# Abstract

Creating a mobile app that satisfies the end-user and the client is a difficult task to accomplish. Moreover, deciding what software to use for development could be challenging since each software has its own benefits. Comprehending a solution to the underlying problems, the project presented in this paper explores the uses of use cases, react native and agile methodology to create a mobile app that meets both client and user needs. The process in deciding the software, research methods and methodology will be discussed in the later sections.

# 1.Introduction

The use of smart phones such as iPhone and android has increased exponentially in the recent years, which is proven by the sales of smartphones to the end users increasing from 296 million smartphone sold in a year in 2010 to 1433 million smartphones sold in a year in 2021 (O'Dea, 2022). The end-user is expected to grow larger in the near future. Due to the accessibility and versatility in smart phones, they are more commonly used than laptops or pc. The vast increase in popularity of smartphones is mostly likely due to the mobile apps which are available for any user’s needs. For example, there is a mobile app that tracks the heart rate of a user to provide health advice which could be essential for any user with a weak heart or user that enjoys working out/running. Although the target audiences for mobile apps are huge, developing a mobile app that meets the user needs are challenging.

In this project I will be developing a mobile app for a nail salon called “Sharon” based in Japan Tokyo, Kita-senju. Nail Salon Sharon is a nail salon owned by Kaoru Ogundiran, in which who is my main client for this project. The nail salon does several different services, such as hand nail gradation, foot nail gradation, etc. The prices of each service are dependent on what the customer orders, but it could range from 20 pounds to 60 pounds.

## Problem Statement

Kaoru Ogundiran an owner of the nail salon Sharon requires an application in English that allows the customer to book appointments. Currently the nail salon uses a third-party website called “Hot Pepper Beauty”, the website allows for viewing menus for the customer, viewing contact details and booking appointment. however, the owner needs an app mobile app which is in English to attract customers from wide area. Furthermore, she wants an app where the owner or the employee of the salon could check profits and customer appointment details easily.

# 2.Related Work

There are several software and database used for mobile app development each with its own benefits. However, deciding what framework is best fitting for the project your building is challenging since there are about dozens of frameworks to choose from, few known frameworks being React Native, Flutter, Swiftic, Ionic and Jetpack Compose. In this section I will be looking into the performance between React Native, Flutter and Jetpack Compose.

## 2.1 React Native

React native is a cross platform framework used in developing mobile applications, it was released in 2015 by Facebook (Wu, 2018). The programming language in which React Native is written in JavaScript however React Native applications are not hybrid or html5 applications, the usage of underlying native interface allows for it to render views and access native hardware such as camera and storage (Wu, 2018). React Native is famous for being open source with a strong community, Facebook developers, individual and even Microsoft and Samsung contribute to developing React Native3. React Native uses JSX which is a special syntax extension of JavaScript to describe how the UI displays. When the application is built the JSX is also compiled. There are two major data models in React, being Props and State, each having its own distinct feature. Props are set externally and is used to customize the component whereas State are set Internally and is used to initialize values.

## 2.2 Flutter

Flutter is also a cross-platform framework used in developing mobile applications, it was released in 2016 by Google (Wu, 2018). Not only can Flutter applications run on Android and IOS, but it also runs on Fuchsia. On the contrast Flutter renders every view component using its own high-performance rendering engine instead of relying on web views like React Native. Every application in Flutter is written with Dart which is a programming language developed and maintained by Google, since Dart was developed as a replacement and successor of JavaScript it implements most of the characteristics of JavaScript’s next standard (ES7). When it comes to data structure Flutter is unique, as it uses Widgets instead of props and state, hence the application is essentially a collection of widgets interacting each other. Widgets contain features such as creating input Text Box, Buttons, etc. one of the crucial advantage Flutter has over React Native is that it allows rapid development with the availability of widgets even for animation and gesture detection. Another interesting fact about Flutter is that it allows for hot reload, meaning it allows for quick update of application state when save button is clicked.

## 2.3 Jetpack Compose

Jetpack Compose is a framework developed by Google for creating native Android applications using Kotlin. The first stable public version was released in July 2021 (Soininen, 2021). Unlike the framework mentioned above, Jetpack Compose does not support cross platform and only supports android platform.

## 2.4 Performance Comparison

There is an study by Wenhao Wu comparing React Native and Flutter which are the two-leading framework in mobile app development. Wenhao compares the two frameworks by writing a simple TV show application in React Native and rewriting it in Flutter. Upon using each framework, he discovered that Flutter and React Native both do an excellent job in performance regarding scrolling where the average fps (frame per second) while scrolling being over 60 fps (Wu, 2018). Although React Native did an excellent job regarding scrolling, when the list it’s scrolling through is dense it’s said to have a significant drop in fps, whereas flutter remained stable. Wenhao had another comparison regarding the speed of input and output system (I/O) of React Native and Flutter, upon the comparison he found out that React Native has an advantage on both the average time and single time consuming (Wu, 2018). The measurement was calculated by calculating the time of an obtained list of epochs from opening a file to successfully writing on a file.

From the comparison above we can conclude that react native is a better framework then Flutter however there are many app development frameworks other than flutter. In a study paper produced by Visa Soininen, the author compares the performance between react native and Jetpack Compose. He developed two applications using React Native and Jetpack Composer, the app had features such as user authentication (login/sign up), bottom navigation bar and item listing. When testing the performance of each application he evaluates the compiling speed and rendering speed. From the comparison, author discovered that React Native does has a better performance with both compiling speed and rendering speed. The initial time taken to bundle the application and the subsequent time after the initial bundle were used for the comparison. After taking an average of 5 results, he got a result of react Native obtained an Initial bundle time and a Subsequent build time of 21.5seconds and 54 milliseconds, whereas Jetpack Compose obtained an initial bundle time and a subsequent build time of 22.5 seconds and 1112 milliseconds (Soininen, 2021). Furthermore, the rendering speed of each framework were obtained by utilising the button components in each framework to render a new object. After taking an average of 5 results, we obtained the 450 milliseconds rendering speed for Jetpack Compose and 371 milliseconds for React Native (Soininen, 2021). We can conclude from the data gained from the study from research papers above, that React Native is the optimal mobile development framework to be used in this project.

# 3.Methodology

Diagram

Description automatically generated

Here is a simplified version of the system diagram. It’s a site map which displays the user flow from login to homepage to logout. There are two different homepages, one for the customer and another for the owner/employee of the nail salon Sharon. The homepage of customers includes 6 different screen page, gallery page, contact page, booking page, profile page, drawer navigator. The homepage of the owner/employee includes the drawer navigator and setting which includes features, which is discussed later. Moreover, the screen pages in both the owner and the customer have a logout feature which navigates the user back to the login screen.

Each state in the system diagram has it’s on features for example contact page has feature which allows the user to send a mail to the owner. To understand what each screen pages do I will be creating a system diagram for pages with features.

## 3.1 Login

Diagram

Description automatically generated

Here is the detailed system flow diagram of Login. The user is first navigated to the login page upon opening the app, in the login page there are options to navigate to the client login page, sign up/register page or authenticate by logging in. Similarly, client login has feature to authenticate by logging in and navigate back to the login page. The features on sign up page is different from client login and login page since it does not require the user to authenticate, however the user cannot register unless the email address the user inputted is verified. The verification is done through mail. When the user is authenticated or a new user is created, if the user is authenticated through the customer login page, the customer gets navigated to the homepage. Whereas if the user is authenticated through the owner/employee login page, the page is navigated to the owner/employee homepage.

## 3.2 Gallery Page

## 3.3 Contact Page

Diagram

Description automatically generated

The user flow of the contact page is simple, it has three features, email us, phone number and location. Upon the button pressed email us directs the user out of the nail salon app into the Gmail app, the user is then able to contact the company, moreover upon pressing the phone number button the user is then given an option to call the company phone number, finally the location button navigates the user to google map with the location of the nail salon already given.

## 3.4 Booking Page

Diagram

Description automatically generated

The flow diagram of the booking page consists of four stages, booking page for the user to select a service from the menu, booking page for the user to select the date and time, booking page for the user to confirm the appointment details. In booking page (menu), the user will need to have a service selected in the menu to proceed to booking page (date). Similarly in booking page (date) the date and time are needed to select to proceed to the booking confirm page. After the details are confirmed, the confirm page navigates the user back to the homepage.

# 5.Project Management

Competitive analysis

In app development, competitive analysis plays a crucial role in the development cycle. Competitive analysis allows the programmer to understand the market, forecast market potential, track competitor pricing and product and many other. For my program I have decided to conduct competitive analysis on two nail salon company, Young LDN and Townhouse nail salon. They are London based nail salon which does other services such as hair and skin.

Young LDN nail salon:

Townhouse nail salon:

# 4.Evaluation

# 5.Conclusion

[SUPR-Qm: A Questionnaire to Measure the Mobile App User Experience - JUX (uxpajournal.org)](https://uxpajournal.org/supr-qm-measure-mobile-ux/)

# Bibliography

O'Dea, S. (2022, November 17). *Smartphones-Statistcs and Facts*. Retrieved from statista: https://www.statista.com/topics/840/smartphones/#topicHeader\_\_wrapper

Soininen, V. (2021). Jetpack Compose vs React Native – Differences in UI Development. *Metropolia University of Applied Science* (p. 32). Metropolia University of Applied Science.

Wu, W. (2018). React Native vs Flutter, cross-platform mobile . (p. 28). Metropolia University of Applied Sciences.