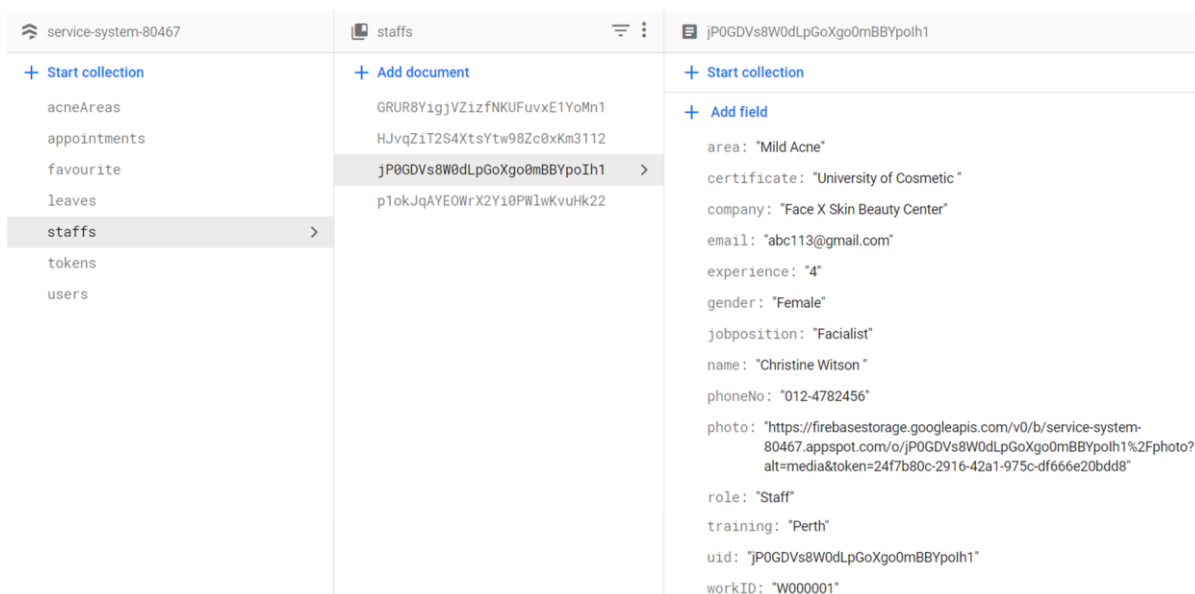


Figure 6.2.3.9: Test Result for test case TC\_EPD\_01

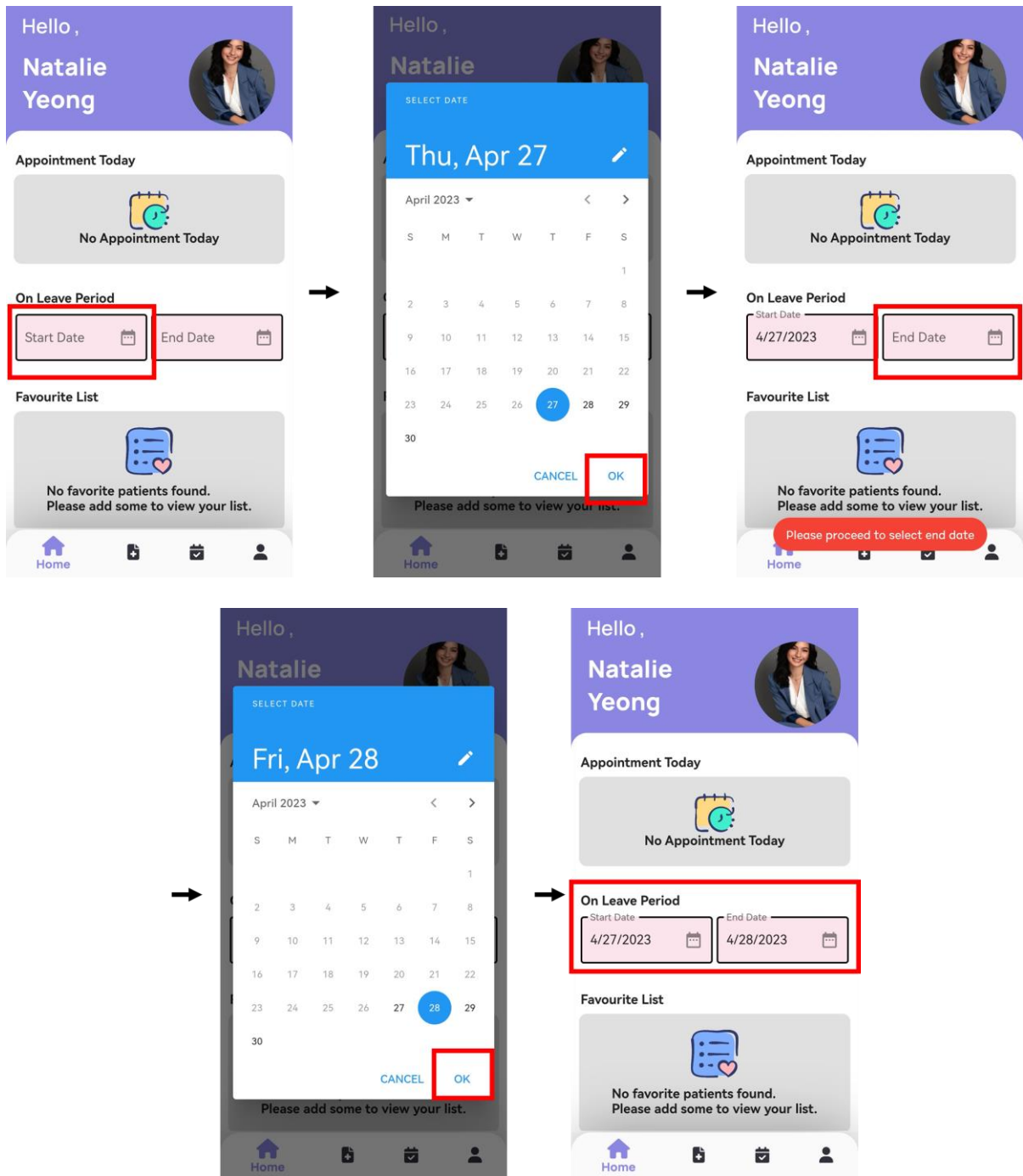


**Figure 6.2.3.10: Updated Staff Profile Details in Cloud Firestore Database**

- **Add On-Leave Period**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Staff Module – Add On-Leave Period
<b>Reference Document</b>	home_body.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_AOLP_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of on-leave period on the homepage which allow the staff to manage the appointment calendar.
<b>Test Case Description</b>	Test the functionality of adding the start date and end date to the on-leave period to manage the appointment calendar.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	1. Staff is in the Homepage after logging into the account.
<b>Expected Result</b>	Staff should be able to add the start date and end date of the on-leave period and the date will be displayed on the text fields.
<b>Actual Result</b>	Staff is able to add the start date and end date, the text fields has displayed the date respectively.
<b>Test Result</b>	PASSED

Test Step	Test Data
1. Input start date	4/27/2023
2. Input end date	4/28/2023
3. Staff view the start date and end date on the text field	-



**Figure 6.2.3.11: Test Result for test case TC\_AOLP\_01**

## 6.2.4 Patient Module

- **Register User Account**

<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Patient Module – Register User Account	
<b>Reference Document</b>	authenticate.dart, background_signup.dart, sign_up.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_RUA_01	
<b>Test Scenario (Purpose)</b>	To verify the functionality of register a user account which successfully register an account to the system.	
<b>Test Case Description</b>	Test the functionality of inserting new patient details such as entering email address, password and confirm password and register an account to enter the system.	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Patient is in the Login page when starting the application.</li> <li>2. Patient must click on the “Sign Up” which is at the bottom if the Login page.</li> <li>3. Patient is in the Sign Up page after clicking the “Sign Up” from the Login page.</li> </ol>	
<b>Expected Result</b>	Patient should be able to register an account and redirect to the homepage.	
<b>Actual Result</b>	Patient able to register a user account successfully and has been redirect to the homepage with the new user data displayed on the screen.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Select role	User	
2. Input email address	qbq@gmail.com	
3. Input password	123456	
4. Input confirm password	123456	
5. Click “Register” button once finish filled up all the registration details	Register	
6. Redirect to the Homepage	Registration Successful	

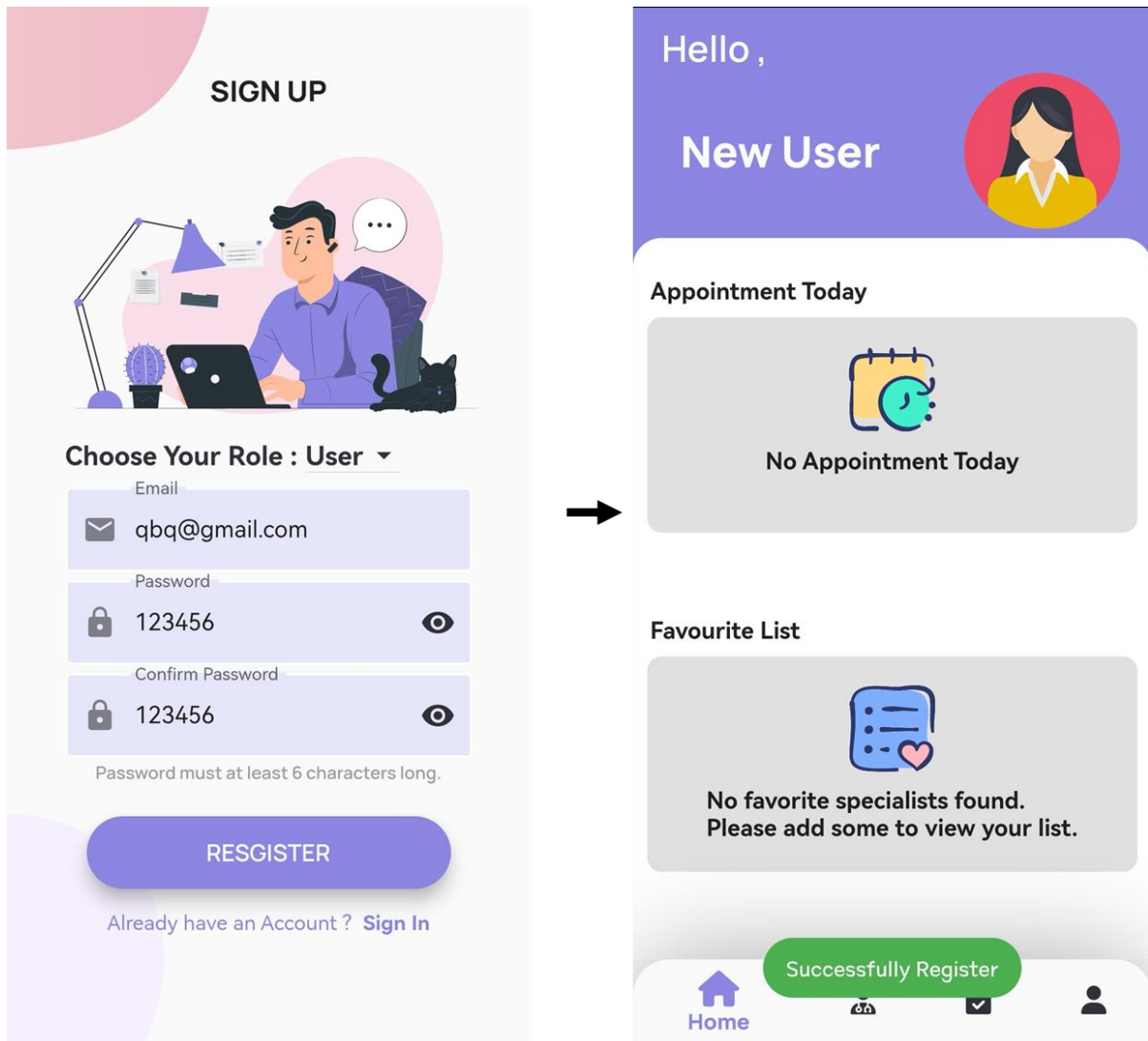
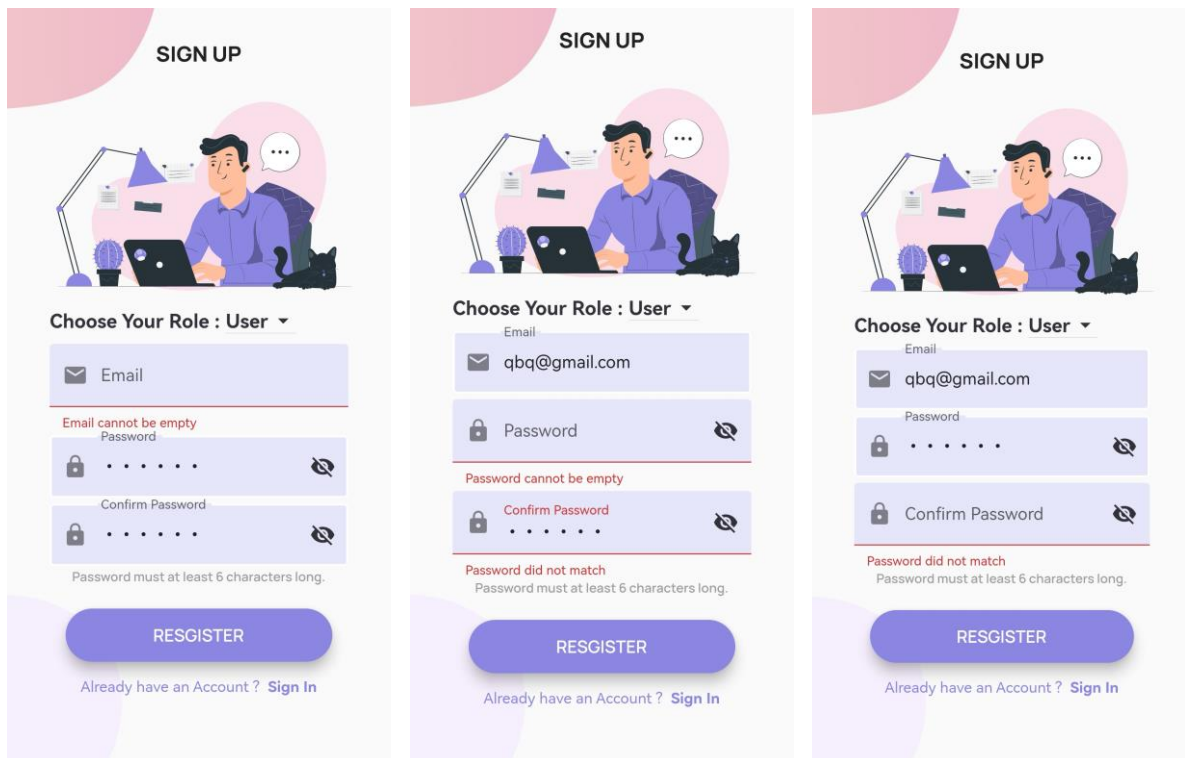


Figure 6.2.4.1: Test Result for test case TC\_RUA\_01

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Patient Module – Register User Account
<b>Reference Document</b>	authenticate.dart, background_signup.dart, sign_up.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_RUA_02
<b>Test Scenario (Purpose)</b>	To verify the functionality of register a user account which the registration is failed.
<b>Test Case Description</b>	Test the functionality of not filling in required details and unable to register an account.

<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Patient is in the Login page when starting the application.</li> <li>2. Patient must click on the “Sign Up” which is at the bottom of the Login page.</li> <li>3. Patient is in the Sign Up page after clicking the “Sign Up” from the Login page.</li> </ol>
<b>Expected Result</b>	Patient unable to register an account and will not be redirect to the homepage.
<b>Actual Result</b>	Patient unable to register a user account successfully and not redirect to the homepage.
<b>Test Result</b>	PASSED
<b>Test Step</b>	<b>Test Data</b>
1. Select role	User
2. Input is blank within 1 of the text fields	(blank)
3. Click “Register” button	Register
4. Redirect to the Homepage	Registration Successful

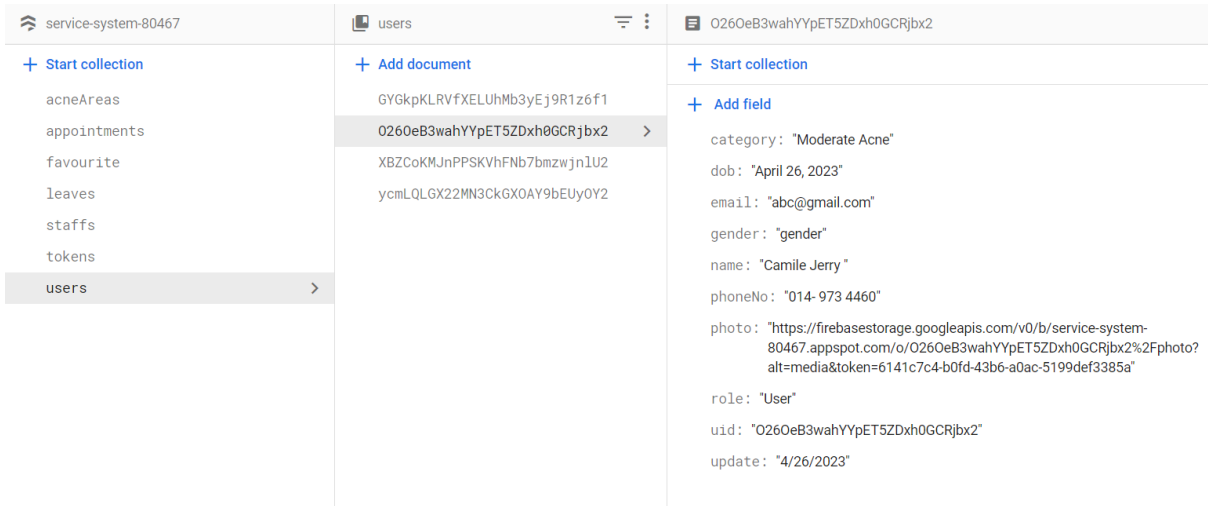


**Figure 6.2.4.2: Test Result for test case TC\_RUA\_02**

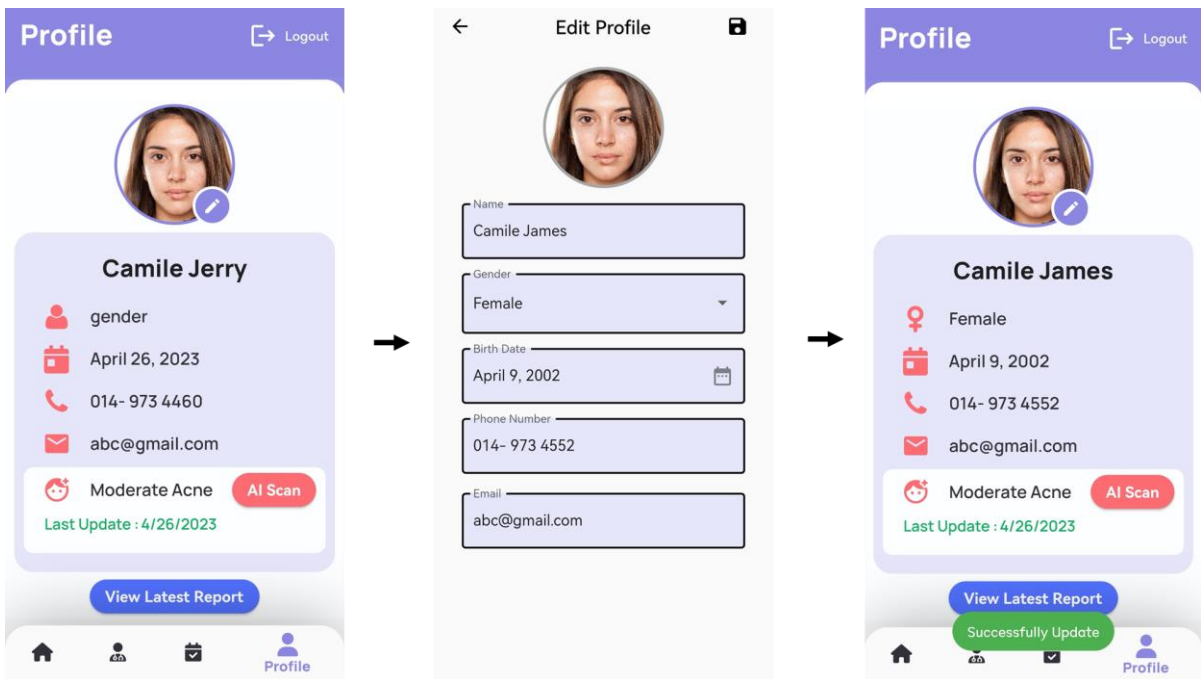


- **Edit Profile Details**

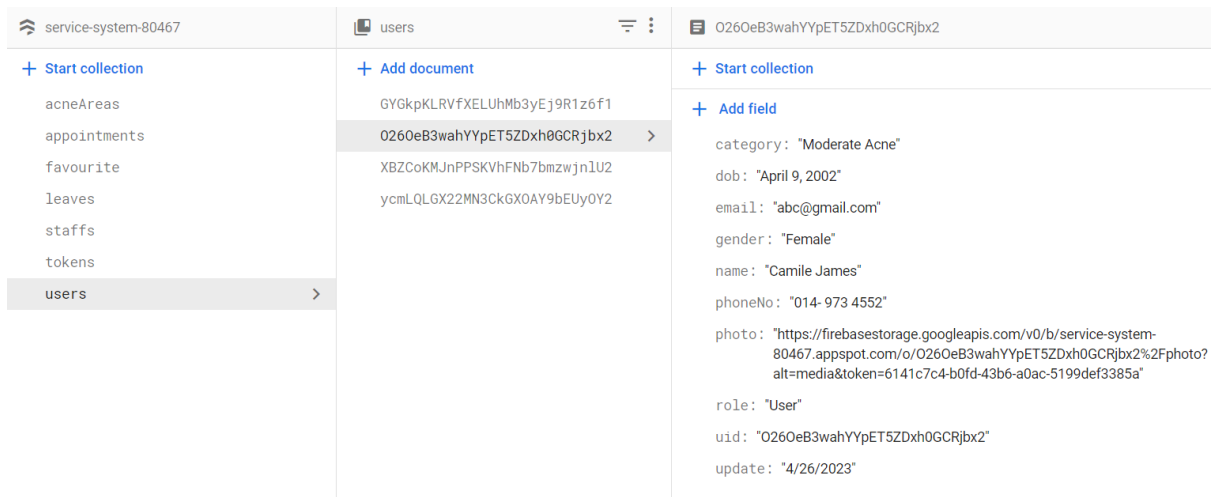
<b>Project Name</b>	Facial and Skincare Service System	
<b>Module Name</b>	Patient Module – Edit Profile Details	
<b>Reference Document</b>	edit_staff_setting.dart	
<b>Created By</b>	Low Chia Hui	
<b>Test Case ID</b>	TC_EPD_01	
<b>Test Scenario (Purpose)</b>	To verify the functionality of profile details in the edit profile page.	
<b>Test Case Description</b>	Test the functionality of editing the details in the patient profile.	
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Patient is in the Profile page by clicking the “Profile” on the bottom navigation tab.</li> <li>2. Patient must click on the profile picture which have the edit button attach to the profile picture in profile page.</li> <li>3. Patient is in the edit profile page after clicking the profile picture in the profile page.</li> </ol>	
<b>Expected Result</b>	Patient should be able to edit the profile details and submit save changes.	
<b>Actual Result</b>	Patient is able to edit the profile details and successfully save the changes make.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Input name	Camile James	
2. Input gender	Female	
3. Input birth date	April 9, 2002	
4. Input phone number	014- 973 4552	
5. Patient clicks on the “Save” button	Successfully update	



**Figure 6.2.4.3: Initial Patient Profile Details in Cloud Firestore Database**



**Figure 6.2.4.4: Test Result for test case TC\_EPD\_01**

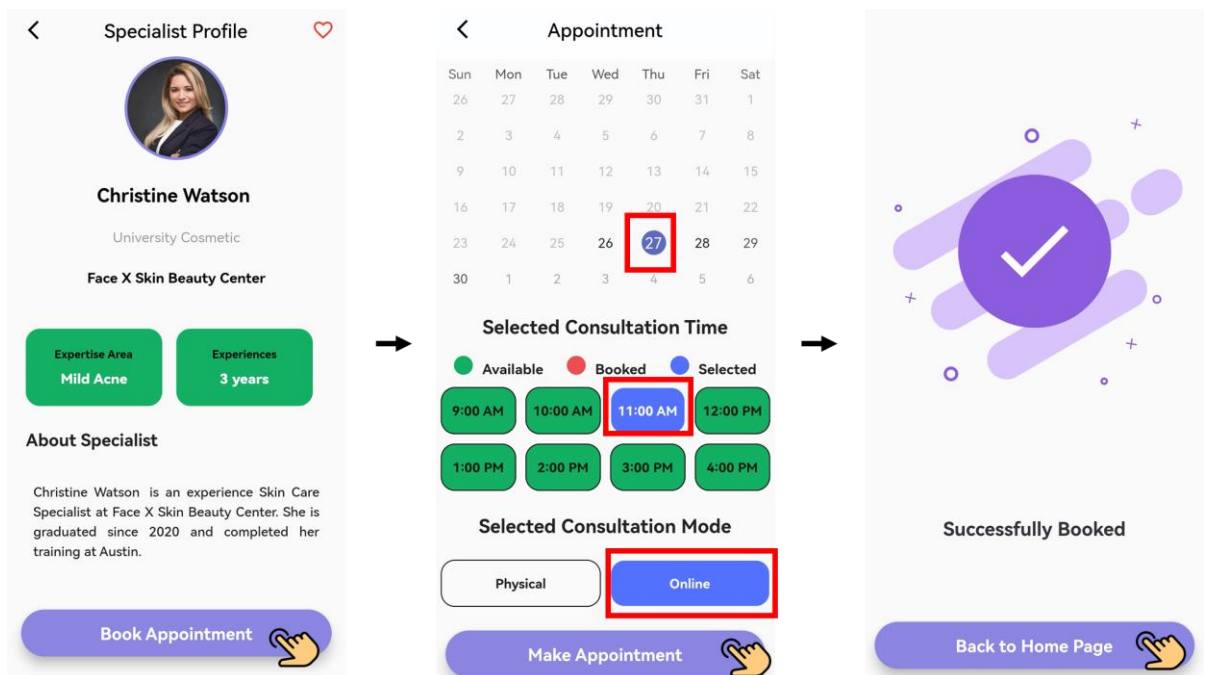


**Figure 6.2.4.5: Updated Patient Profile Details in Cloud Firestore Database**

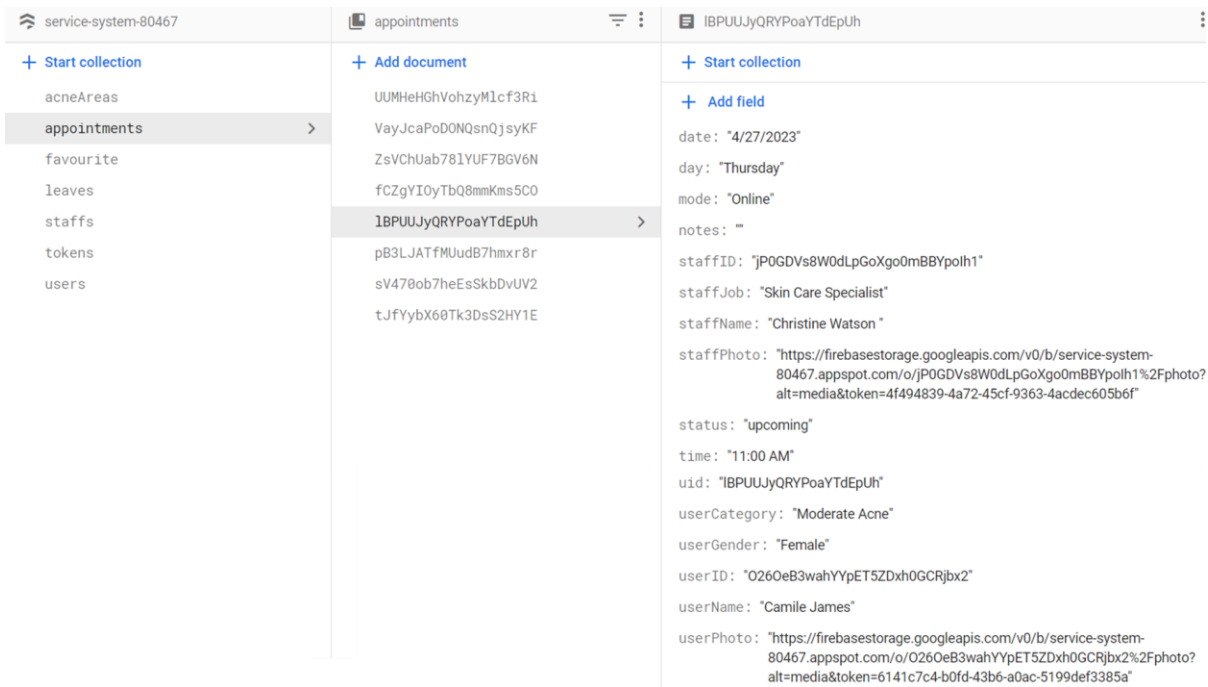
- **Add Appointment Booking Details**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Patient Module – Add Appointment Booking Details
<b>Reference Document</b>	booking_page.dart, success_booked.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_AABD_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of add appointment booking details in the booking page.
<b>Test Case Description</b>	Test the functionality of adding the appointment booking details with appointment date, time and mode.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Patient is in the Specialist page by clicking the “Specialist” on the bottom navigation tab.</li> <li>2. Patient must click on the selected specialist profile.</li> <li>3. Patient is in the specialist profile details.</li> </ol>
<b>Expected Result</b>	Patient should be able to add a new appointment booking details.
<b>Actual Result</b>	Patient is able to add the details of the appointment booking.
<b>Test Result</b>	PASSED

Test Step	Test Data
1. Patient clicks on the “Book Appointment”	“Book Appointment”
2. Redirect patient to booking page	-
3. Input appointment date	4/27/2023
4. Input appointment time	11:00 AM
5. Input appointment mode	Online
6. Patient clicks on the “Make Appointment” button	“Make Appointment”
7. Redirect patient to confirmation booking page	-
8. Patient clicks on the “Back to Home Page” to exit the confirmation booking page	“Back to Home Page”



**Figure 6.2.4.6: Test Result for test case TC\_AABD\_01**

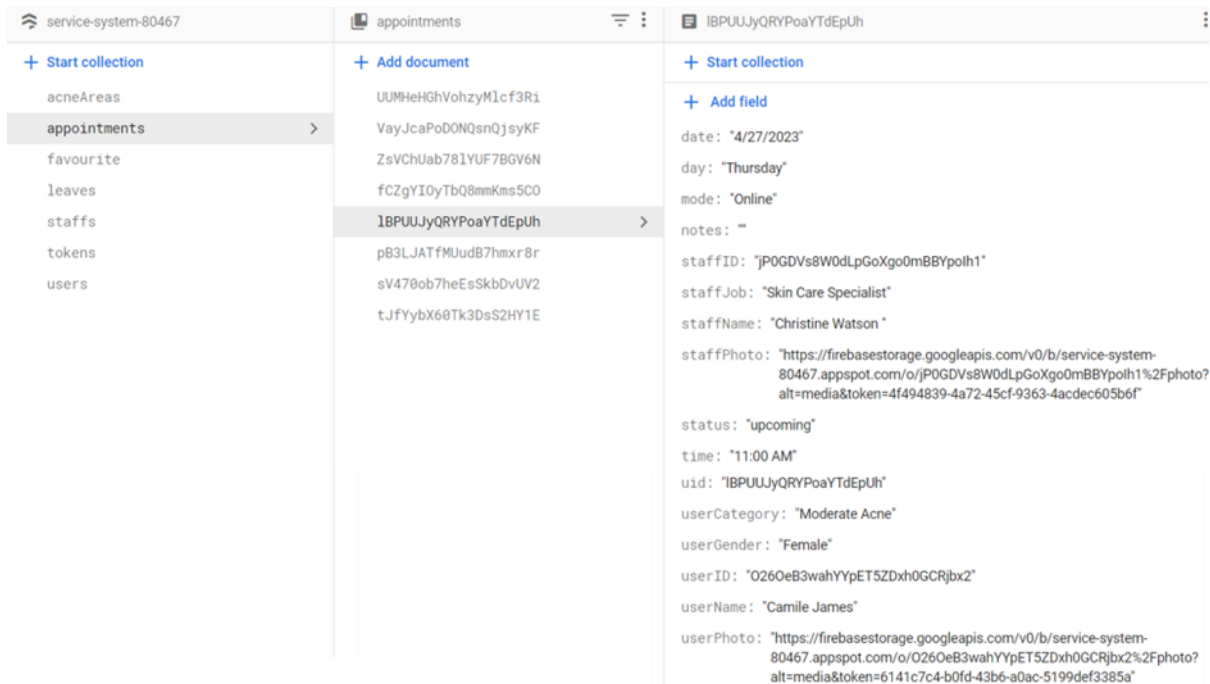


**Figure 6.2.4.7: Added Appointment Booking Details in Cloud Firestore Database**

- **Reschedule Appointment Booking**

<b>Project Name</b>	Facial and Skincare Service System
<b>Module Name</b>	Patient Module – Reschedule Appointment Booking
<b>Reference Document</b>	reschedule_page.dart, success_booked.dart
<b>Created By</b>	Low Chia Hui
<b>Test Case ID</b>	TC_RAB_01
<b>Test Scenario (Purpose)</b>	To verify the functionality of reschedule appointment booking details in the schedule page.
<b>Test Case Description</b>	Test the functionality of adding the reschedule appointment booking details with new appointment date, time and mode.
<b>Pre- Requisite (Special Conditions or Limitations)</b>	<ol style="list-style-type: none"> <li>1. Patient is in the Schedule page by clicking the “Schedule” on the bottom navigation tab.</li> <li>2. Patient is in upcoming tab or cancel tab in the schedule page.</li> </ol>
<b>Expected Result</b>	Patient should be able to reschedule a new appointment booking.

<b>Actual Result</b>	Patient is able to reschedule the new appointment booking with new appointment date, time and mode.	
<b>Test Result</b>	PASSED	
<b>Test Step</b>	<b>Test Data</b>	
1. Patient clicks on the “Reschedule”	“Reschedule”	
2. Redirect patient to reschedule booking page	-	
3. Input appointment date		
4. Input appointment time		
5. Input appointment mode		
6. Patient clicks on the “Reschedule Appointment” button	“Reschedule Appointment”	
7. Redirect patient to confirmation booking page	-	
8. Patient clicks on the “Back to Home Page” to exit the confirmation booking page	“Back to Home Page”	



**Figure 6.2.4.8: Initial Appointment Booking Details in Cloud Firestore Database**

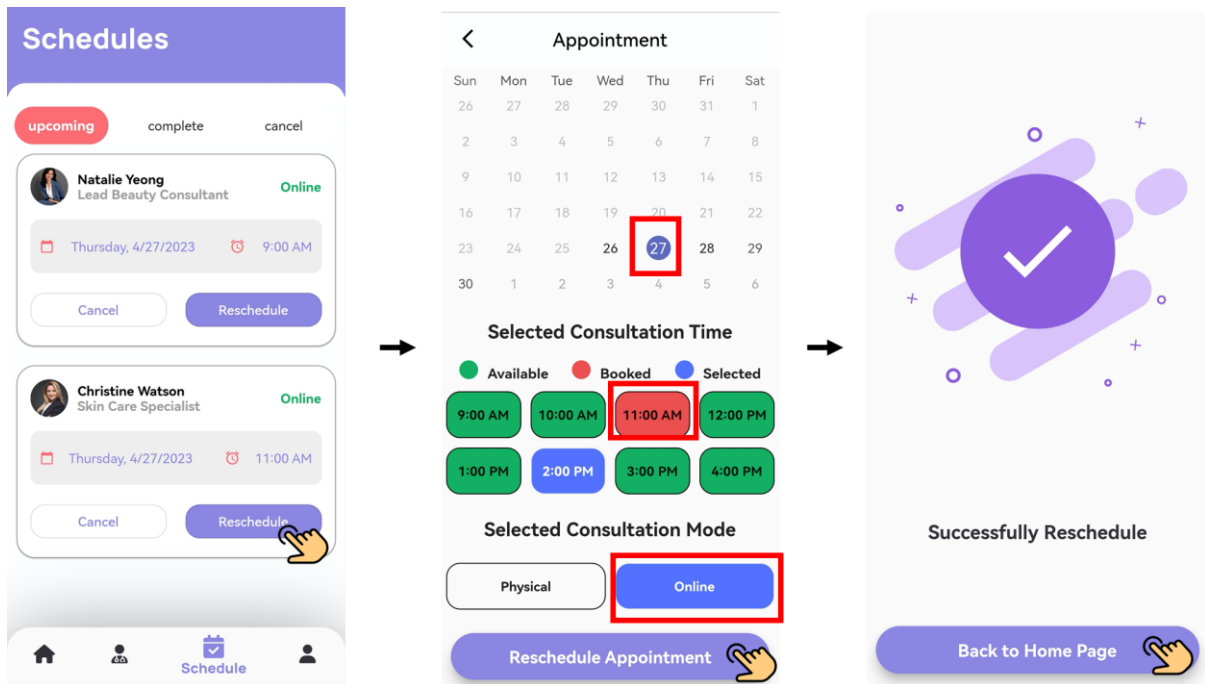


Figure 6.2.4.9: Test Result for test case TC\_RAB\_01

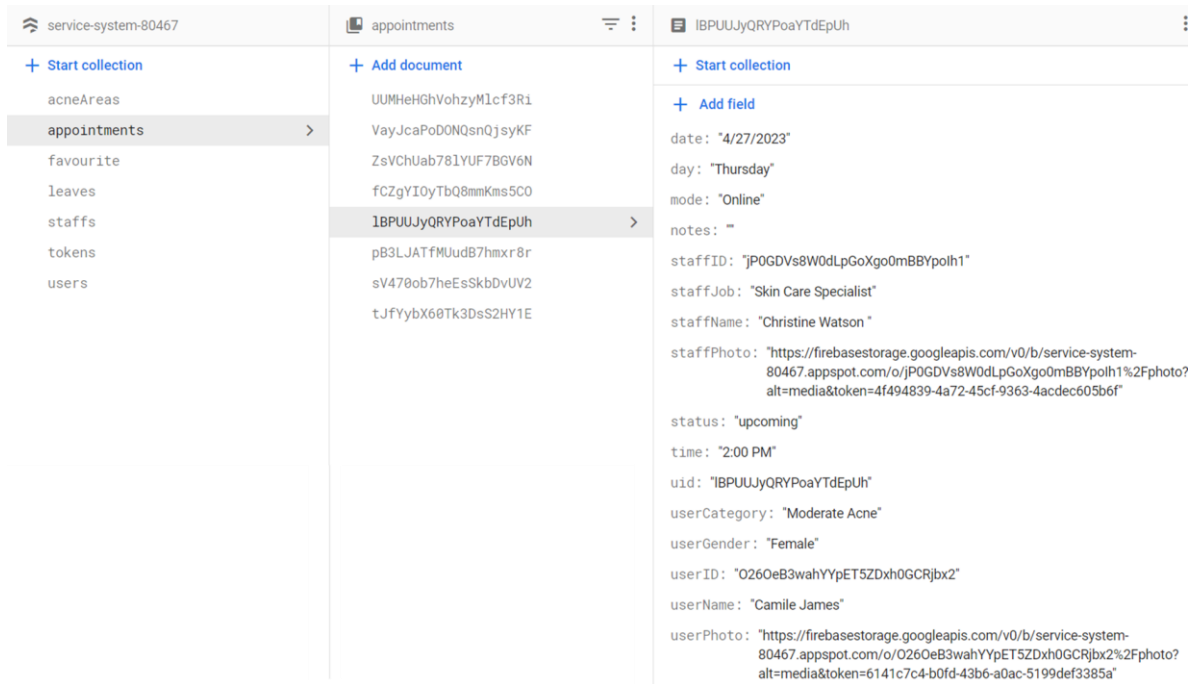


Figure 6.2.4.10: Updated Appointment Booking Details in Cloud Firestore Database

## **6.3 Project Challenges**

### **6.3.1 Resolved Project Challenges**

The process of implementing mobile applications has encountered numerous challenges, but these have been overcome through various solutions.

#### **1. Incompatibles of Library Version**

The incompatibles of Library Version occur when running the command flutter build APK in the terminal to build the APK file where the version of the SDK build, and platform tools version is not correct and causes the failures in building the APK file. This issue can be easily solved by changing the SDK version to the respective supported version.

#### **2. Understanding on deep learning neural networks**

Performing research on deep learning neural networks requires a comprehensive understanding of different factors such as model selection, pre-trained model mechanisms, and layer construction. It is also crucial to analyze the latest deep learning models and compare them to determine the best baseline model from those accessible in open-source resources like TensorFlow and Pytorch.

#### **3. Integration of ZEGOCLOUD and TensorFlow Lite to Flutter App**

There are several technical challenges in configuring and optimizing the integration of ZEGOCLOUD and TensorFlow Lite for performance. The integration process involves converting the TensorFlow model to TensorFlow Lite format and deploying it to flutter by including the dependencies. Integration of both ZEGOCLOUD and TensorFlow Lite required expertise, as well as knowledge of mobile app development using Flutter.



### **6.3.2 Unresolved Project Challenges**

Despite efforts to overcome obstacles, certain challenges remain unresolved.

#### **1. Accuracy of the TensorFlow Lite Model**

Even after deployment, the TensorFlow Lite model may still exhibit some misclassification due to the limitations of its generalization power acquired during training. Moreover, improving the model's accuracy may require an extensive dataset of images. Furthermore, to enhance the model's ability to learn complex features from the data, one can increase the number of layers in the transfer learning approach. In addition, optimizing hyperparameters like the learning rate and batch size is also needed to lead better accuracy.

#### **2. Implementation bugs in Mobile Application**

The implementation of dart code was performed well during the development and actual use. However, there are some bugs in the mobile application which unable to solve, for instance, the user uploads an image but did not want to save change on the image when editing the profile details, the image will somehow change according to the images that the user upload.

#### **3. Incompatibles of Plugin Version**

The version of plugin is incompatibles occurs when adding the dependencies of TensorFlow Lite. The plugin is using a deprecated version of the Android embedding. This required the plugin author to migrate the plugin to the V2 embedding. Therefore, this issue cannot be solved, and it may cause an unexpected runtime failures or future build failures when running the plugin.

## 6.4 Objectives Evaluation

The stated objectives have been successfully accomplished by the proposed system.

- **To provide video call consultation using video call plugin**

In this project, the staff make a video call to reach the patient and the patient is required to join the video call. This video call is one-to-one video call room which only allow two users in the room. The staff is able to conduct an online consultation when the appointment mode is online. While the patient can join the video call when they booked for online consultation session with the specialist. The staff and patient need to give the permission access of the device for camera and microphone so that they are able to join the video call room successfully.

- **To develop appointment booking in the mobile application**

The user is able book their desire date for the consultation with the specialist without asking for the availability of the time slot that the specialist have with is not convenient through phone call or texting. The appointment booking features in this project allow the user to book their consultation time slot by choosing the specialist that they want to consult and select the date from the calendar that have been provided, then select the time and mode of the appointment.

- **To implement skin classification on acne type**

In this project, both staff and patient can use the features of the skin classification to diagnose the skin. The staff and patient can upload the image from the gallery or use phone camera to capture the photo. The staff can only use this feature when the patient is having physical consultation session with them. The patient can use this feature whenever they want because it is a feature that available in their profile. The patient can use this feature to find the specialist that suit them because the suggested specialist is according to the classification result that they obtain after uploading image or taking photo.

## **6.5 Concluding Remark**

In conclusion, the goals of the project have been met, and the solution to the inefficient communication between staff and patients has been implemented. The staff and patient are allowed to have a virtual consultation via online platform which may assist to address the issue of the patient when they are not present in the location of the beauty center. The staff can follow up the skin condition of the patient through online platform to guide the patient regarding their routine for skincare. Other than that, the mobile application also helps to solve the problem of patient when booking for appointment, where the system allows the patient to book their appointment using the build in calendar that sync with the staff. The staff can also manage the calendar by adding their on-leave period to avoid the patient to book the slot when there are not around. Lastly, the system has provided the classification features which allow the staff and patient to utilize the advancement of the machine learning to gain the information of the skin condition. The patient can know their skin category after scanning their face using the classification features, while the specialist can use the features when the patient is visiting the beauty center physically to ensure the method of using the features is correct.

# Chapter 7

## Conclusion and Recommendation

### 7.1 Conclusion

All in all, the primary objective of this proposed system is to address the issue that arises from the manual diagnosis of a skin problem and the physical consultation using the technology gadget which is smartphones. The implementation of AI scan can help to prevent the time spend on the diagnose for the skin problem and the video call can help to improve the communication between the consultant and the customer. Furthermore, the suggested method may also assist in lowering the amount of paper that is used since it allows the consultant to keep treatment notes inside the system, where they can be written alongside other relevant notes. Moreover, the system also helps the patient to book an appointment by themselves which is more reliable as they do not need to discuss the time with their consultant. They can book the time slot based on the availability slot on the appointment booking calendar. Lastly, utilizing the advancement of the technology and the technique can help to improve the quality of the treatment and services.

The proposed system has helped to improve and enhance the traditional diagnose and the consultation session which is more convenient and workable to manage the consultation of the customer for the skincare clinic. The contribution for this project is boosting the confidence of the customer to the skincare clinic where the online consultation can help to build a stronger bond relationship between the customer and the consultant. Consultant can monitor the skin condition of the customer easily and provide appropriate treatment and advice for the customer which can help to fasten the progress of the healing for the skin. Furthermore, the proposed system has also help the reduce the burden of the consultant which reduce the time of the consultant where they do not need to observe the skin condition manually but they can use the build in AI technologies which is the AI skin classification to obtain the result of the skin condition for the customer and give the advice and guidance for the treatment that need to be done.

After doing this project, it helps me to explore more software and online platform which is available in the market. I have also learned a new programming language which is dart

programming language that use in Flutter to build mobile application. Other than that, I have also learnt how to do the backend part for a system by connect the interface to the Firebase which is a cloud database. I have also learnt how to deploy a model into mobile application which is interesting to know and explore because it is the trend of the market where the machine learning technology is widely utilize in every industry to help in improving the efficient of the work.

In short, this project has been completed and achieving the objectives that have been stated and it helps me to enhance my skills in programming and researching the resources.

## 7.2 Recommendation

For this service system, there are some improvements or enhancement can be done for this project. Firstly, the accuracy for the classification of the acne is not high which cause the problem in classifying the correct result of the acne type. Hence, the more dataset is needed to train the model in order to get higher accuracy. Other than that, the notification for the today appointment cannot run in the background. When the application is close the notification will not pop out even though there is an appointment today. The notification will only pop up when the user opens the application which is not reliable. Other than that, an admin login can be implemented so handle the account registration for the staff. Therefore, the user will not have the access to view the registration form for the staff and try to login as the staff which is more reliable and secure. The rating review for the specialist can be implemented so that the patient can have a reference when choosing the specialist that they want to consult. The apply leave form can be included as a formal way of applying leave from the work.

## REFERENCES

- [1] A. Palmer, "How acne is diagnosed and graded," *Verywell Health*, 15-Sep-2021. [Online]. Available: <https://www.verywellhealth.com/acne-diagnosis-15760>.
- [2] "Helping Malaysians break free from acne", *The Star*, 2022. [Online]. Available: <https://www.thestar.com.my/metro/community/2016/05/07/helping-malaysians-break-free-from-acne>.
- [3] "Why is a skincare consultation so important?," *Sanova Dermatology*, 14-Feb-2023. [Online]. Available: <https://www.sanovadermatology.com/skin-care/why-is-a-skincare-consultation-so-important/>.
- [4] "Why virtual beauty consultations are here to stay: News," *HRA Global*, 18-Jun-2021. [Online]. Available: <https://hra-global.com/news/beauty-iphone-beholder-virtual-beauty-consultations-here-to-stay/>.
- [5] "5 benefits of online scheduling for Healthcare practices," *PatientPop*, 24-Mar-2022. [Online]. Available: <https://www.patientpop.com/whitepapers/3-benefits-of-online-scheduling/>. [Accessed: 27-Apr-2023].
- [6] "What is Deep Learning and How Does It Work?," *SearchEnterpriseAI*, 2022. [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network>.
- [7] A. Biswal, "Top 10 Deep Learning Algorithms You Should Know in 2022", *Simplilearn*, 2022. [Online]. Available: <https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm>.
- [8] N. Pancholi, S. Goel, R. Nijhawan and S. Gupta, "Classification and Detection of Acne on the Skin using Deep Learning Algorithms," *2021 19th OITS International Conference on Information Technology (OCIT)*, 2021, pp. 110-114, doi: 10.1109/OCIT53463.2021.00032.

- [9] Quattrini A, Boër C, Leidi T, Paydar R. A Deep Learning-Based Facial Acne Classification System. *Clin Cosmet Investig Dermatol*. 2022;15:851-857 <https://doi.org/10.2147/CCID.S360450>
- [10] Alarifi, J.S., Goyal, M., Davison, A.K., Dancey, D., Khan, R., Yap, M.H. (2017). Facial Skin Classification Using Convolutional Neural Networks. In: Karray, F., Campilho, A., Cheriet, F. (eds) *Image Analysis and Recognition. ICIAR 2017. Lecture Notes in Computer Science()*, vol 10317. Springer, Cham. [https://doi.org/10.1007/978-3-319-59876-5\\_53](https://doi.org/10.1007/978-3-319-59876-5_53)
- [11] "Want to take your Medspa a cut above the competition?", *Go.zenoti.com*, 2022. [Online]. Available: [https://go.zenoti.com/medspa-software-b-cp/?utm\\_source=capterra](https://go.zenoti.com/medspa-software-b-cp/?utm_source=capterra).
- [12] "Aesthetic Clinic - Beauty Clinic Appointment Software", *WESS - Beauty, Spa & Salon Software*, 2022. [Online]. Available: <https://wessconnect.com/aesthetic-clinic/>.
- [13] "Skin Clinic Software - Nookal", *Nookal.com*, 2022. [Online]. Available: <https://www.nookal.com/info/skin-clinic-software>.
- [14] "Aesthetic Clinic Software | Medical Record system | Clinicminds", *Clinicminds*, 2022. [Online]. Available: <https://www.clinicminds.com/>.
- [15] "Tf.keras.applications.efficientnet\_v2.EFFICIENTNETV2B0 : tensorflow V2.12.0," *TensorFlow*. [Online]. Available: [https://www.tensorflow.org/api\\_docs/python/tf/keras/applications/efficientnet\\_v2/EfficientNetV2B0](https://www.tensorflow.org/api_docs/python/tf/keras/applications/efficientnet_v2/EfficientNetV2B0). [Accessed: 27-Apr-2023].
- [16] "Acne grading classificationdataset", *Kaggle.com*, 2022. [Online]. Available: <https://www.kaggle.com/datasets/rutviklathiyateksun/acne-grading-classificationdataset>.
- [17] "Acne: Treatment, Types, Causes & Prevention", *Cleveland Clinic*, 2022. [Online]. Available: <https://my.clevelandclinic.org/health/diseases/12233-acne>.
- [18] "Transfer learning and fine-tuning : Tensorflow Core," *TensorFlow*. [Online]. Available: [https://www.tensorflow.org/tutorials/images/transfer\\_learning](https://www.tensorflow.org/tutorials/images/transfer_learning).



- [19] “Convert Tensorflow Models : Tensorflow Lite,” *TensorFlow*. [Online]. Available: [https://www.tensorflow.org/lite/models/convert/convert\\_models](https://www.tensorflow.org/lite/models/convert/convert_models).
- [20] Microsoft, “Visual studio code - code editing. redefined,” *RSS*, 03-Nov-2021. [Online]. Available: <https://code.visualstudio.com/>.
- [21] “Build apps for any screen,” *Flutter*. [Online]. Available: <https://flutter.dev/>.
- [22] “Dart programming language,” *Dart*. [Online]. Available: <https://dart.dev/>.
- [23] “Tensorflow,” *TensorFlow*. [Online]. Available: <https://www.tensorflow.org/>.
- [24] K. Team, “Simple. flexible. powerful.,” *Keras*. [Online]. Available: <https://keras.io/>.

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: Trimester 3 Year 3</b>	<b>Study week no.: 1</b>
<b>Student Name &amp; ID: Low Chia Hui (20ACB01686)</b>	
<b>Supervisor: Ts Lai Siew Cheng</b>	
<b>Project Title: Facial and Skincare Service System</b>	

## 1. WORK DONE

Install and set up the TensorFlow environment.

Identify the scope that need to be done in FYP 2.

Seeking help from the supervisor regarding the work and requirement for FYP2.

## 2. WORK TO BE DONE

Read the documentation for the Flutter.

Find suitable resources for reference developing the core features.

Planning the task that need to be carry for completing the FYP 2.

## 3. PROBLEMS ENCOUNTERED

SDK version incompatible

## 4. SELF EVALUATION OF THE PROGRESS

Prepare all the set up before hand to ease the progress.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: Trimester 3 Year 3</b>	<b>Study week no.: 2</b>
<b>Student Name &amp; ID: Low Chia Hui (20ACB01686)</b>	
<b>Supervisor: Ts Lai Siew Cheng</b>	
<b>Project Title: Facial and Skincare Service System</b>	

## 1. WORK DONE

Read the documentation of the Flutter and get familiar with the Dart Programming.  
Find relevant online tutorial to learn how to code using Dart programming.  
Find some useful resources as reference and guideline for the project.  
Complete writing the Chapter 2 for the FYP report.

## 2. WORK TO BE DONE

Complete the feature for different user login and registration authentication page.  
Write Chapter 1 for the FYP report.

## 3. PROBLEMS ENCOUNTERED

Time spending for self-learning and finding the solution to solve the problem.

## 4. SELF EVALUATION OF THE PROGRESS

Progress is slow because require time to get familiar a new programming language.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: Trimester 3 Year 3</b>	<b>Study week no.: 3</b>
<b>Student Name &amp; ID: Low Chia Hui (20ACB01686)</b>	
<b>Supervisor: Ts Lai Siew Cheng</b>	
<b>Project Title: Facial and Skincare Service System</b>	

## 1. WORK DONE

Design the UI for other features in the system.  
Search the relevant design as reference for the UI design.  
Complete the Chapter 1 for the FYP report.

## 2. WORK TO BE DONE

Finalize the UI design for the system.  
Write Chapter 3 for the FYP report.  
Prepare to code the system program.

## 3. PROBLEMS ENCOUNTERED

UI design require a lot of consideration on the flow of the system and the suitability of the function for the system.  
Consideration also include user friendly and simplicity of the application.

## 4. SELF EVALUATION OF THE PROGRESS

Progress is slow because spending time in designing the interface of the application.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 4
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Code for the profile with new layout is done.  
Complete writing the design specification in Chapter 3.

## 2. WORK TO BE DONE

Prepare to code for the interface of the program.  
Complete the report writing for Chapter 3.  
Write Chapter 4 for the FYP report.

## 3. PROBLEMS ENCOUNTERED

Spending time in searching the code for the UI design that using Flutter.

## 4. SELF EVALUATION OF THE PROGRESS

Require more time to find the suitable design for the UI design that have been draft out.



\_\_\_\_\_  
Supervisor's signature



\_\_\_\_\_  
Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 5
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Code for the UI design for the patient list and specialist interface of the system is partially done.

Complete the Chapter 3 for FYP report.

## 2. WORK TO BE DONE

Connect new interface data with database.

Write the Chapter 4 for the FYP report.

## 3. PROBLEMS ENCOUNTERED

Finding the suitable reference for the UI design.

## 4. SELF EVALUATION OF THE PROGRESS

Progress is delay because time span on the searching for the UI design.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 6
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Prepare to code for schedule interface.  
Complete the code for display list of patients and specialist.

## 2. WORK TO BE DONE

Find suitable resources for reference developing the schedule interface.

## 3. PROBLEMS ENCOUNTERED

No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

Able to understand the dart programming and perform well.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 7
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Complete the partial part for the schedule module.  
Prepare to code for home interface.

## 2. WORK TO BE DONE

Find suitable resources for reference developing the favourite list.

## 3. PROBLEMS ENCOUNTERED

No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

Able to understand the dart programming and perform well.



\_\_\_\_\_  
Supervisor's signature



\_\_\_\_\_  
Student's signature



# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 8
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Completed the schedule interface.  
Complete the favourite list features.  
Prepare to code for booking interface.

## 2. WORK TO BE DONE

Find suitable resources for reference developing the booking interface.

## 3. PROBLEMS ENCOUNTERED

Validation for appointment date, time, and mode.

## 4. SELF EVALUATION OF THE PROGRESS

The logic and flow of the appointment need time to figure out.

Supervisor's signature

Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 9
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Complete the booking interface.  
Prepare to code for consultation history interface.

## 2. WORK TO BE DONE

Read the documentation for the TensorFlow.  
Find suitable resources for reference developing the consultation history interface.

## 3. PROBLEMS ENCOUNTERED

Difficulty in understanding the TensorFlow.

## 4. SELF EVALUATION OF THE PROGRESS

Required more time to learn the TensorFlow



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 10
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Complete the core features of the mobile application.  
Prepare to train the model using TensorFlow transfer learning.

## 2. WORK TO BE DONE

Read the documentation for the TensorFlow.  
Find suitable resources for reference training the dataset using transfer learning in TensorFlow.

## 3. PROBLEMS ENCOUNTERED

Difficulty in understanding the flow of the train model.

## 4. SELF EVALUATION OF THE PROGRESS

Hard to start training the model.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 11
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Prepare the environment and dataset to train the model.

## 2. WORK TO BE DONE

Find suitable resources for reference training the dataset using transfer learning in TensorFlow.

Refer to the resources online and start to train the model.

## 3. PROBLEMS ENCOUNTERED

The accuracy of the model is not high.

## 4. SELF EVALUATION OF THE PROGRESS

Difficulty in finding the problem of the process in training the model.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 12
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Complete the all the features in the system.

## 2. WORK TO BE DONE

Complete writing the Chapter 4, Chapter 5, Chapter 6 and Chapter 7.

## 3. PROBLEMS ENCOUNTERED

No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

Able to finish the features of the system.

\_\_\_\_\_  
Supervisor's signature

\_\_\_\_\_  
Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3 Year 3	Study week no.: 13
Student Name & ID: Low Chia Hui (20ACB01686)	
Supervisor: Ts Lai Siew Cheng	
Project Title: Facial and Skincare Service System	

## 1. WORK DONE

Finish report Chapter 4, Chapter 5, Chapter 6 and Chapter 7.

## 2. WORK TO BE DONE

Finalizing the parts in the project

## 3. PROBLEMS ENCOUNTERED

No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

Progress is moderate.



Supervisor's signature



Student's signature

## POSTER

The poster features a purple and pink color scheme. At the top, the title 'Facial and Skincare Service System' is written in a white, cursive font. Below the title, there are illustrations of various skincare products like bottles and jars. On the right side, there is a cartoon illustration of a smiling woman with dark hair. The background is decorated with faint, repeating icons of skincare tools and products.

# Facial and Skincare Service System

## INTRODUCTION

- Provide alternative way to conduct consultation session for the customer with a consultant
- Include Artificial Intelligence technology using Deep learning Algorithm for classification
- Allow management process anywhere, any time

## OBJECTIVE

- To provide video call consultation using ZEGOCLOUD video call plugin
- To develop appointment booking with the Flutter calendar
- To implement a skin classification on acne type using TensorFlow lite

## METHODOLOGIES

- Developed using Flutter App
- Dart programming for Mobile app develop
- Firebase for database storage
- Tensor Flow for the skin classification
- Skin classification using EfficientNet Architecture

## WHY THE PROPOSED SYSTEM IN THIS PROJECT IS BETTER THAN EXISTING SYSTEM?

- Alternative consultation – Video call
- Better analysis – Skin classification
- Easier to use – user friendly



Project Developer: Low Chia Hui  
Project Supervisor: Ts Lai Siew Cheng



## PLAGIARISM CHECK RESULT

### Facial and Skincare Service System

#### ORIGINALITY REPORT

8%

SIMILARITY INDEX

7%

INTERNET SOURCES

0%

PUBLICATIONS

6%

STUDENT PAPERS

#### PRIMARY SOURCES

1

Submitted to Universiti Tunku Abdul Rahman

Student Paper

2%

2

[eprints.utar.edu.my](http://eprints.utar.edu.my)

Internet Source

2%

3

[eprints.utm.my](http://eprints.utm.my)

Internet Source

<1%

4

Submitted to Nottingham Trent University

Student Paper

<1%

5

Submitted to Birkbeck College

Student Paper

<1%

6

Submitted to Universiti Teknologi Malaysia

Student Paper

<1%

7

[hdl.handle.net](http://hdl.handle.net)

Internet Source

<1%

8

[www.coursehero.com](http://www.coursehero.com)

Internet Source

<1%

9

Submitted to Greenwich School of Management

Student Paper

<1%



10	Submitted to Georgia State University Student Paper	<1 %
11	Submitted to Southern New Hampshire University - Continuing Education Student Paper	<1 %
12	d-nb.info Internet Source	<1 %
13	Submitted to Kennesaw State University Student Paper	<1 %
14	Submitted to UNITEC Institute of Technology Student Paper	<1 %
15	Submitted to Universiti Tenaga Nasional Student Paper	<1 %
16	www.fcrn.org.uk Internet Source	<1 %
17	Submitted to University of Mauritius Student Paper	<1 %
18	Submitted to University of New South Wales Student Paper	<1 %
19	ntnuopen.ntnu.no Internet Source	<1 %
20	www.partycity.com Internet Source	<1 %
21	digital.library.unt.edu Internet Source	<1 %

		<1 %
22	Submitted to HELP UNIVERSITY Student Paper	<1 %
23	Submitted to Middle East Technical University Student Paper	<1 %
24	www.science.gov Internet Source	<1 %
25	"Proceedings of the 8th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT'18), Vol.1", Springer Science and Business Media LLC, 2020 Publication	<1 %
26	Submitted to Macon State College Student Paper	<1 %
27	Submitted to University of Greenwich Student Paper	<1 %
28	iq.opengenus.org Internet Source	<1 %
29	lumbinifinance.com.np Internet Source	<1 %

<b>Universiti Tunku Abdul Rahman</b>			
<b>Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)</b>			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**

<b>Full Name(s) of Candidate(s)</b>	Low Chia Hui
<b>ID Number(s)</b>	20ACB01686
<b>Programme / Course</b>	Bachelor of Computer Science (Honours)
<b>Title of Final Year Project</b>	Facial and Skincare Service System

<b>Similarity</b>	<b>Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)</b>
<b>Overall similarity index: <u>8</u> %</b>  <b>Similarity by source</b> Internet Sources: <u>7</u> % Publications: <u>0</u> % Student Papers: <u>6</u> %	
<b>Number of individual sources listed of more than 3% similarity: <u>0</u></b>	
<b>Parameters of originality required and limits approved by UTAR are as Follows:</b> (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

**Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.**



\_\_\_\_\_  
Signature of Supervisor

Name: Lai Siew Cheng

Date: 28/4/2023

\_\_\_\_\_  
Signature of Co-Supervisor

Name: \_\_\_\_\_

Date: \_\_\_\_\_



**UNIVERSITI TUNKU ABDUL RAHMAN**

**FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY  
(KAMPAR CAMPUS)**

**CHECKLIST FOR FYP2 THESIS SUBMISSION**

Student Id	20ACB01686
Student Name	Low Chia Hui
Supervisor Name	Ts Lai Siew Cheng

<b>TICK (√)</b>	<b>DOCUMENT ITEMS</b>
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

\*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)

Date: 28/04/2023