

COACH ME: PERSONAL COACHING MOBILE APPLICATION

BASSIMAN BIN ANUAR

AI190276

BACHELOR OF COMPUTER SCIENCE (SOFTWARE ENGINEERING) WITH
HONOURS

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

ABSTRACT

With number of closing gymnasiums increasing exponentially, it entails the steady decline of unemployment in the fitness industry particularly amongst the fitness coaches or instructors. Thus, personal trainers and coaches do not have a dedicated platform during the pandemic to train students from home due to the COVID-19 Pandemic. Hence, this project aims to design, develop, and test a personal coaching mobile application called Coach Me by using the incremental prototyping as the software methodology to tackle these problems. The mobile application will be built using Flutter, a Dart framework, Node.js runtime environment and MySQL database. The significance of this project is that the mobile application boosts the camaraderie between trainers and their students even with the physical limitation of not being able to enter a gym providing a platform for both parties to connect. Albeit if the pandemic has reached its endpoint, the mobile application will not lose its traction and continue to thrive as it will drive coaches and students who are not able to attend a physical class in using the mobile application as a platform to professionally connect and re-watch video lessons.

CONTENTS

TITLE	i
ABSTRACT	ii
CONTENTS	iii
LIST OF APPENDIX	Error! Bookmark not defined.
1 CHAPTER 1	1
1.1 Project Background	1
1.2 Problem Statement	3
1.3 Objective	4
1.4 Scope	4
1.5 Expected Result	7
1.6 Project Significance	8
1.7 Report Organization	8
2 CHAPTER 2	10
2.1 Introduction	10
2.2 Case Study: Fitness and Coaching at Ultimate MMA Fitness & Self Defense	10
2.3 Mobile Application Development	12
2.3.1 Browser Access Mobile Applications	12
2.3.2 Native Mobile Applications	12
2.3.3 Hybrid Web Mobile Applications	12
2.3.4 Hybrid Mixed Mobile Applications	13
2.3.5 Mobile Application Evolution	13
2.4 Existing System Comparison	14
2.4.1 Openfit 14	
2.4.2 Nike Training Club	16

2.4.3	Peloton	18
2.4.4	Proposed Mobile Application	19
2.5	Comparison Between Existing Mobile Applications and Proposed Mobile Application	21
2.6	Chapter Summary	22
CHAPTER 3 24		
3.1	Introduction	24
3.2	Prototyping Model	24
3.2.1	Planning Phase	25
3.2.2	Analysis Phase	26
3.2.3	Design Phase	26
3.2.4	Prototyping Phase	27
3.2.5	User Evaluation Phase	27
3.2.6	Implementation Phase	27
3.2.7	Testing Phase	28
3.3	System Development Workflow	28
3.4	Chapter Summary	30
4	CHAPTER 4	31
4.1	System Requirement Analysis	31
4.1.1	Functional and Non-Functional Requirements	32
4.1.2	User Requirement Analysis	35
4.1.3	System Hardware and Software Requirements	36
4.1.4	UML Diagram	36
4.1.5	Use Case Diagram	37
4.1.6	Use Case Specification	37
4.1.7	Class Diagram	51
4.1.8	To-Be Model	52
4.1.9	Requirement Traceability Matrix	53
4.2	Design	55
4.2.1	System Architecture	55
4.2.2	System User Interface Design	56
4.2.3	System Database Design	71
4.2.4	Data Dictionary	72

4.3 Chapter Summary	77
REFERENCES	78
Appendix A	82

CHAPTER 1

INTRODUCTION

1.1 Project Background

The year of 2020 has left many of us in a dismal tone and will be remembered as one of the worst global health crises in modern times going down in history, the COVID-19 pandemic (Karin and Jesper, 2021). According to a Worldwide Trade Group for the health and fitness sector, the fitness industry employed 3 million part-time and full-time personnel prior to the pandemic. However, with the steady incline of people being infected, more than 480, 000 jobs are getting destroyed per month all around the world. While other small companies might pivot to establish their own online commerce and takeout orders, health clubs and their staffs of personal trainers as well as fitness instructors has no such option. A professional fitness coach or instructor can be defined as a person who aid participants in exercise programs, evaluate various fitness components, instruct exercises to improve the participants health, and help participants with specific conditions (Howley et al., 2007). With number of closing gymnasiums increasing exponentially, it entails the steady decline of unemployment in the fitness industry particularly amongst the fitness coaches or instructors.

Prior to the pandemic, coaches will teach a myriad of classes in their local gym depending on their skills and certifications. These classes typically last anywhere between an hour to 3 hours. Any people who are interested in joining their classes are then enrolled with a certain amount of fee in order to become a student of the class. A class is then

divided into lessons where multiple students enrolled in the class will be coached simultaneously. For instance, the first lesson of a Yoga Class is warming up. Moreover, private sessions are also held in which the students are coached and given consultations one-to-one by the instructors. Importantly, the objective of their fitness programs aims to provide a social situation in which participants, supervised by licensed group fitness instructors, benefits socially and emotionally (Andreasson et al., 2015). Case in point, physical classes is held within a local gym whereby students are coached by a certified or a skillful instructor.

Ergo, the aftermath forces most fitness coaches to resort to online classes by utilizing digital platforms such as WhatsApp, Zoom and Skype. Physical classes are substituted with online conference calls, while private sessions between a coach and a student are strictly forbidden, they are replaced with private chats which constricts the effectiveness of a consultation session as the coach cannot provide an accurate feedback and advice due the limited functionality of a private chat. The up-bringing atmosphere of conducting a class within a gym where music, fitness partners are no longer accessible. The effect of social distancing allotment on overall physical activity is an important stimulus of health, especially if social distancing is prolonged (Geoffrey, 2020). Thus, it implies that fitness centers will not be opening anytime soon.

Henceforth, a Personal Coaching Mobile Application (Coach Me) is proposed to connect fitness coaches and their students in a digital platform further cementing the effectiveness of a fitness class. The overview of the app is that personal coaches can upload, store their prerecorded lessons in a class where students can enroll with a certain amount of fee based on the coach to then watch the prerecorded lessons and follow along with their coaches. A private one-to-one session between a coach and a student can also be conducted. Students can then view their progress in class and consult with their coach to help clarify misunderstanding regarding a specific topic or asking for a few tips and tricks which they may not know due to their impeding progress. Albeit if the pandemic has reached its endpoint, the mobile application will not lose its traction and continue to thrive as it will drive coaches and students who are not able to attend a physical class in using the mobile application as a platform to professionally connect and re-watch video lessons.

1.2 Problem Statement

Many parts of life, including physical activities, has been affected as a result of lockdowns in workplaces, schools, and general social life, which has become a routine to battle the disease (Laredo et al., 2021). As of now, personal trainers and coaches do not have a dedicated platform during the pandemic to train students from home. Prior to the pandemic, coaches will teach in their own local gym within a class of a few people. However, with our current country's Health Standard Operating Procedure, gyms are now closed and shut for good until the number of infected people are within a certain figure. This inevitably leads to the instructor's unemployment, which in turn leads to the fall of traction within the fitness community.

Certainly, the current alternative to these hindrances of not being able to conduct a physical class is by digitalizing the classes by using WhatsApp or Skype, but there are many limitations with regard to what personal trainer and coaches can do over third-party apps. One of the many standard operating procedures that governs the fitness industry is to eradicate fitness establishments such the gymnasium. Among the actions taken, the government determines that people should stay at home. Hence, ceasing their social life, and the routine of physical activities, entailing to an upward trend in physical inactivity (Pinho et al., 2020). These constrictions may demotivate the student from their progress or some students might just unenroll from the class. Moreover, the obvious drawback of using these third-party apps in order to conduct a professional class is poor communication over text with no constant feedback.

Reaching coaches that are internationally certified has always been challenging, especially if the person seeks instructors that are specific to their own niche. International experiences with regards to athletic training hard to identify and limited (Wright, 2019). Given a scenario whereby a person wants to find a coach that is experienced and knowledgeable in inline hockey which has low popularity in Malaysia, he would have done searches for them manually through mutual contacts or many thorough Google searches in order to find detailed information regarding a specific coach.

This further emphasizes that it is near impossible for professional coaches to conduct their classes and monitor the progress of their students online without a dedicated platform which removes the professionalism out of the equation.

1.3 Objective

This project is proposed to achieve the following objectives:

- i. To design a personal coach mobile app using object-oriented approach.
- ii. To develop a personal coach mobile application
- iii. To test the developed mobile application.

1.4 Scope

The scope of this project is broken down into several chunks. The investigated case study in this project is how fitness coaches conduct their classes. The stakeholder in charge of overseeing the investigation is the manager and a coach of Ultimate MMA Fitness & Self Defense, Mr. Alvin Yap.

Secondly, the design of the mobile application leans towards coaches and their students. The proposed mobile application is targeted for everyone that wants to stay in shape by hiring a personal trainer in connecting coaches to their students by joining classes with prerecorded video lessons by their coaches where students can watch and follow at their own pace. The expected user of the mobile application is the administrator, coaches and students. The system administrator consists of the developer or a team hired to specifically manage and maintain the mobile application. The common user of this mobile application is anyone who seeks out exercise programs that the coaches provide. Enrolled students(public user) can also create one-to-one private consultation sessions between their coaches (public user). The application administrator's responsibilities are to affirm account deletion, issuing refunds, and create monthly reports for number of engaged students, highest rated coaches for each categories and number of downloads per day.

Next, there are five primary modules which make up the backbone of the mobile application, namely, Sign Up and Log In module, Coaching module, Learning module,

Scheduling module and Administrator module which are then broken down into multiple submodules.

The Sign Up and Log In module manages the user's information and is responsible for routing the users to dedicated pages when authenticated. This module, then subdivides into Signing Up submodule, Logging In submodule and Account Recovery submodule. The description of these submodules is titular. For instance, a student will need to sign up with the necessary credentials and then log in to be authenticated. The student will then be brought into the Student's Homepage where they can enroll into a specific class according to their interests.

Next, the Coaching module is broken into View Account, View Students, Create and Manage Classes, Create and Manage Video Lessons within a class and Join a Private One-to-one Session submodules. The View Account submodule shows the user (coach) their personal information as well as certification and skills, view garnered income and banking credentials. The View Students submodule is responsible for showing the user (coach) the simplified (expandable) view of enrolled students under the coach. The Create Classes submodule generates titling classes based on the form filled. Then, the user (coach) can upload their prerecorded video lessons using the Create Video Lessons submodule. Join a Private One-to-one Session submodule connects the coach to a private chat where they can consult their student over a specific matter appointed by their student. The price of classes is based on the coaches. These fees can be free, or priced. In which if the coaches choose the classes to be free, it will be free forever as opposed to priced classes which cannot change the price of the class once it has been made public. For example, an Introduction to Weight Loss class is free, whereas a Weight Loss Program 1 class will cost the students RM 60. All payments of class should only be made one time only as the payment submodule is for the coaches to receive the payments.

Moreover, the Learning module can be broken down into 5 submodules which are View Account, View Coaches, Join Classes, Watch Video Lessons, Create a Private One-to-one Session (Coach me). The View Account submodule shows the basic user (student) information such as name, email, phone number, date of birth, height, weight, payment history while the View Coaches submodule allows the user (student) to view the coach's profile which includes their certification and skills. By using the Join Classes submodule,

the user(student) can then enroll in a specific class with a certain sum of fee of which the price is based on the coach during class creation. Enrolled students are then free to watch video lessons in the class by Using the Watch Video Lessons submodule. The Create a Private One-to-one Session connects the student to their coach in which topics such as why their progress is impeded are then discussed. The payment submodule is made so that the students can pay based on the coach's fee.

Next, is the Scheduling module, whereby the coaches can manage their schedule in regard to the private-one-to-one coaching session. This module can be view by the students when they want to request for the private session. It provides necessary information such as the time and date of when a session is open for the students to pick form. This module gives the coaches time to plan ahead and to better manage their private sessions as the sessions made by other students will not clash with each other.

The Administrator module is broken down into 3 submodules namely, Account deletion, Refund Issuing, and Reporting. It manages user's (student and coach) account whether it is deleting an account, approve a refund issue for specific payment and create monthly reports for number of engaged students, highest rated coaches for each categories and number of downloads per day. Table 1.1 shows the system module and its function of the Coach Me mobile application.

Table 1.1: Function and User for each System Module

System Module	Function	User
Sign Up and Log In module	This module is responsible for routing the users to dedicated pages when authenticated.	Administrator, Coach and Student
Coaching module	<ol style="list-style-type: none"> 1. View Account. 2. View Students. 3. Create and manage (update and delete) Classes. 4. Create and manage (update and delete) video lessons within a class. 5. Join a private one-to-one session. 6. Receive payments. 	Coach

Learning module	<ol style="list-style-type: none"> 1. View account. 2. View coaches. 3. Join or unenroll from classes. 4. Watch video lessons. 5. Create a private one-to-one session(coach me). 6. Make payments. 	Student
Scheduling module	<ol style="list-style-type: none"> 1. coaches can manage their schedule in regard to the private-one-to-one coaching session. 2. Students can choose the date and time of an empty session. 	Coach, Student
Administrator module	<ol style="list-style-type: none"> 1. Account deletion. 2. Refund issuing. 3. Report (Create Monthly report for number of engaged students, highest rated coaches for each categories and number of downloads per day). 	Administrator

1.5 Expected Result

At the end of the project, the mobile application for personal coaches is fully developed and used by both student and coaches whilst being managed by administrators. The mobile application is able to sign up new users and log in existing users as well as recovering an account upon request and deleting them. Moreover, the proposed system is expected that the coach user is able to view personal and banking details within the View account component. The mobile application is to ensure that the coach user is able to create classes by filling out necessary forms and then upload prerecorded video as lessons in the class. The coach user is also expected to be able to view student's personal information and progress. Next, the proposed mobile application is to ensure that the student user is able to view personal details regarding the student user's account. The student user should also

be expected to enroll into a class and watch a prerecorded video lesson by the coach. A one-to-one private chat session between the student and coach should also be ensured by the proposed mobile application. The administrator will have the necessary authority to access the system to delete an account, issue a refund and also monitor data analytics. Hence, the proposed system is able to connect professional coaches to their students in a digital platform that excels in functionality.

1.6 Project Significance

The significance of this project is that the mobile application boosts the camaraderie between trainers and their students even with the physical limitation of not being able to enter a gym. Next, by deploying the mobile application into the global market, more people are interested in getting shape even during lockdown or strict Standard Operation Procedures - they can be coached within the comfort of their own home by a personal trainer, with a skill set extending the primary care of physicians to a range of settings (Mulkey, 2021). Lastly, this project will reduce the number of unemployed personal trainers as the platform provides a firm ground for the persona coacher to work on even with the eradications of a physical establishment.

1.7 Report Organization

This report is organized in the order of as follows:- Chapter 1 conveys the introduction of the project which consists of the project background, problem statement, objective, scope expected result, project significance and the chapter summary. Next, Chapter 2 primarily discusses matters regarding the study of related concepts which can be abridged as the literature review. It is expounded into several sub chapter which encompasses the domain of the study, existing systems, and the correlation between these existing systems. Chapter 3 goes in depth into the methodology utilized for the project. The methodology then is broken down and addressed phase by phase. Chapter 4 provisions the analysis and design of the project. It describes the system's requirement analysis, use case scenarios, and depicts the sequence diagram, requirement traceability matrix, entity relationship

diagram, interface design and schema table. Furthermore, Chapter 5 outlines the implementation and testing of the mobile application. It provides the necessary code segments and the interfaces and the test cases. Lastly, Chapter 6 concludes the report as a whole and summarizes the system advantage, disadvantages, and the further improvements to be done in the near future.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses the literature review that is done during the development of the proposed mobile application. The literature review is conducted to further study and scrutinize several existing mobile applications that are related to the development of the proposed mobile application. The domain of the study is fitness and coaching. Section 2.1 discusses matters regarding the overview of fitness and coaching process and its data management. The overview of mobile application technology will be further discussed in Section 2.2. Section 2.3 will then expound the existing mobile applications within the fitness and coaching domain. Three similar types of existing mobile applications are then further studied and compared alongside with the proposed mobile application. The last section within this chapter, Section 2.4 summarizes the chapter as a whole and highlights the key points within this chapter.

2.2 Case Study: Fitness and Coaching at Ultimate MMA Fitness & Self Defense

Prior to the epidemic, the fitness industry employed 3 million part-time and full-time workers, according to a Worldwide Trade Group for the health and fitness sector. However, as the number of those affected continues to rise, more than 480, 000 jobs are lost each month throughout the world. Unlike other small businesses, health clubs and their staff of personal trainers and fitness instructors do not have the option of establishing

their own web commerce and takeaway orders. With the number of gyms closing at an exponential rate, the fitness sector is experiencing a continuous reduction in employment, particularly among fitness trainers and instructors.

Simply described, a fitness trainer or personal coach is someone who manages an exercise program. Coaches will teach a variety of classes at their local gym prior to the outbreak, based on their talents and credentials. These lessons usually run between an hour and three hours. Anyone interested in enrolling in their lessons is required to pay a particular cost to become a student of the class. A class is then separated into lessons, with numerous pupils in the class being instructed at the same time.

To further understand the coaching process, a case study within Ultimate MMA Fitness & Self Defense which is a fitness club and institute teaching myriad of programs such as self-defense, weight loss and MMA. The person who oversees the investigation was Mr. Alvin Yap, manager and coach of Ultimate MMA Fitness & Self Defense. The result of the case study is depicted as a model shown in Figure 2.1.

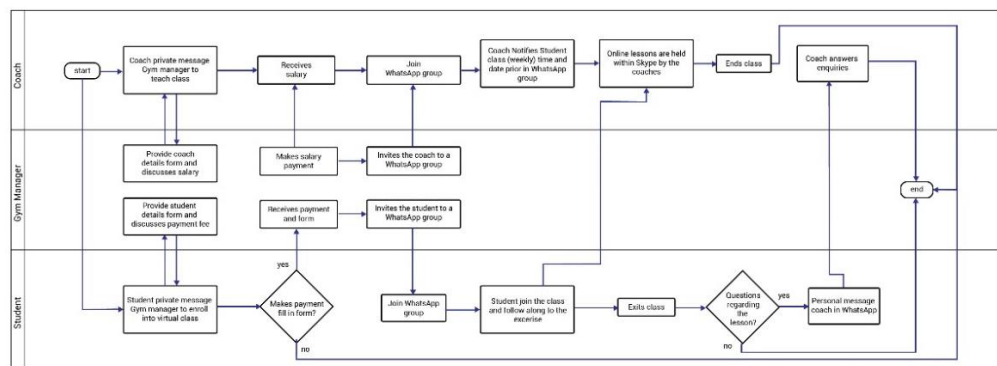


Figure 2.1: As-Is model of Ultimate MMA Fitness & Self Defense

The current problem faced by Ultimate MMA Fitness & Self Defense as stated by Mr. Alvin Yap was that using two mobile applications to digitalize class is inefficient. Moreover, the students escalate their complaints regarding missing a class and not being able to revise or watch then again. This leads to the loss of traction and motivation within the students to join a lesson and affects the coach's workload. Thus, a dedicated mobile application is needed to overcome the said problems.

2.3 Mobile Application Development

A type of application software that runs on a mobile device is a mobile application, or a common shorthand for it would be called apps. Due to the recent decades of technological advancement, it has resulted more technologies being innovated and reinvented by the wheel.

According to Phongtraychack and Dolgaya (2018), there are four types of mobile application categorized into their services, namely, native mobile application, hybrid web mobile application, hybrid mixed mobile application and also browser access mobile application. These four types of mobile application services will be further discussed within the next sub-chapters.

2.3.1 Browser Access Mobile Applications

Any application which is accessed through a native browser for example Mozilla Firefox is a browser access mobile application (Charland and Leroux, 2011). [Www.google.com](http://www.google.com) and www.man-anr.com are the examples of bona fide browser access mobile application

2.3.2 Native Mobile Applications

A mobile application that is built and installed specifically on a mobile operating system is called a native mobile application. Apple's Clock application is a perfect example of a native mobile application because it is specifically built for iOS and cannot be installed in any other mobile application.

2.3.3 Hybrid Web Mobile Applications

A hybrid web mobile application is a mobile application that requires internet connection, can be accessed by a web browser and is built by using web technologies for instance, Html, Cascading Style Sheet and JavaScript which uses the browser's engine for example

Google V8 engine that executes and renders those content (Panhale, 2016). Some good instances of a hybrid mobile application is social networking applications such as Facebook and Twitter.

2.3.4 Hybrid Mixed Mobile Applications

Furthermore, a hybrid mixed mobile application can be defined as a mobile application that may or may not use internet connections (Hoekman, 2010). A sound example of the said mobile application is offline and online games, Samsung Health which keeps record of the user's health data.

2.3.5 Mobile Application Evolution

Case in point, mobile applications have evolved from a primitive and verbose architecture to a more complex yet optimised architecture. According to Lehman and Belady (1985), there laws governing software evolution but only 2 is contextually important which are, continuing change, increasing complexity. Both laws are interwoven such that any mobile application even tough useful, will continue to change and get more complex over time. Albeit these laws can be considered as dilapidated, it still being applied within the current predicament - General mobile applications given the example of Facebook, is proof that the said laws still apply in which its technology and user interface continues to change, and its data processing pipeline gets more complicated over new releases.

Same laws apply even to mobile applications specifically within the health and fitness niche. For example, the earlier versions of Samsung Health do not have metrics tracking and it has recently implemented which make the mobile application more complicated and change overtime (Tudor-Locke et. at., 2002). The process of adding features and resolving bugs which each new release is key to an exemplary mobile application. In the next sub-chapter, the features of existing systems will be compared and analysed.

2.4 Existing System Comparison

In order to form a basic guideline to develop the application, the study of existing mobile applications is key to understanding the architecture of other mobile applications. The study will primarily focus on the functionalities and features provided by each of the three existing mobile applications. These 3 mobile applications to be discussed are namely Openfit, Nike Training Club, Peloton. The proposed mobile application will also be explained in this section.

2.4.1 Openfit

Openfit was voted as one of the best personal coaching mobile application in 2021. This is partly due to the COVID-19 pandemic outbreak triggering the number of travel limitations at an all-time high (Chinazzi et al., 2020). At its core, Openfit is a live coaching based mobile application which does not need any form of payments to access. This means that Openfit stream live classes with a live trainer. The mobile application runs on the Android Operating System as well as iOS. The look and feel of the application are smooth and aesthetically pleasing as it utilizes material design for instance cards and top navigation bar as the basis in user interface elements design. As depicted by New York Post (2021), figure 2.1 shows the user interface for Openfit.

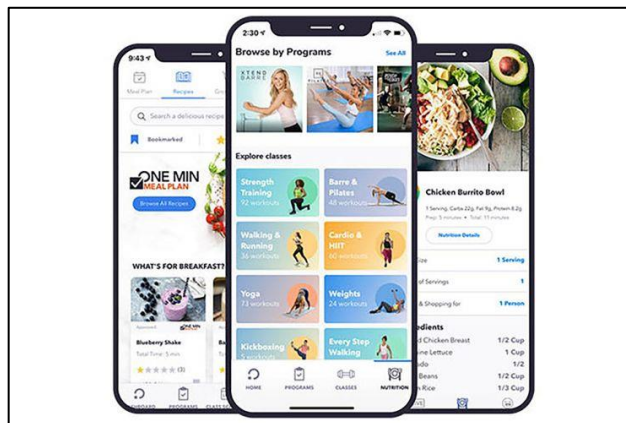


Figure 2.1: Openfit User Interfaces (New York Post, 2021).

To use the Openfit, the user will first need to sign up with the necessary credentials such as their name, email, phone number and age, height and weight. Then, the user will log in and use the mobile application. The main feature of Openfit is streaming live classes with a certified trainer. On the surface, the basis of feature looks pragmatic, but the idea of having a certified personal coach in a live stream where the coach needs to supervise many students itself is flawed. This is due to the limited feedback or supervision at a given time making the live stream ineffective for both student and coach. An example of such problem is as such that in a live stream of 50 students, only one coach is there to supervise all 50 students though the small resolution of a mobile phone or a tablet. However, within a class of less than 5 students, the learning process then becomes much more effective as the coaches can then focus and supervise each student individually.

Next, besides the major feature of the mobile application, it also has additional features namely personalized nutrition program, daily meal plans and workout programs. A few instances of the workout programs include 600 Secs – a high intensity and efficient workout within 10 minute and Xtend Barre which helps to increase muscle gain and burn calories with a little bit of cardio, ballet and Pilates. The customized meal plans help the students to arrange their meals within a certain number of calories by providing them prewritten recipes such as Blueberry Shake and French Toast with Berries. The mobile application focuses heavily on metabolic fitness and nutritional based diets.

A few of the advantages of a mobile application that relies on live stream classes are that all interactions within the class are done in real time, which provides instant feedback. Thus, increasing the efficiency of the class. Openfit is also an all-in-one workout mobile application which helps students to gain muscle and lose weight.

The drawback of Openfit is that all interactions with the coach are contained within a live stream as it does not have a personal coaching session where students can personally interact with the coach themselves. Moreover, since the coach are hired by Openfit themselves, it reduces the authenticity of the class environment. It also does not have a personal coaching schedule feature which constricts the primary task of being able to interact with online personal coaches.

In a nutshell, the disadvantages of Openfit outshines the minute number of advantages it has. The idea of live streaming classes where there are many users that needs to be proactive all the time in ineffective and adds more stress and load to the person teaching.

2.4.2 Nike Training Club

With the cutting-edge technological advancements of sensors and mobile screen, users can receive coaching, motivate and track their exercises (Wang et al, 2021). Nike Training Club is a personal coaching and workout mobile application. It runs on both Android Operating System and iOS. The premise of this mobile application is that it guides students in exercising by providing them with catered programs, and guidance from trainers, athletes, and wellness experts. The mobile application adopts a flat and minimalistic user interface design. As depicted by Lee (2021), figure 2.2 shows the user interface for Nike Training Club.

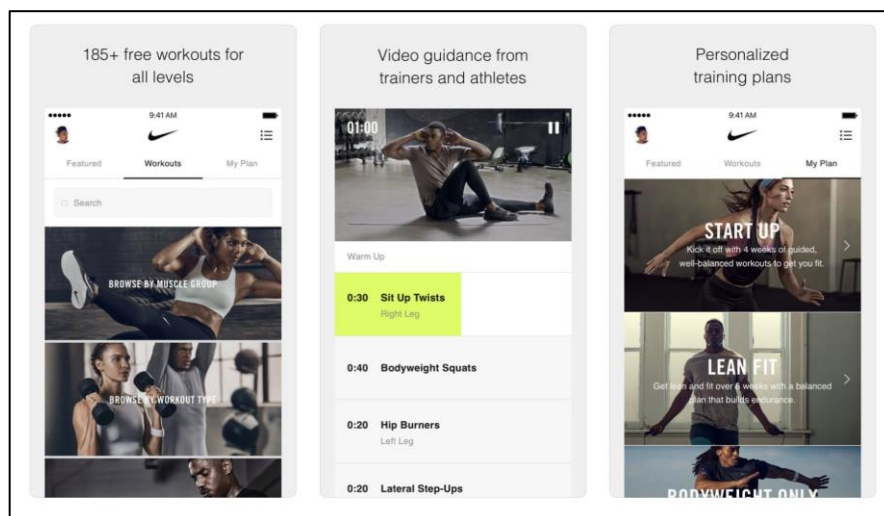


Figure 2.2: Nike Training Club User Interfaces (Lee, 2021).

Just like any other health and coaching app, the user must first sign in and provide their credentials. These credentials include the user's information such their name, email address, phone number which is optional and address. Nike Training Club provides video lessons from their array of coaches. This means that for a personal coach to make use of the mobile application, they first would need to contact Nike Training Club. Thus, there is no option for a personal coach that is not under Nike to sign up.

The additional features that Nike Training Club provides are as such: - Workout programs for various level of athletes or students, nutritional wellness and motivational feed. The examples of workout programs are high intensity training workout which strengthens the core, gain muscle mass and increased flexibility. The Nike Training Club TV provides easy to follow video recipes and guided mediation. Latest wellness and workout content is provided within the motivational feed section. It broadcast the most up to date tips and guidance from Nike's certified trainers and the features are made free of any pricing.

With a mobile application that solely anchors its features to the Nike brand itself, it provides the users a sense of affirmation regarding the quality of content and certification of trainers. Moreover, it works alongside with Nike Run Club mobile application to track daily runs and provide meal suggestions based on the calories lost.

Despite the mobile application being an exemplary ecosystem of what a fitness application should be, it does not embody a full-fledged mobile application for personal coaches. Due to the controlled array of coaches from Nike, the niche of the workout program is significantly smaller. Sports such as Badminton and Tennis could not be found anywhere in the mobile application as it is made only for metabolic workouts and nutritional diets. In addition to the minute choice of coaches, it also does not provide a scheduling feature whereby the students or athletes can arrange a personal session with the coaches. This significantly reduces the effectiveness of a personal coaching mobile application.

In short, the limited ecosystem that Nike Training Club brings to the table does not justify personal coaches as they do not have the option to use the mobile application. Instead of satisfying both parties, the mobile application chooses to only focus on the students or athletes thus hindering the full potential of the mobile application.

2.4.3 Peloton

The realization of not being about to exercise within a gymnasium due to the pandemic has led people to think of new alternatives (Andreucci, 2020). Which explains why the number of downloads for a fitness and coaching mobile application called Peloton has skyrocketed. Peloton is a mobile application that runs on both iOS and Android Operating System. It requires the user to sign up and subscribe to its membership which currently cost RM 54.11 per month billed to their card. A 30-day trial period is given for new users to try and experience the mobile application. Peloton has a tactile user interface design with blue being its primary colour of choice. As depicted by McAuliffe (2021), figure 2.3 shows the user interface of Peloton.

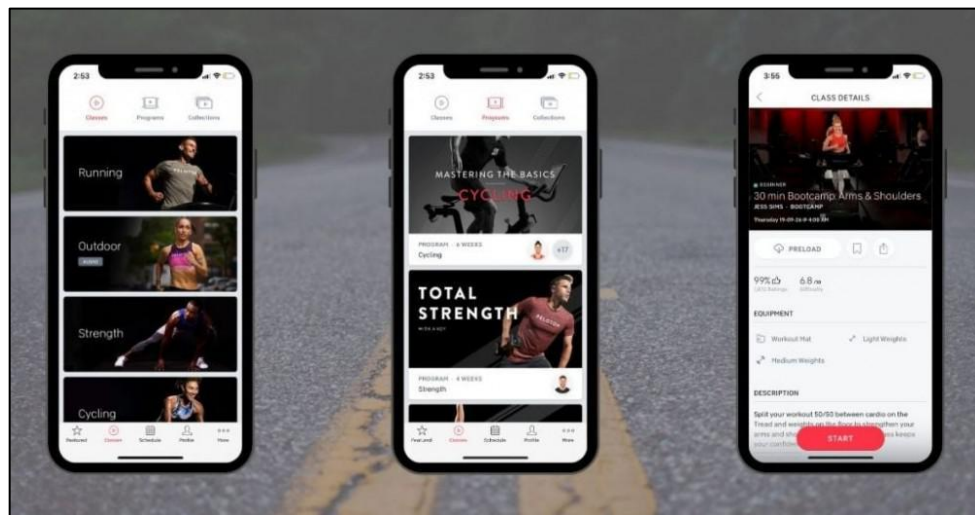


Figure 2.3: Peloton User interface (McAuliffe 2021).

The key feature Peloton provide is that of a diverse range of live streamed lessons. Such lessons include yoga, high intensity interval training, indoor cycling, treadmill running, stretching, walking, meditation and cardio workout. These classes are then

coached by their team of certified instructors and their live stream classes are based in their London and New York studios. Pre-recorded lessons are also available for the user to watch and follow along.

Being able to play music while following an exercise program, tracking active days of workout within a user's profile and compatible with android TV or Chromecast is the additional features of Peloton. The mobile application provides a library within thousands of songs and can curate a playlist based on classes and favourite artist. Peloton also supports the use of a Bluetooth heart rate monitor to view a user's metrics when in a live stream class.

The advantages of using the mobile application are that it provides a 30-day free trial for new users. Moreover, it has a live and on-demand classes which is coached by their certified trainer. Pre-recorded video lessons that stretch across a myriad of workout categories are provided. It also has a positive and devoted community which motivates the user to excel in their workout. Peloton maximize the use of a sibling platform such as Android TV and Chromecast.

Despite the large amount of benefit, there are still some drawbacks within the mobile application. Being a cross platform mobile application, some of the features that are in one platform may not exist in another. Such instances are that of on the Android platform lack workout programs compared to the iOS platform. Moreover, users cannot schedule a personal session within a coach which decreases the usability as a fitness and coaching mobile application. There is also no option to view the full credential of coaches or trainers.

To give a brief rundown of Peloton, it is a freemium based fitness and coaching mobile application that provides its users a wide range of exercise programs and live-streamed classes to choose from. Coaches that are Peloton certified are teaches such class form their London or New York based studios.

2.4.4 Proposed Mobile Application

The design of the mobile application leans towards coaches and their students. The proposed mobile application is targeted for everyone that wants to stay in shape by hiring

a personal trainer in connecting coaches to their students by joining classes with pre-recorded video lessons by their coaches where students can watch and follow at their own pace. There are four primary features which make up the backbone of the mobile application, namely, Sign Up and Log In account, coaching feature, learning from pre-recorded lessons and personal session scheduling.

When users are authorised, the Sign Up and Log In function controls their information and directs them to specific pages. For example, to be authorised, a student must first sign up with the required credentials and then log in. After that, the student will be sent to the Student's Homepage, where they can enrol in a specific class based on their interests.

View accounts, view students, build and manage courses, Create and Manage Video Lessons inside a class, and Join a Private One-to-One Session are the next features in the Coaching section. The View Account function displays the coach's personal information, certifications, and talents, as well as their earned revenue and banking credentials. The view students function oversees in providing the coach with a streamlined and expanded view of all registered students under their supervision and the coach can upload their pre-recorded video lessons. The Join a Private One-on-One Session feature links the coach to a private chat where they may counsel their student on a topic of their choosing. The cost of lessons is determined by the coaches. These fees may be free or chargeable. In which case, if the coaches chose to make the lessons free, they will remain free indefinitely, as opposed to paid classes, which cannot modify their pricing after they have been made public. An Introduction to Weight Loss class, for example, is free, but a Weight Loss Program 1 session costs RM 60. Because the payment option is for the coaches to receive the fees, all class payments should be paid only once.

View Account, View Coaches, Join Classes, Watch Video Lessons, and Create a Private One-to-One Session are the five features of the Learning function. View Account displays basic student information such as name, email, phone number, date of birth, height, weight, and payment history, whereas View Coaches displays the coach's profile, which includes their certification and expertise. The student may then use the Join Classes tool to join a specific class for a set cost, which is determined by the coach during class formation. Enrolled students can then use the Watch Video Lessons option to watch video

lessons in class. The Create a Private One-on-One Session feature links the student with their coach, allowing them to address issues such as why their development is being slowed. The payment option is designed to allow students to pay according to the coach's charge.

Finally, there's the Scheduling function, which allows instructors to organise their schedules for private one-on-one coaching sessions. When students wish to request a private session, they can look at this option. It gives important details such as the time and date when a session is available for students to choose from. This feature allows coaches to better organise and manage their private sessions because other students' sessions will not conflict with their own.

2.5 Comparison Between Existing Mobile Applications and Proposed Mobile Application

Although the Openfit, Nike Training Club, Peloton and Coach me have some similarities such as all mobile applications have the same log-in and sign-up features and multi-platform, the key differences are obvious.

Class scheduling is not available for all three of the existing mobile application whereas the proposed mobile application has a private one-to-one coaching session. The process of conducting a personal -one-to-tone coaching session starts with the student user picking an available time and date which then the coach user chooses to accept or decline.

Moreover, both Openfit Nike Training Club and Peloton offers the coaching to be done by their own hired trainers whereas the proposed mobile application offers coaches from any background to sign up. Table 2.2 provides a concise rundown of the key similarities and differences extracted from the said mobile applications. Thus, it is crystal clear that the proposed mobile application needs to be developed as it includes all the necessary features within a personal coaching application.

Table 2.2: Comparison Between Existing Mobile Applications and Proposed Mobile Application.

<div>Application</div> <div>Features</div>	Openfit	Nike Training Club	Peloton	Coach Me
Platform	iOS, Android, Web	iOS, Android	iOS, Android, Web.	
Log in and Sign up	Users are required to sign up and log in			
Class Scheduling	No class scheduling is available.			A private one to one session or a class can be conducted by the coach.
Pre-recorded classes	Held by the company’s hired personal coacher			A personal coach can a apply to be in the application
Personal one to one coaching	Not available			Users can arrange a personal one-to-one session by choosing an open date and time from the coach’s schedule.
Coach Background	Student cannot view or check the background and certification of the personal coaches.			Students can view the coach’s certification and background within minute detail.
Workout Plans	Goal focused nutrition plan.	Fitness based workout Plan	No workout plan available.	

2.6 Chapter Summary

The aim of this literature review is to investigate and evaluate many current mobile apps that are relevant to the creation of the proposed mobile application. The study's topic falls under the category of fitness and coaching. The chapter introduction is written in Section

2.1. In Section 2.2, the conducted case study within Ultimate MMA Fitness & Self Defense is discussed and an as-is model is produced as the result of the case study. In Section 2.3, the overview of mobile application technologies will be expanded upon. The existing mobile applications in the fitness and coaching arena will next be discussed in Section 2.4. Openfit, Nike Training Club and Peloton are further studied and compared alongside with the proposed mobile application. Although that the existing mobile application may have some similarities to the proposed mobile application, it is found that the existing applications lack many features such as no class scheduling. Range of coaches are also limited with no personal coaching available and students cannot view the coach's certification and background.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter explain the use of prototype model in this project and the activities that had been carried out in each phase.

3.2 Prototyping Model

A software development methodology can be put simply as a way of managing projects related with software development (Young, 2017). It is a structured and uniform process when approaching a specific project. However, each model will face their own respective problems.

The Prototyping Model more specifically, Incremental Prototyping is chosen as the crux methodology in order to develop the mobile application. Prototyping, here, refers to where a model is produced in advance, having all the necessary features of a final product (Guida et al., 2013). In the model, the final product is subdivided into various small prototypes, and each is then developed individually. Once the user is satisfied with iteration of each prototype, the different prototypes are then merged to create a single product. The prototyping model is helpful in reducing the feedback time between the user and the application development process.

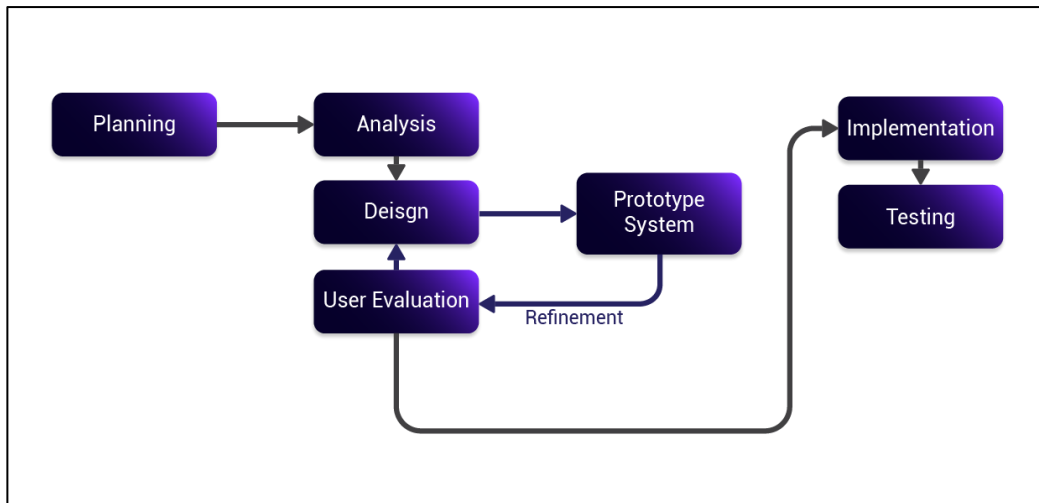


Figure 3.1: Prototyping Model Diagram (Coronel and Morris, 2016)

As depicted in Figure 3.1 based on the diagram drawn by (Coronel and Morris, 2016), there are 7 phases in total namely, planning, analysis, design, user evaluation, implementation and testing. The goal of using a prototyping methodology is to explore, experiment and evolve. A prototyping development approach aids in building and refine, a product to meet end-user expectations (Nacheva, 2017). Thus, the process of iterating the design, system prototyping, and user evaluation is key to assure user satisfaction. The prototyping process encourages the methodical development of applications by breaking down complex problems into several complete yet simpler parts (Besard et al.,2019).

3.2.1 Planning Phase

The first step in an incremental prototyping model is the Planning phase. During this phase the ideas are proposed, and the timeframe of the project system requirements are defined in detail. This phase determines the scope and the goal of the project. In order to depict the project schedule in a graphical form, a Gantt chart is utilized to show the start and finish dates of a project element such as its dependencies, resources, and their respective activities. The significance of this phase is to provide an overview of the proposed system requirements.

3.2.2 Analysis Phase

Throughout this phase, the requirements of the system are defined in detail such as functional and non-functional requirements of the system. Upon succeeding the Planning phase, the required data and analysis process are performed to dictate the requirement of the system. The users of the system are also interviewed to know what their expectation from the system is. This phase is crucial in designing and developing the prototype in the next phase as it will determine the overall look and feel of the prototype. During this phase, the user requirements were gathered from the stakeholder in charge which is Ultimate MMA Fitness & Self Defense, Mr. Alvin Yap. From those initial requirements, functional and non-functional requirements were defined. The tool Visual Paradigm is used to depict diagrams and by using the unified modeling language, use case diagrams, use case specifications, activity diagrams, sequence diagrams and also the requirement traceability matrix will be drawn in order to further understand and analyse the proposed system.

3.2.3 Design Phase

During the design phase, front-end and back-end of the system are preliminary designed. In this phase rough wireframe mock-ups are drawn to depict what the prototype should look like. Data management diagrams concerning with designing, creating, deploying, and managing the system's data architecture are drawn to visualize and normalize while reducing data anomalies in the final system. The quick design helps in developing the prototype. The system architecture diagram and user interface design is illustrated. The user interface design is drawn using a low-fidelity wireframe mock-up by using Figma which is a vector based graphic editor. The database design is also mapped out during this phase which includes, database schema and data dictionary.

3.2.4 Prototyping Phase

In the prototyping system phase where an actual prototype is designed based on the information gathered from the prior phases. A prototype will be built for each module and will contain various types of content ranging from visuals to navigation structure which will be built using node.js as a Restful API and flutter as the client-side programming framework using the Dart programming language..

3.2.5 User Evaluation Phase

The fourth phase in this model is the User Evaluation phase. The cycle of designing, prototyping, and acquiring user evaluation are iterative meaning that these phases are recurrent until the user is satisfied with the overall look and feel of the prototype. During this phase, the manager of Ultimate MMA Fitness & Self Defense, Mr. Alvin Yap will evaluate the latest iteration of each module prototype. Until the user's (Mr. Alvin Yap) satisfaction for each module prototype has not been fulfilled, more refinement is made to further fortify the individual prototypes based on the user's (Mr. Alvin Yap) suggestions and feedback. This prototyping cycle (Design Phase, Prototyping System Phase, and User Evaluation Phase) will be over until all the requirements specified by the user (Mr. Alvin Yap) are met.

3.2.6 Implementation Phase

Once the user is satisfied with the developed prototype, a final system is developed based on the approved final prototype. This is called the Implementation phase. During this phase, node.js – an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine will be used to implement the back-end APIs of the system. The V8 engine gives Node a huge boost in performance because it cuts out the middleman, preferring straight compilation into native machine code over executing bytecode or using an interpreter. Flutter will be used to build the front-end of the mobile application (Mike et al., 2014). Flutter is a cross-platform framework that aims at

developing high-performance mobile applications (Wenhao, 2019). It is an open-source UI software development kit created by Google will be used to implement the front-end user interface. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase which reduces significant amount of time to implement the system.

3.2.7 Testing Phase

After the final system is developed based on the final prototype, it will thoroughly be tested and deployed to production which will be conducted during the Testing phase. The system undergoes routine maintenance for minimizing downtime and prevent large-scale failures.

3.3 System Development Workflow

The Gantt chart of the system is attached in Appendix A which shows the time and resources needed to develop the mobile application. There are total of nine phases from the prototype model. As shown in Table 3.1, each phase has its own task and output that need to produce during the entire project development. The table further shows the phases, tasks within a phase, and the output of the task in a concise manner.

Table 3.1: Software development activities and their task

Phase	Task	Output
Planning	<ul style="list-style-type: none"> Identify the Objectives, Problems and Scope. Identify the Software Requirements. 	<ul style="list-style-type: none"> Proposal. Gantt Chart.
Requirement Analysis	<ul style="list-style-type: none"> Data Collection. Information Analysis. Identify Specific Programs and Programming Language. Create UML (use case diagram), Class Diagram and RTM. 	<ul style="list-style-type: none"> The functional and nonfunctional requirements of the system. Software Requirement Specification

		Document. UML Diagrams <ul style="list-style-type: none"> • Class diagram • Requirements Traceability Matrix (RTM) • To-be model (Swimlane diagram)
Design	<ul style="list-style-type: none"> • Design User Interface of The System. • Design Application Programming Interface of The System. • Design the Flow of Backend to Frontend. 	<ul style="list-style-type: none"> • System architecture • Database design (schema and data dictionaries) • User interface design
Prototyping System	<ul style="list-style-type: none"> • Creating prototype for Sign Up and Log In module. • Creating prototype for Coach module. • Creating prototype for Student module. • Creating prototype for Administrator module. • Integrate all of the module into one prototype. 	<ul style="list-style-type: none"> • Figma mockup high fidelity prototype.
User Evaluation	<ul style="list-style-type: none"> • System is presented to the client for an evaluation. • Collect suggestion and feedback. • Analyze Suggestion and feedback. 	<ul style="list-style-type: none"> • Test cases
Design II	<ul style="list-style-type: none"> • Design user interface of the system. • Design application programming interface of the system. • Design the flow of backend to frontend. 	<ul style="list-style-type: none"> • The refined wireframe sketch of the system.
Prototyping System II	<ul style="list-style-type: none"> • Refine prototype for Sign Up and Log In module. • Refine prototype for Coach module. • Refine prototype for Student module. • Refine prototype for Administrator module. • Refine all of the module into one prototype. 	<ul style="list-style-type: none"> • Refined Figma mockup high fidelity prototype.
User Evaluation II	<ul style="list-style-type: none"> • System is presented to the client for an evaluation. • Collect suggestion and feedback. • Analyze Suggestion and feedback. 	<ul style="list-style-type: none"> • Test cases
Implementation	<ul style="list-style-type: none"> • Implementing Sign Up and Log In module. • Implementing prototype for Coach module. • Implementing prototype for Student module. • Implementing prototype for Administrator module. • Integrate all of the module into one whole system. 	<ul style="list-style-type: none"> • Flutter front-end. • Node.js back-end. • Firebase server.

Testing	<ul style="list-style-type: none">• Pilot Testing.• System Testing.	<ul style="list-style-type: none">• Test cases
---------	--	--

3.4 Chapter Summary

The incremental prototyping model as a software methodology is used in this project. The advantages of using the incremental prototyping software methodology is that it reduces the feedback time of the user and is efficient throughout the development process. There are 5 total phases within the model, 3 of which are iterative. As the prototype evolves with each iteration, it is closer in meeting the user expectations. The system development workflow involves the design phase, prototype phase and also the user evaluation to be iterated twice as to ensure the pivotal functionality of the user requirements are met.

CHAPTER 4

ANALYSIS AND DESIGN

4.1 System Requirement Analysis

System requirement analysis can be defined as a process and technique of defining conditions or user requirements for a modification of an existing system or developing a new system (Demirel and Das, 2018). However, in requirement engineering, it is more specific to capturing and analysing requirements. To fully develop a competent system, the process of analysing requirement is mandatory. The requirements of the mobile application are analysed. The modules, functions and its users are briefly summarised into table 4.1. Table 4.2 and 4.3 shows the functional and non-functional requirements of the Coach Me Mobile Application respectively.

Table 4.1: Modules, Function and User for Coach Me Mobile Application

System Module	Function	User
Sign Up and Log In module	This module is responsible for routing the users to dedicated pages when authenticated.	Administrator, Coach and Student
Coaching module	This module is responsible for the coach user to manage their class, revenues and payments, lessons and pre-recorded videos.	Coach

Learning module	This module is responsible for the student user to enrol or unenroll from classes, lessons, watch pre-recorded videos and make payments.	Student
Scheduling module	To manage a personal one-to-one session between the coach and student user.	Coach, Student
Administrator module	To produce reports and handle account deletions as well as issuing refunds.	Administrator

4.1.1 Functional and Non-Functional Requirements

Functional requirement is defined as capturing the expected behaviour of the system such as certain task to perform, services or the functions that are required for the system to perform (Malan and Bredemeyer, 2001). It essentially consists of what the system is expected not to do or the other way around. Table 4.2 shows the functional requirements by each system module for the mobile application.

Table 4.2: Functional Requirements Specification for Coach Me Mobile Application

No.	System Module	Functionality
1.	Sign Up and Log In module	<ol style="list-style-type: none"> 1. The system should allow the users to sign up by filling up their first name, last name, username, password, role, email, phone number and address. 2. The system should allow the users to log in with the correct username and password. 3. The system should allow the users to reset their password through their email in case if the users forget their password. 4. The system should show a message if the username or password is incorrect. 5. The system should allow the users that are signed in to logout.
2.	Coaching module	<ol style="list-style-type: none"> 1. The system is able to show the coach user their account information. 2. The system is able to show the coach user which students are enrolled in which class. 3. The system should allow the coach user to create classes.

		<ol style="list-style-type: none"> 4. The system should allow the coach user to create lessons within each created class. 5. The system should allow the coach user to upload pre-recorded videos within each created lesson. 6. The system should allow the coach user to accept or decline a private one to one session request from a student user. 7. The system is to ensure the coach user receives appropriate payment when the student user enrolls in their class.
3.	Learning module	<ol style="list-style-type: none"> 1. The system is able to show the student user their account information. 2. The system is able to show the student user their enrolled classes. 3. The system should show the student user their available lessons 4. The system should allow the student user to watch pre-recorded videos within a lesson. 5. The system should allow the student user to make payments during class enrolment. 6. The system should allow the student to issue a refund within 72 hours after class enrolment. 7. The system should allow the student user to request a personal one-to-one session with a coach based on the availability of the coach.
4.	Scheduling module	<ol style="list-style-type: none"> 1. The system should allow the coach user to manage their available date and time for a personal one to one session 2. The system should allow the student user to pick available date and time for a personal coaching session. 3. The system should not show the date and time that has passed. 4. The student user cannot pick a date and time that has passed.
5.	Administrator module	<ol style="list-style-type: none"> 1. The system should allow the administrator to approve student and coach user account deletion. 2. The system should allow the administrator to accept or decline a refund issue. 3. The system should generate monthly report for the coach rating. 4. The system should generate monthly report on the most popular categories.

		<ol style="list-style-type: none"> 5. The system should generate a monthly report for the number of engaged students. 6. The system should not allow the student and coach user to view the monthly reports.
--	--	--

Non-functional requirements describe how should the system perform said task or the performance, reliability, security and also the portability of the system (Chung, 2012). Such requirements are key to designing a good system as it decides the alternative design, and characteristic selection of a system. Table 4.3 shows the non-functional requirements for the mobile application.

Table 4.3: Non-functional Requirements Specification for Coach Me Mobile Application

No.	Requirements	Description
1.	Operational	<ol style="list-style-type: none"> 1. The mobile application is only for the Android operating system. 2. The update of the mobile application should be easily rolled out. 3. The application should be user-friendly.
2.	Performance	<ol style="list-style-type: none"> 1. The time to load the mobile application should not be more than 3000 milliseconds.
3.	Security	<ol style="list-style-type: none"> 1. The user password should be encrypted before storing in database.

4.1.2 User Requirement Analysis

User requirement is critical in designing a system. It can be described as a process of defining and stating the user expectations for a new system or a system modification (Maguire and Bevan, 2002). Table 4.4 shows the user requirements specification for the mobile application.

Table 4.4: User Requirements Specification for Coach Me Mobile Application

No.	User Requirements
1.	All users are only allowed to sign up only when the username, password and email is filled.
2.	All users can log in to the system with the correct username and password.
3.	Each role of user will be directed to their own specified homepage.
4.	All users can request to reset their password with their username and email address.
5.	The coach user is able to manage classes.
6.	The coach user is able to manage lessons within classes.
7.	The coach user is able to manage pre-recorded videos within lessons.
8.	The coach user can see students enrolled in their class their profile
9.	The coach user can receive payments or revenues from an enrolled class.
10.	The coach user is able to accept or decline session requests.
11.	The student user is able to enrol or unenroll from a class.
12.	The student user is able to view a brief information on each lesson within a class.
13.	The student user is able to watch pre-recorded videos within an enrolled class.
14.	The student user is able to make payments for a paid class.
15.	The student user is able to request a personal session from a specific coach given their available date and time.

4.1.3 System Hardware and Software Requirements

System hardware and software requirements should be met in order for the system to perform. It can be simply described as the hardware and software compatibility of the computer's resources which in this case is a mobile application (Bourque et al., 1999). Table 4.5 and 4.6 shows the mobile application hardware and software requirements specification respectively.

Table 4.5: Hardware Requirements Specification for Coach Me Mobile Application

No.	Hardware	Specification
1.	CPU (Central Processing Unit)	<ul style="list-style-type: none"> Intel Core i5 8th Gen 8250U
2.	RAM (Random Access Memory)	<ul style="list-style-type: none"> 4 Gigabyte (GB).
3.	Storage	<ul style="list-style-type: none"> 500 Gigabyte (GB).
4.	Mobile Device	<ul style="list-style-type: none"> Android Version 5.0 Lollipop and above.

Table 4.6: Software Requirements Specification for Coach Me Mobile Application

No.	Software	Specification
1.	Windows 10	<ul style="list-style-type: none"> Operating system.
2.	Flutter	<ul style="list-style-type: none"> As a front-end framework.
3.	Node.js	<ul style="list-style-type: none"> As a back-end framework.
4.	Firebase	<ul style="list-style-type: none"> As a database management system.

4.1.4 UML Diagram

UML diagram is key in designing an overall system. It transfers the proposed system functionality into the graphical diagram according to specific requirements. In this section, the system design of the proposed system is described. Object oriented approach is used

to generate the UML diagrams which are Use Case Diagram, Activity Diagram, Sequence Diagram and Class Diagram.

4.1.5 Use Case Diagram

A use case diagram can be defined as the primary depiction of the user's interaction whereby the base use case is the user's expected behavior. Figure 4.1 shows the use case diagram that represents the overall activity of the Coach Me Mobile Application.

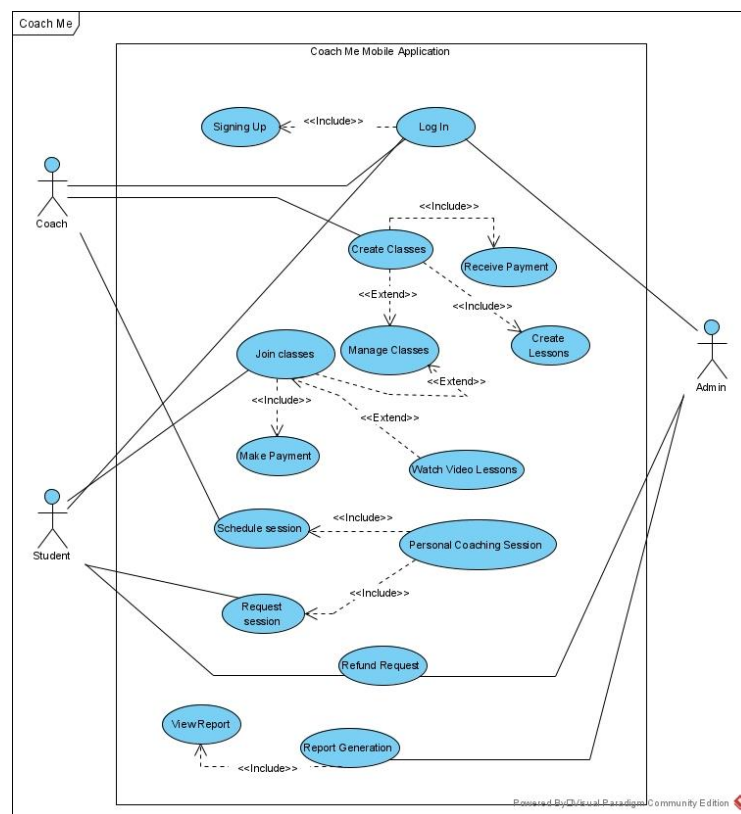


Figure 4.1: Use Case Diagram of Proposed System

4.1.6 Use Case Specification

Use case specification is used to describe and explain the details of use case diagrams of Coach Me Mobile Application. The following section is the overview of each use case.

(i) **Use Case 1: User Login**

The intended users of the system are customers and Administrator who use their username and password to sign in. Users can perform their activities once the data they enter is valid. If the customers are a new user, they will need to sign up first. The data will be stored in the database.

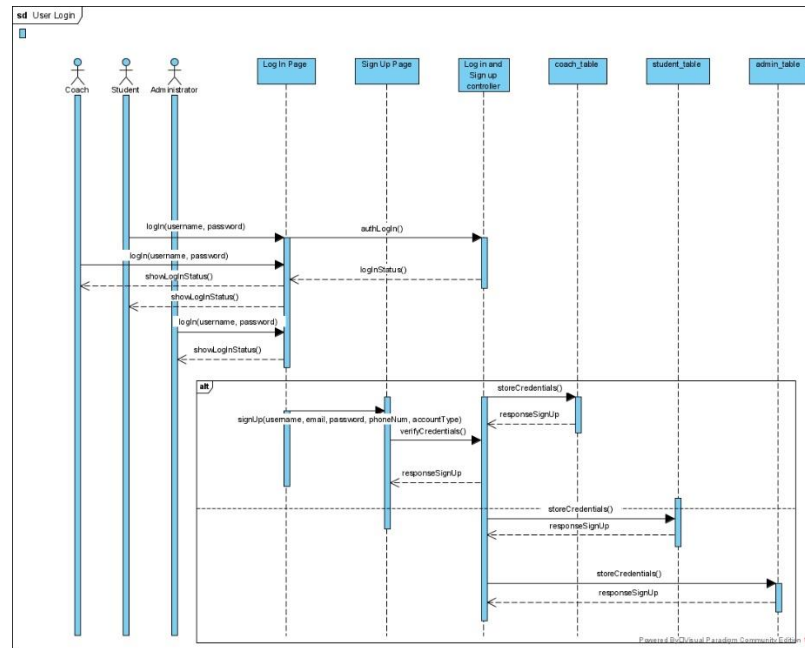
Table 4.7: User Login Use Case Specification

History Log	1.0.0	1. Create initial use case	
	1.0.1	1. Fixed alternative flow in normal flow. 2. Changed ID for alternative flow and exceptions.	
	2.0.0	1. Added history log. 2. Functional requirements are added.	
Version	2.0.0		
Use Case ID	UC-1		
Use Case Name	User Login		
Created By	Bassiman	Updated By	Bassiman
Date Created	03/12/21	Last Revision Date	06/12/21
Actors	Coach, Students, Administrator		
Description	Users’ login to access system		
Preconditions	Users need to insert username and password.		
Post conditions	The users will be redirect to the homepage		
Normal Flow	1.0 Verify the users 1. Users need to insert valid username and password. 2. System redirects users to the homepage. 3. If the customers is new user, they need to register account first, see A.1.		
Alternative Flow	A.1.0 Register 1. New coach and student insert username, password, contact number and email address. 2. System display ‘Registration Success” toast message 3. User is redirect to home page. 4. Return to step 1 in normal flow.		
Exceptions	E.1.0 Fail to login		

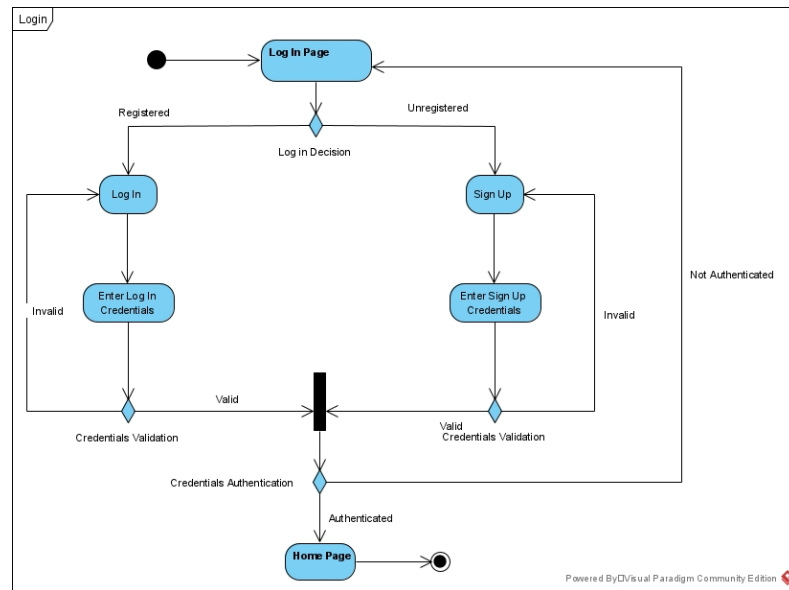
	1. System displays toast to inform users that invalid username or password.		
Related requirement	ID	Requirement	Priority
	FR01-01	The system should allow the users to sign up by filling up their first name, last name, username, password, role, email, phone number and address.	Basic
	FR01-02	The system should allow the users to log in with the correct username and password.	High
	FR01-03	The system should show a message if the username or password is incorrect.	Basic

Table 4.7: User Login Use Case Specification (Continued)

Sequence Diagram:



Activity Diagram:



(ii) **User Case 2: Manage Classes**

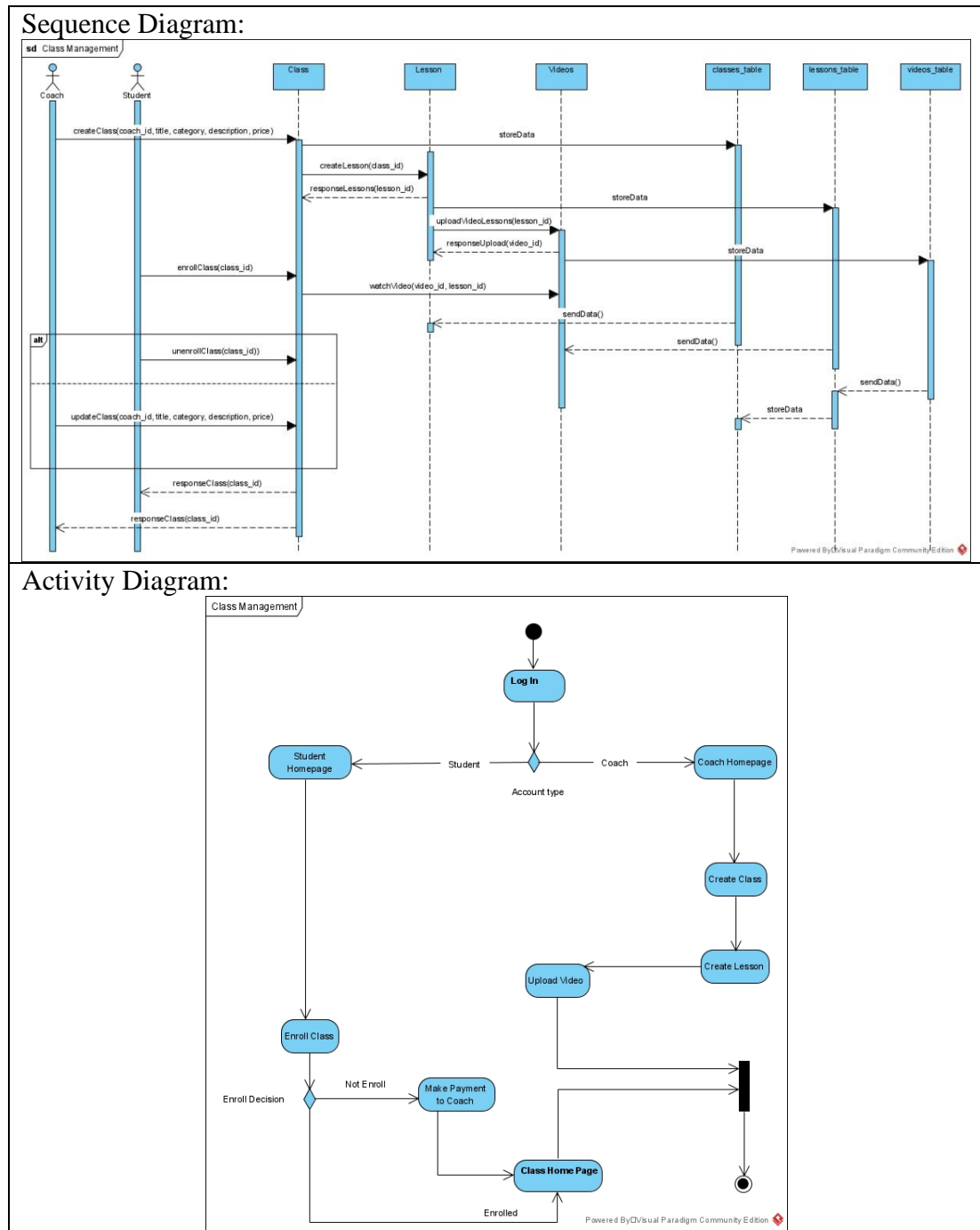
Table 4.8 shows the use case specification of manage classes. Users can manage their classes for example coach can create classes and student can enrol or unenroll.

Table 4.8: Manage Class Use Case Specification

History Log	1.0.0	1. Create initial use case	
	1.0.1	1. Fixed alternative flow in normal flow. 2. Changed ID for alternative flow and exceptions.	
	2.0.0	1. Added history log. 2. Functional requirements are added.	
Version	2.0.0		
Use Case ID	UC-3		
Use Case Name	Manage classes		
Created By	Bassiman	Updated By	Bassiman
Date Created	03/12/21	Last Revision Date	05/12/21
Actors	Coach, Student		
Description	Manage and join classes, lessons and watch videos and make, receive payments		
Preconditions	1. User must be logged in 2. Logged in account type must be not be Administrator.		
Post conditions	Coach User Must Have Available Sessions set.		
Normal Flow	2.0 Student User Join classes 1. Student Browses interesting classes. 2. Student touches “More Info” button. 3. Class info page is displayed. 4. “Join Class” button is pushed. 5. Student Makes payments to join class 6. Student are shown lessons page 7. Student can watch videos within the lesson		
	2.1 Coach User Manage Class 1. Coach Creates Class by filling information such as class category, name, title, number of lessons and short description. 2. Coach creates lessons within a class.		

	<ol style="list-style-type: none"> Coach upload video on a lesson Class is made public for every user to see. Coach receives payment when user join the class 		
Alternative Flow	<p>A.2.0 Student Unenroll form class</p> <ol style="list-style-type: none"> Student presses on “Unenroll button”. Lessons in class is no longer available for student watch. <p>A2.1 Coach Updates Class</p> <ol style="list-style-type: none"> Coach presses “Update Class” button Coach re-enter information such as class category, name, title, number of lessons and short description. “Confirm” button is pressed. 		
Exceptions	<p>E.2.0 Student fails to join class</p> <ol style="list-style-type: none"> System displays message failed to join class 		
Related requirement	ID	Requirement	Priority
	FR02-01	The system is able to show the coach user their account information.	Basic

Table 4.8: Manage Classes Use Case Specification (Continued)



(iii) User Case 3: Personal Coaching Session

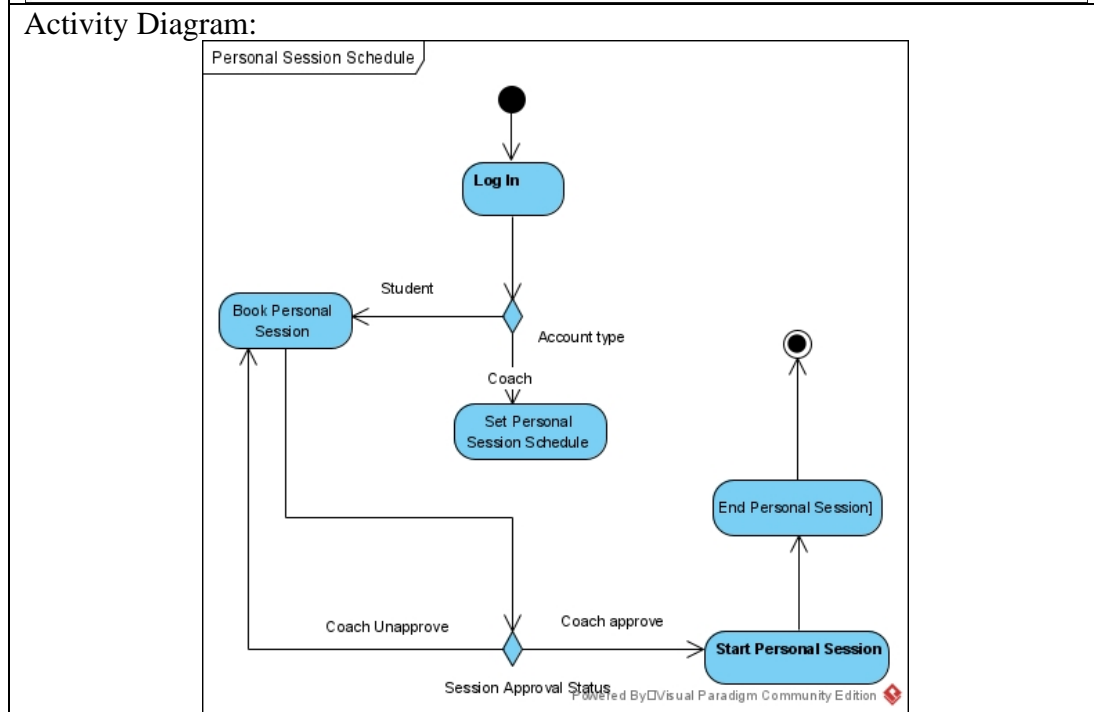
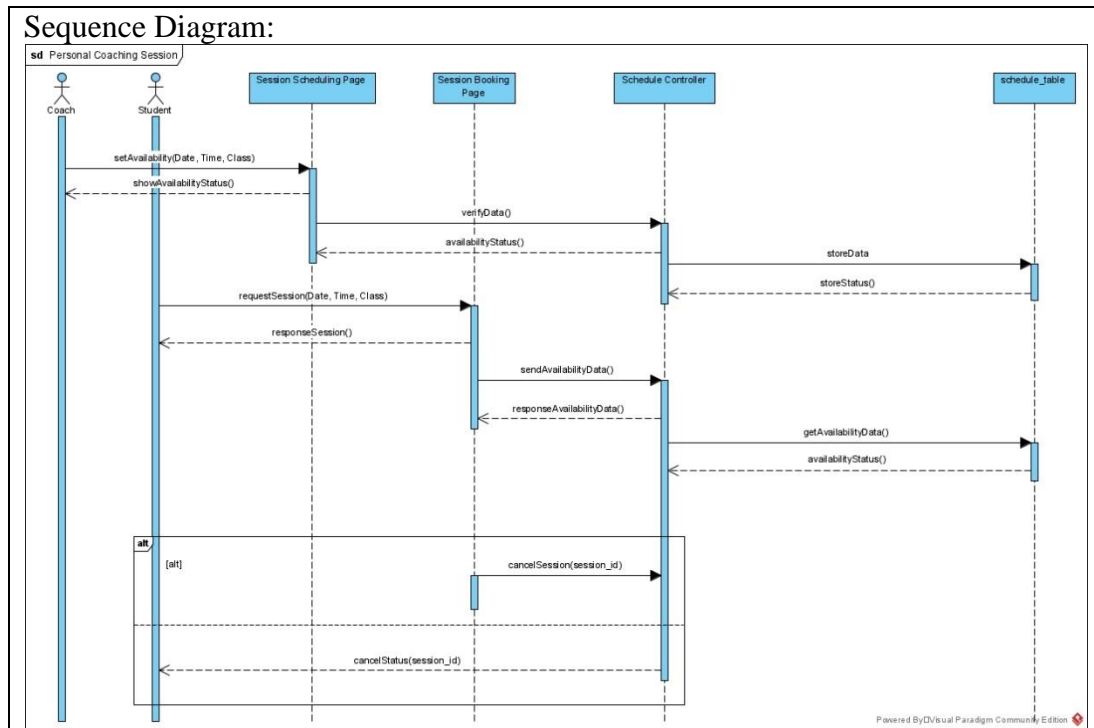
Table 4.9 shows the use case specification of personal coaching session. Users can schedule for personal coaching.

Table 4.9: Personal Coaching Session Use Case Specification

History Log	1.0.0	1. Create initial use case	
	1.0.1	1. Fixed alternative flow in normal flow. 2. Changed ID for alternative flow and exceptions.	
	2.0.0	1. Added history log. 2. Functional requirements are added.	
Version	2.0.0		
Use Case ID	UC-3		
Use Case Name	Personal Coaching Session		
Created By	Bassiman	Updated By	Bassiman
Date Created	03/12/21	Last Revision Date	05/12/21
Actors	Coach, Student		
Description	Request for personal coaching session		
Preconditions	1. User must be logged in 2. Logged in account type must be not be Administrator.		
Post conditions	-		
Normal Flow	3.0 Student User Request Session 8. Student picks date and time of an available session. 9. Student touches “Confirm” button. 10. Session info page is displayed. 11. Student presses “Join session” button.		
	3.1 Coach User Set Session 6. Coach Creates available session by entering date and time. 7. System displays booked session. 8. Session is created at start time. 9. Coach presses “Join Session” button 10. Session is destroyed at end time.		
Alternative Flow	A.3.0 Student Cancel session		

	3. Student presses on “Cancel Session button”. 4. Session is destroyed.		
Exceptions	E.3.0 Student fails to request session 2. System displays message failed to join class		
Related requirement	ID	Requirement	Priority
	CR03-01	The student user cannot pick a date and time that has passed.	Basic

Table 4.9: Personal Coaching Session Use Case Specification (Continued)



(iv) User Case 4: Refund

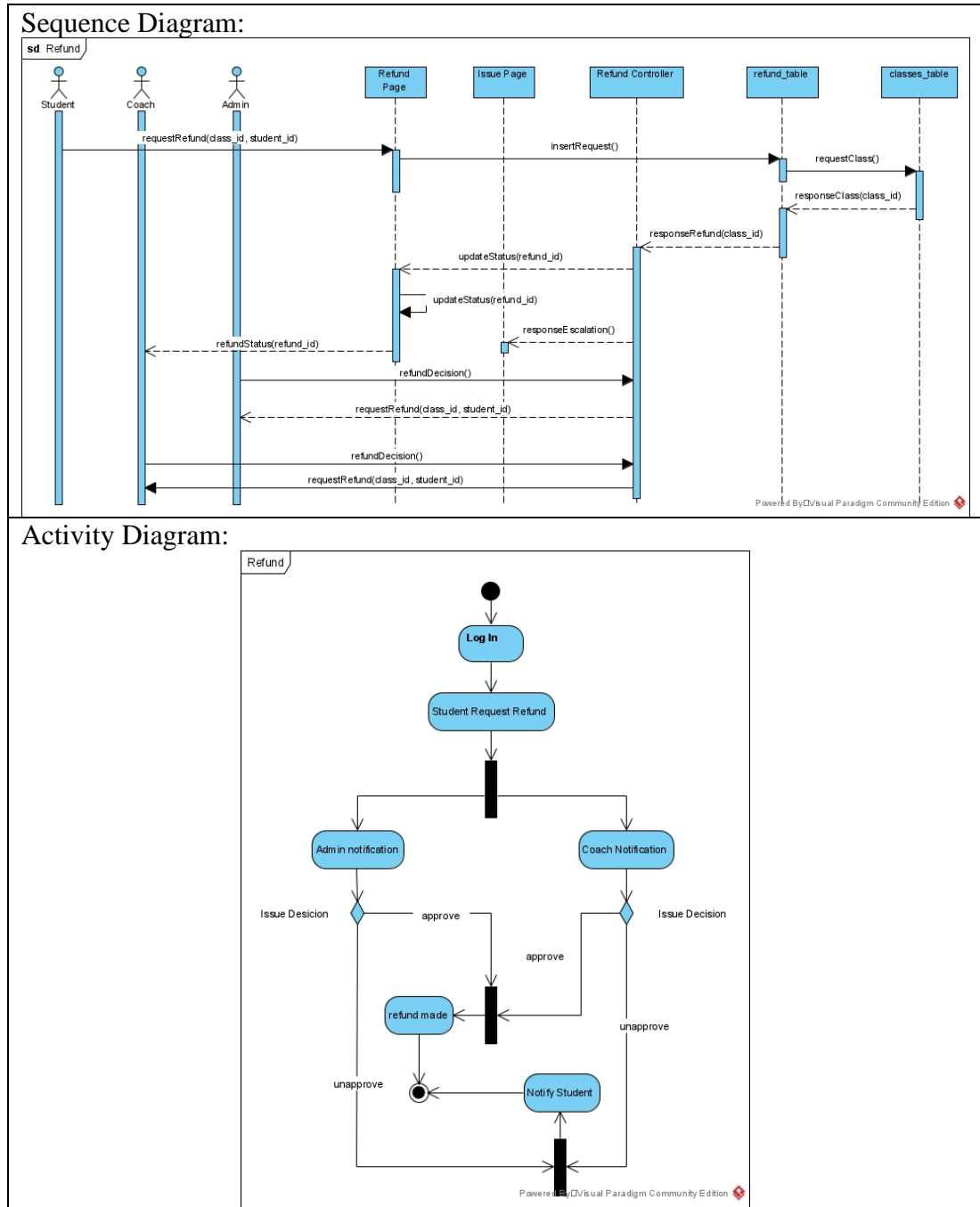
Table 4.10 shows the refund use case specification. Users could request refunds if the class joined in unsatisfactory.

Table 4.10: Refund Use Case Specification

History Log	1.0.0	1. Create initial use case	
	1.0.1	1. Fixed alternative flow in normal flow. 2. Changed ID for alternative flow and exceptions.	
	2.0.0	1. Added history log. 2. Functional requirements are added.	
Version	2.0.0		
Use Case ID	UC-4		
Use Case Name	Refund		
Created By	Bassiman	Updated By	Bassiman
Date Created	03/12/21	Last Revision Date	05/12/21
Actors	Coach, Student, Administrator		
Description	Issuing refunds		
Preconditions	3. User must be logged in 4. Student must be enrolled in the class		
Post conditions	-		
Normal Flow	4.0 Student Request Refund Approved 1. Student presses refund class button. 2. System issues refund request to Administrator and coach. 3. Coach does not respond within 2 days. 4. Administrator accepts refund request. 5. Refund is made to student.		
Alternative Flow	A.4.0 Student Request Refund Unapproved 1. Coach responds within 2 days. 2. Administrators decline refund request. 3. Refund is not made to student. 5. System displays refund status to student.		
Exceptions	-		
Related requirement	ID	Requirement	Priority

	FR04-01	The system should allow the administrator to accept or decline a refund issue.	Basic
--	---------	--	-------

Table 4.10: Refund Use Case Specification (Continued)



(v) **Use Case 5: Report Generation**

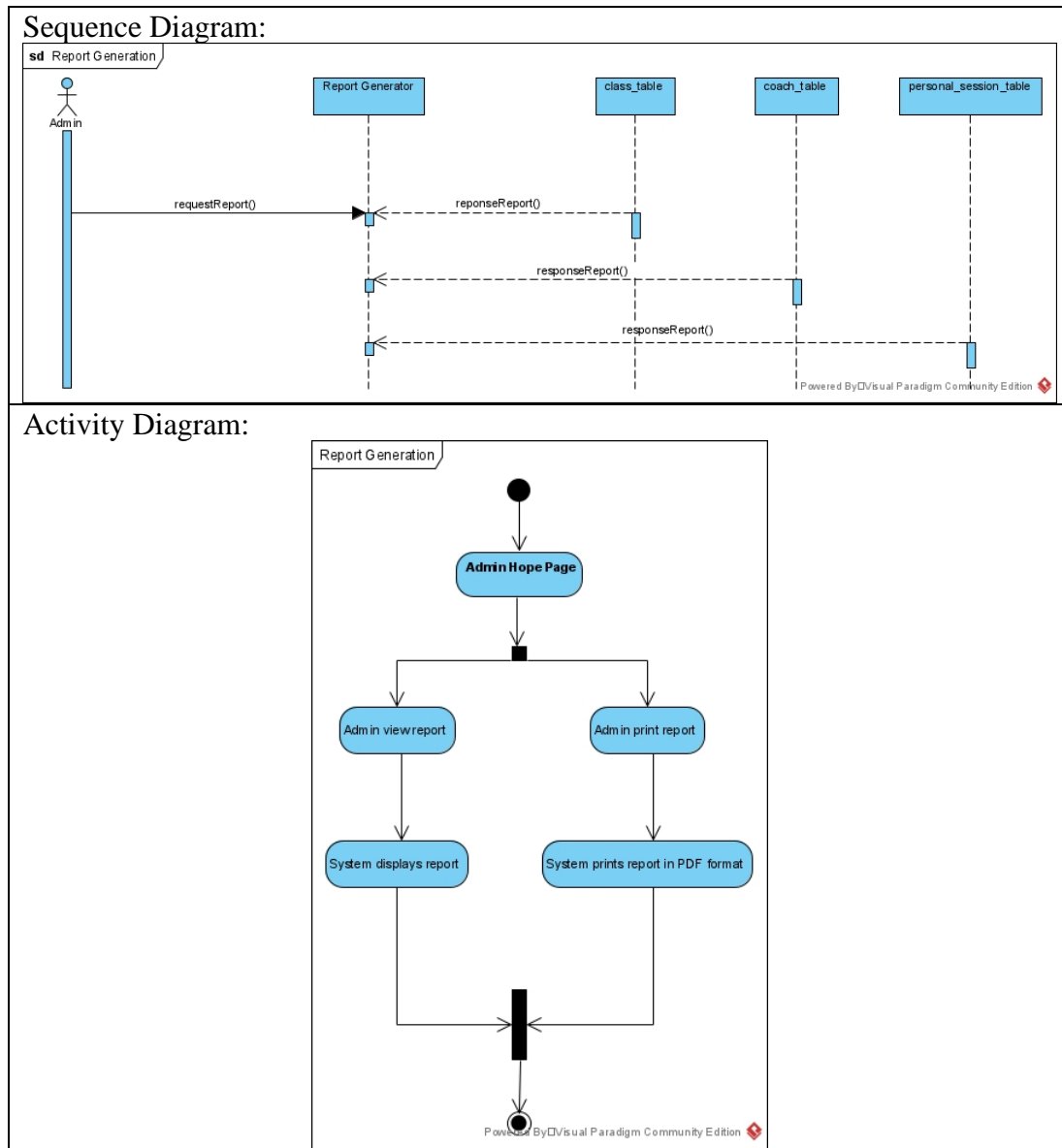
Table 4.11 shows the use case specification of report generation. User specifically the administrator can request for report generation.

Table 4.11: Report Generation Use Case Specification

History Log	1.0.0	1. Create initial use case	
	1.0.1	1. Fixed alternative flow in normal flow. 2. Changed ID for alternative flow and exceptions.	
	2.0.0	1. Added history log. 2. Functional requirements are added.	
Version	2.0.0		
Use Case ID	UC-5		
Use Case Name	Report Generation		
Created By	Bassiman	Updated By	Bassiman
Date Created	03/12/21	Last Revision Date	05/12/21
Actors	Administrator		
Description	Generates report for Administrator		
Preconditions	1. User must be logged in 2. Logged in account type must be Administrator.		
Post conditions	-		
Normal Flow	5.0 Administrator view report 1. Admin chooses reporting tab. 2. System generates report for highest rated coach. 3. System generates report for most enrolled class. 4. System generates report for personal sessions per week/ month.		
Alternative Flow	A5.0 Administrator print report 1. User presses button print report 2. System Generates report in pdf format		
Exceptions	E.5.0 Administrator fails to request report 1. System displays message failed to generate report		
Related requirement	ID	Requirement	Priority

	FR05-01	The system should not allow the student and coach user to view the monthly reports.	Basic
--	---------	---	-------

Table 4.11: Report Generation Use Case Specification (Continued)



4.1.7 Class Diagram

4.1.8 To-Be Model

To be model depicts how the system should behave as a proposed system. Moreover, the to-be model describes basic user and their actions. Figure 4.3 shows the to-be model of the mobile application.

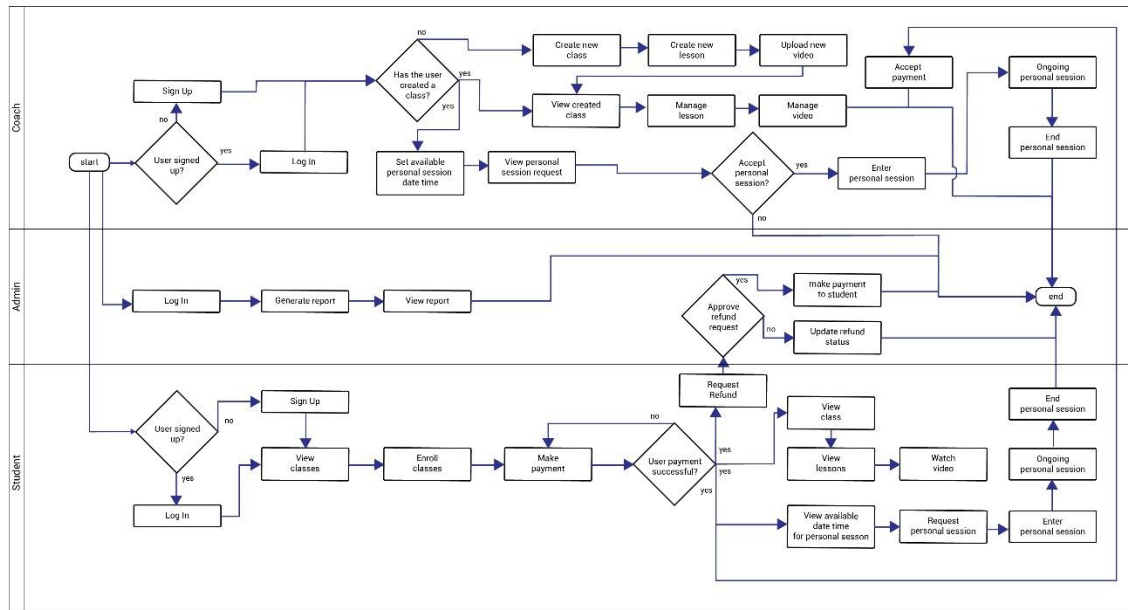


Figure 4.3: To-Be Model of Coach Me Mobile Application

4.1.9 Requirement Traceability Matrix

Requirement Traceability Matrix (RTM) is a document that collects all the user's recommended requirements and proposed system specifications. It is used to determine whether system requirements are met. Therefore, any system features will not be missed during testing. Table 4.12 summarizes the performance matrix of the proposed system requirements.

Table 4.12: Requirement Traceability Matrix of the Proposed System.

Requirement	Software Requirement Specification	Description
Log In and Sign Up SRS_REQ_100	SRS_REQ_101	Students sign up by completing sign up form.
	SRS_REQ_102	Student presses Sign Up button.
	SRS_REQ_103	System verify Sign Up form.
	SRS_REQ_104	System verifies username and password
	SRS_REQ_105	System shows message(s) when Sign Up form is incomplete.
	SRS_REQ_106	System redirect student homepage if verification is complete.
	SRS_REQ_107	Coach sign up by completing sign up form
	SRS_REQ_108	Coach presses Sign Up button
	SRS_REQ_109	System verify Sign Up form
	SRS_REQ_110	System verifies username and password
	SRS_REQ_111	System shows message(s) when Sign Up form is incomplete
	SRS_REQ_112	System redirect coach homepage if verification is complete.
	SRS_REQ_113	Users enter username and password to log in.
	SRS_REQ_114	User presses Log In button
	SRS_REQ_115	System verifies username and password
	SRS_REQ_116	System will redirect user to appropriate homepage
Learning SRS_REQ_200	SRS_REQ_201	System displays enrolled classes.
	SRS_REQ_202	Student browses coaches.
	SRS_REQ_203	Student taps coaches to see more on profile.
	SRS_REQ_204	System displays coach certification and classes.
	SRS_REQ_205	Student browses available classes.
	SRS_REQ_206	Student presses more info button.

	SRS_REQ_207	System display information on class.
	SRS_REQ_208	Student presses Enroll button.
	SRS_REQ_209	Student is redirected to payment confirmation page.
	SRS_REQ_210	Student is redirected to payment gateway page.
	SRS_REQ_211	Upon successful payment, student is welcomed to class page.
	SRS_REQ_212	Student browses lessons within class.
	SRS_REQ_212	Student presses Watch This lesson to play video.
Coaching SRS_REQ_300	SRS_REQ_301	System displays coach classes.
	SRS_REQ_302	Coach creates class by filling up create class form.
	SRS_REQ_303	Coach creates lessons by filling up lesson form.
	SRS_REQ_304	Coach Upload video in the lesson.
	SRS_REQ_305	Upon student enrollment, coach receives payment.
	SRS_REQ_306	Coach can view students in a class by pressing view student button.
	SRS_REQ_307	Coach Update Class information by pressing Update Class button.
	SRS_REQ_308	System Updates Class Information.
	SRS_REQ_309	Coach Update lesson information by pressing update lesson Button.
	SRS_REQ_310	System updates lesson information.
	SRS_REQ_311	Coach presses delete lesson button.
	SRS_REQ_312	System drops lesson from database records.
	SRS_REQ_313	Coach presses delete class button.
	SRS_REQ_314	System drops class from database records.
Scheduling SRS_REQ_400	SRS_REQ_401	Coach presses set available session button.
	SRS_REQ_402	system shows time and date for a personal session in a class.
	SRS_REQ_403	Student picks time and date for personal session in class.
	SRS_REQ_404	Coach presses confirm button.
	SRS_REQ_405	Coach receive notification for the requested session.
Administrator SRS_REQ_500	SRS_REQ_501	Administrator chooses reporting tab.
	SRS_REQ_502	System generates report for highest rated coach.
	SRS_REQ_503	System generates report for most enrolled class.
	SRS_REQ_504	System generates report for personal sessions per week/ month.

	SRS_REQ_505	Administrator presses button print report.
	SRS_REQ_506	System generates report in pdf format.
	SRS_REQ_507	Student presses refund class button.
	SRS_REQ_508	System issues refund request to Administrator and coach.
	SRS_REQ_509	Coach does not respond within 2 days.
	SRS_REQ_510	Administrator accepts refund request.
	SRS_REQ_511	Refund is made to student.
	SRS_REQ_512	Coach responds within 2 days.
	SRS_REQ_513	Administrators decline refund request.
	SRS_REQ_514	Refund is not made to student.
	SRS_REQ_515	System displays refund status to student.

4.2 Design

System design can be defined as the process of designing or laying out the elements within the system such as the user interface, architecture and also the database records as well as its relations. (Bruegge and Dutoit, 2009) It is important so that sufficient information and details of a data is provided to implement the system.

4.2.1 System Architecture

An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components (Weilkiens et. al., 2015) It is significant in the system design phase as it provides a backbone of how the system should be implemented. Figure 4.4 Shows the system architecture diagram of the mobile application.

Figure 4.4: System Architecture Diagram for Coach Me Mobile Application

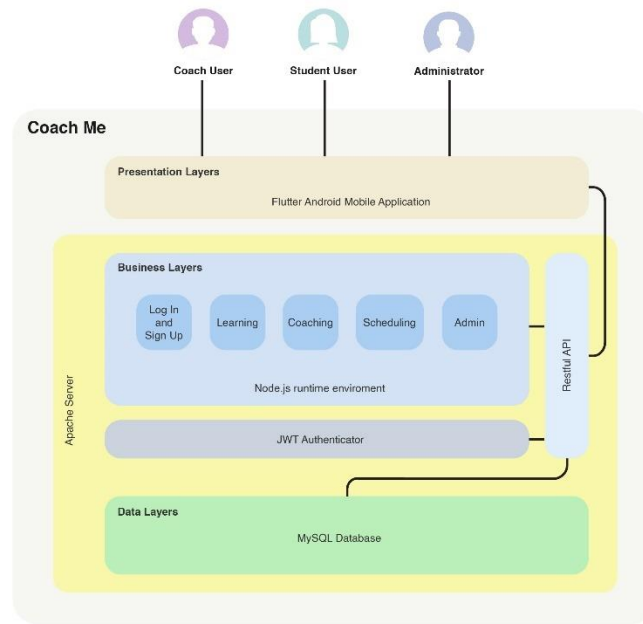


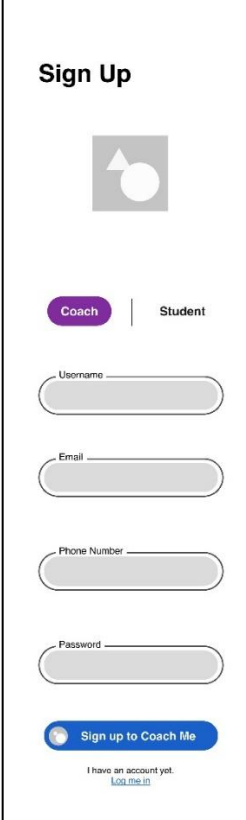
Figure 4.4: System Architecture Diagram for Coach Me Mobile Application

4.2.2 System User Interface Design

It is the process designers use to build interfaces in software or computerized devices, focusing on looks or style. Designers aim to create interfaces which users find easy to use and pleasurable (Johnson, 2010). Thus, a simple wireframe mock up is illustrated.

4.2.2.1 Sign Up

The user can sign up by filling up the user type username, email, phone number and password form. Once signed up, the user will be routed to their respective homepage. Figure 4.5 shows the sign-up user interface.

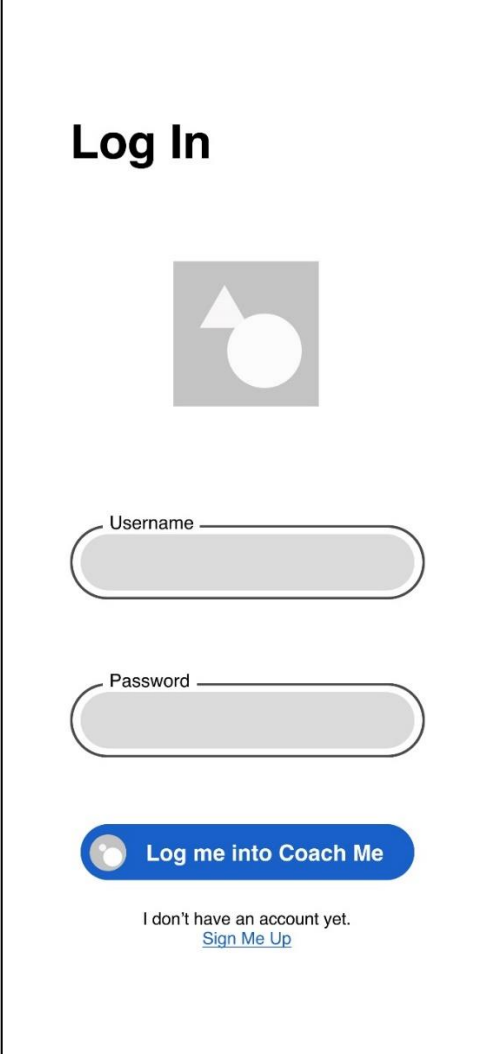


The image shows a mobile app sign-up screen. At the top, the title "Sign Up" is displayed. Below it is a logo consisting of a grey square with a white circle and a triangle. Under the logo are two buttons: "Coach" (purple) and "Student" (grey). Below these are four input fields: "Username", "Email", "Phone Number", and "Password", each with a label and a rounded rectangular input area. At the bottom is a blue button with a white circular icon and the text "Sign up to Coach Me". Below this button is a link that says "I have an account yet. [Log me in](#)".

Figure 4.5: Sign Up User Interface

4.2.2.2 Log In

The user can log into the system by filling out their username and password. Once the user is authenticated, they will be routed to their respective homepage. Figure 4.6 shows the log in user interface for the mobile application.



The image shows a mobile application login screen. At the top, the text "Log In" is displayed in a large, bold, black font. Below this is a square icon with a gray background, containing a white circle and a white triangle. Underneath the icon are two input fields: the first is labeled "Username" and the second is labeled "Password". Both fields are represented by rounded rectangles with a light gray fill and a thin black border. Below the password field is a blue button with rounded corners, featuring a white circular icon on the left and the text "Log me into Coach Me" in white. At the bottom of the screen, the text "I don't have an account yet." is displayed, followed by a blue, underlined link that says "Sign Me Up".

Figure 4.6: Log In User Interface

4.2.2.3 Student Homepage

The student user can search for classes, view classes by their categories, view featured coaches, and also view featured classes. Figure 4.7 shows the student homepage of the mobile application. It has a horizontal list view of featured coaches and featured classes.

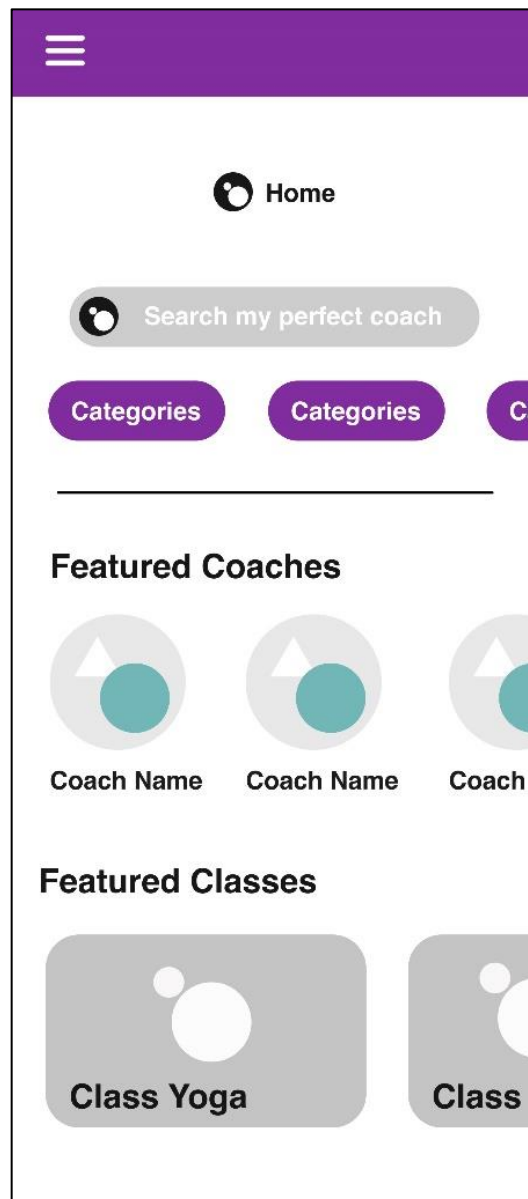


Figure 4.7: Student Homepage User Interface

4.2.2.4 Student Profile

This user interface shows the necessary details of the student such as their name, username, email and also phone number and their purchase history. This user interface is private only to the students to view. Figure 4.8 shows the user interface for student profile.

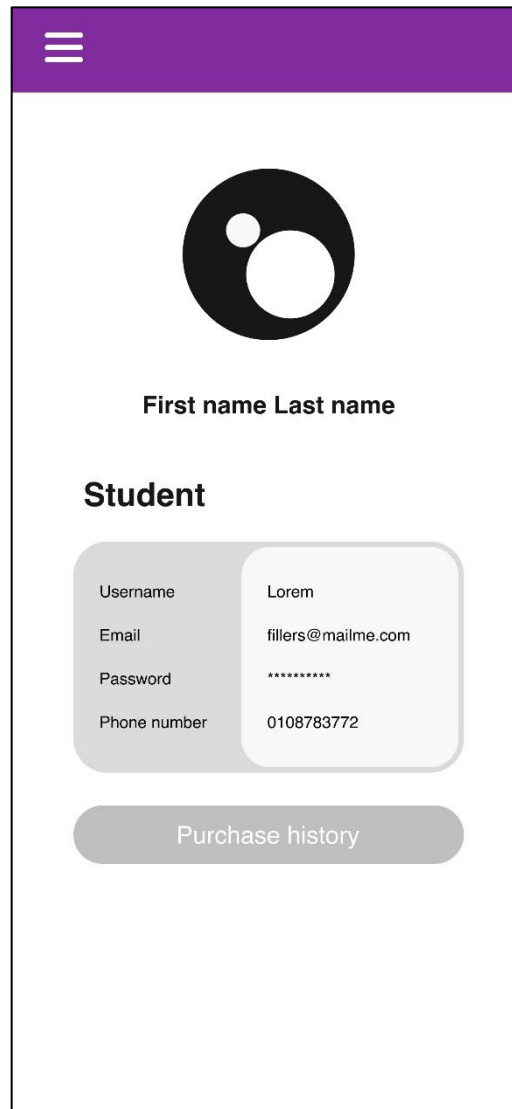


Figure 4.8: Student Profile User Interface

4.2.2.5 Class Information

This user interface provides the class details such as title, the coach's name, description, the title, duration and also number of lessons. The student user can request for a personal session in this user interface. Figure 4.9 displays the class information.

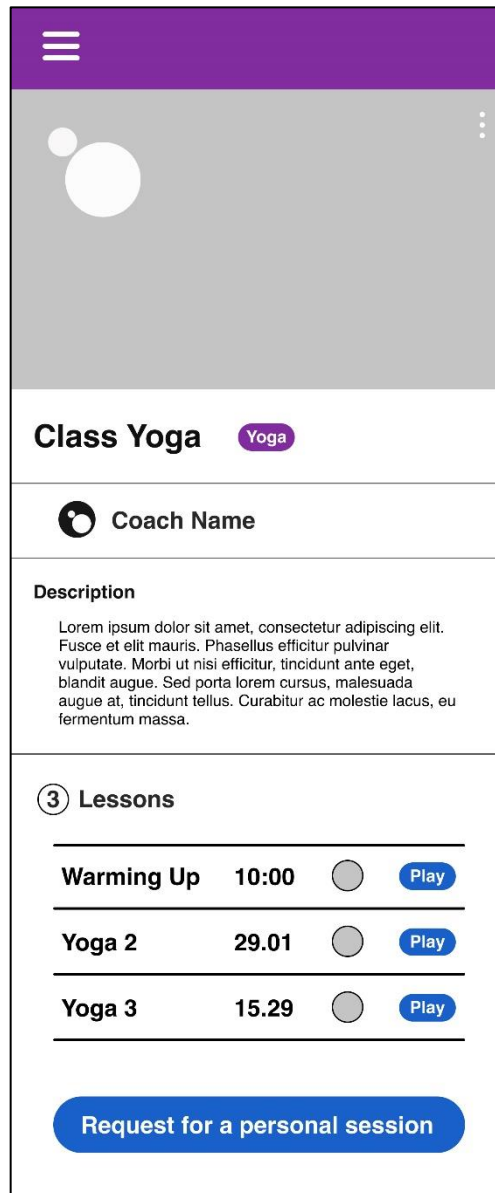


Figure 4.9: Class Information User Interface

4.2.2.6 Lesson Information

Here is the lesson information user interface whereby the students can watch the video lesson and also view details regarding the lessons. Figure 4.10 displays the lesson information which is related from the class.

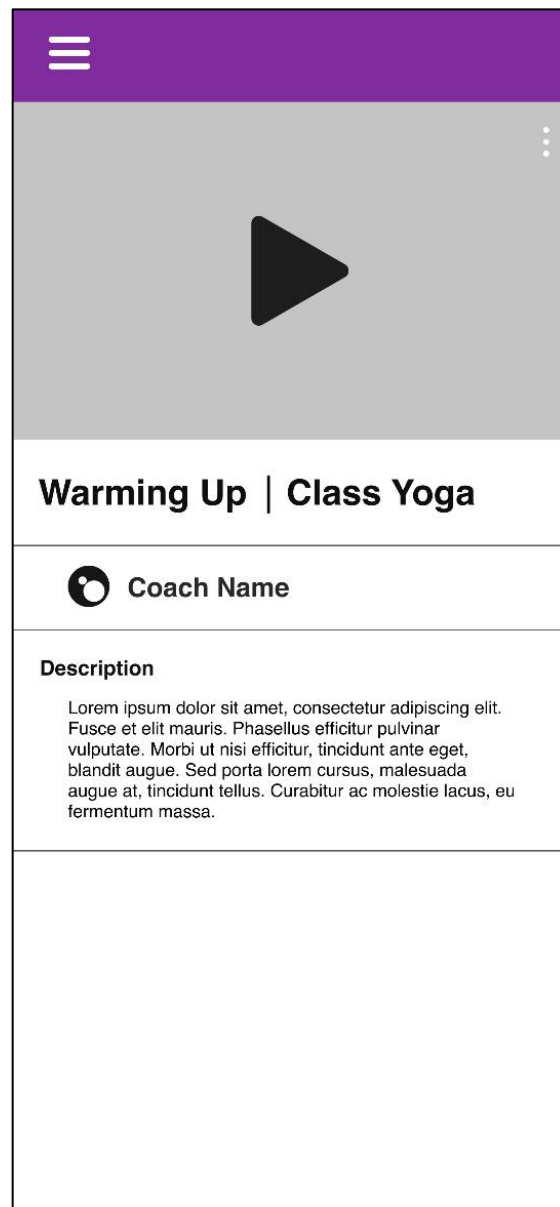


Figure 4.10: Lesson Information User Interface

4.2.2.7 Coach Profile

This user interface provides the necessary details of the coaches such as their name, username, certificate and email. The certificate section is made public for all users to view while all other details are made private. Figure 4.11 displays the coach profile.

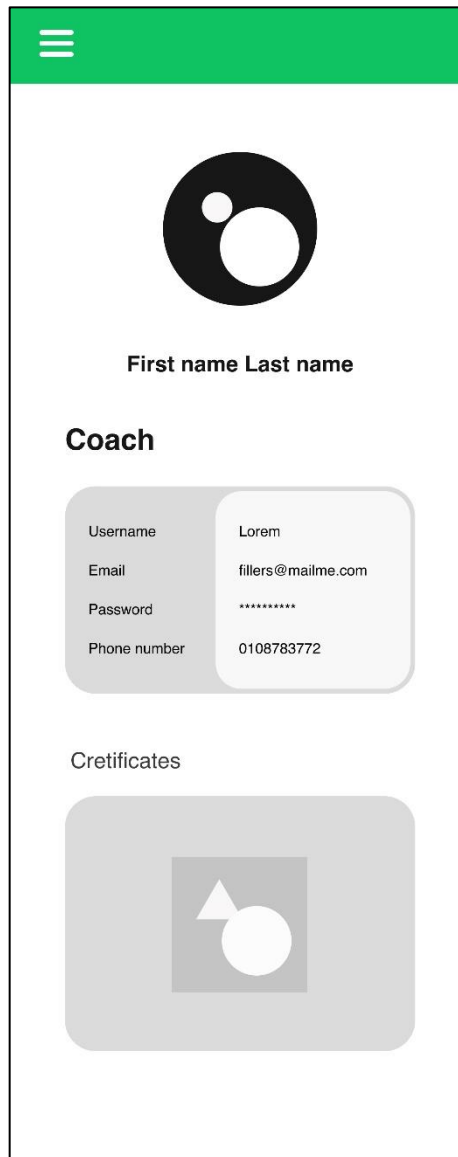


Figure 4.11: Coach Profile User Interface

4.2.2.8 Coach Homepage

This user interface shows the already created classes, such that the coach can manage and view the students enrolled within the class. Figure 4.12 shows the coach homepage user interface.

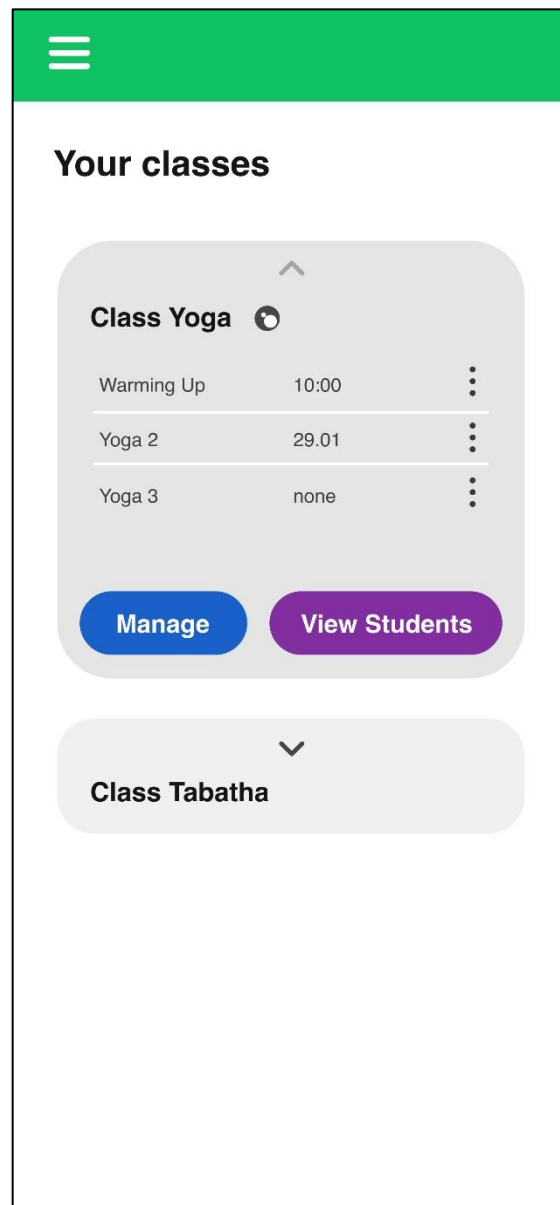
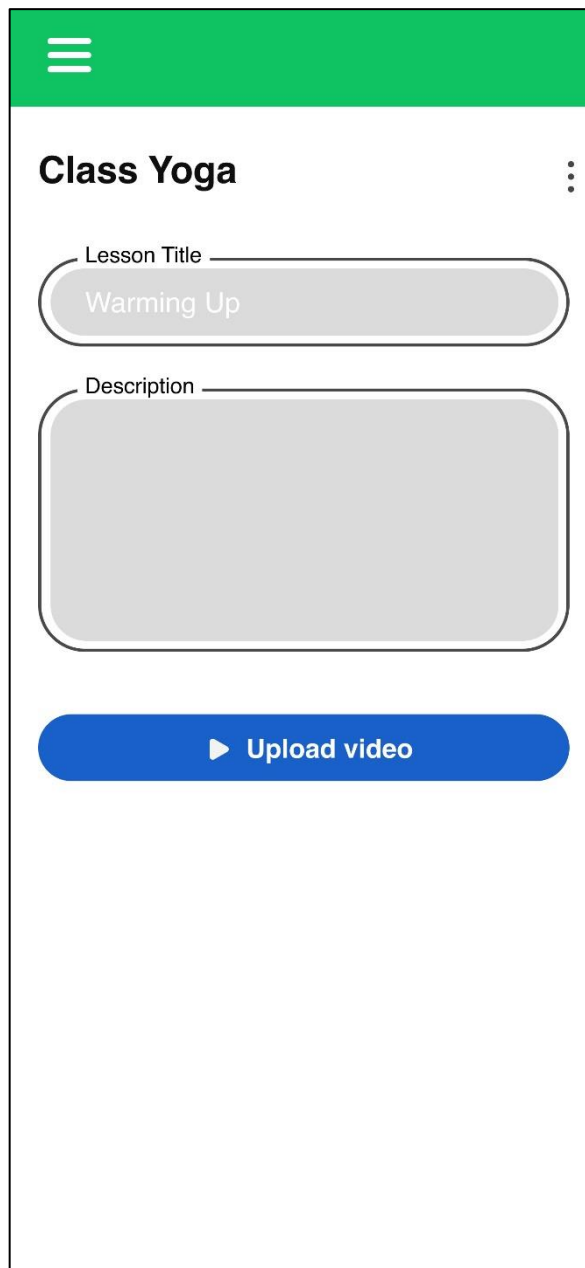


Figure 4.12: Coach Homepage User Interface

4.2.2.9 Lesson Creation

The coach can create lessons within this user interface by filling up the lesson title, description and upload a video. Figure 4.13 shows the lesson creation interface.



The image shows a mobile application interface for creating a lesson. At the top is a green header bar with a white hamburger menu icon. Below the header, the title "Class Yoga" is displayed in bold black text, followed by a vertical ellipsis menu icon. The form consists of three main sections: a "Lesson Title" field with a light gray rounded rectangle containing the text "Warming Up"; a "Description" field with a larger light gray rounded rectangle; and a blue rounded button at the bottom with a white play icon and the text "Upload video".

Figure 4.13: Lesson Creation User Interface

4.2.2.10 View Students

Figure 4.14 shows the view student's enrolled in a class page. The coach user can view the enrolled students within this user interface.

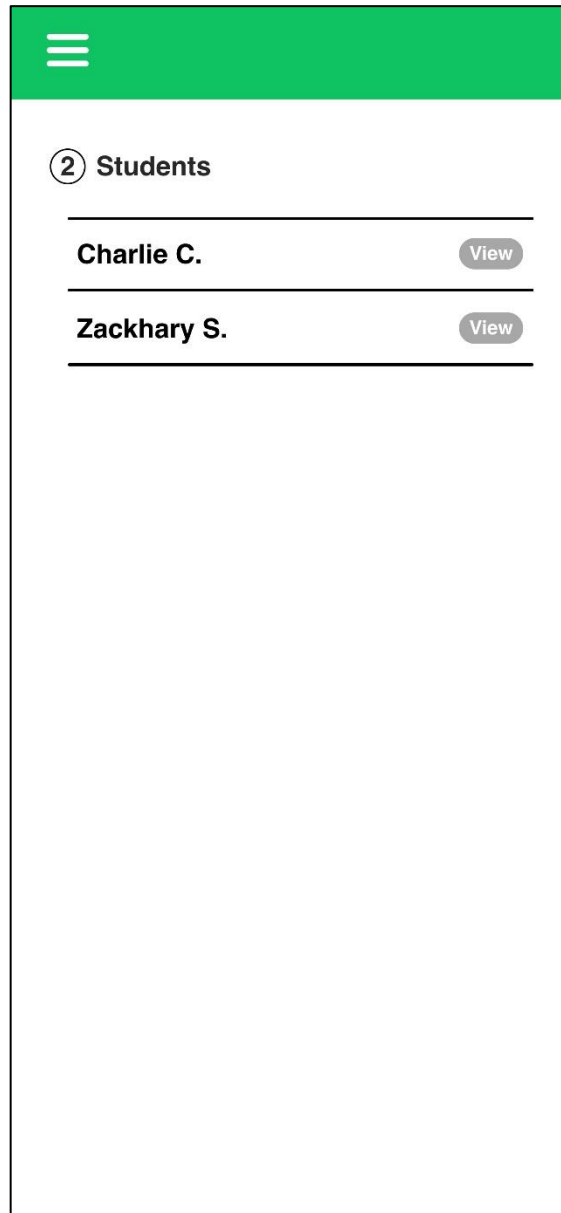
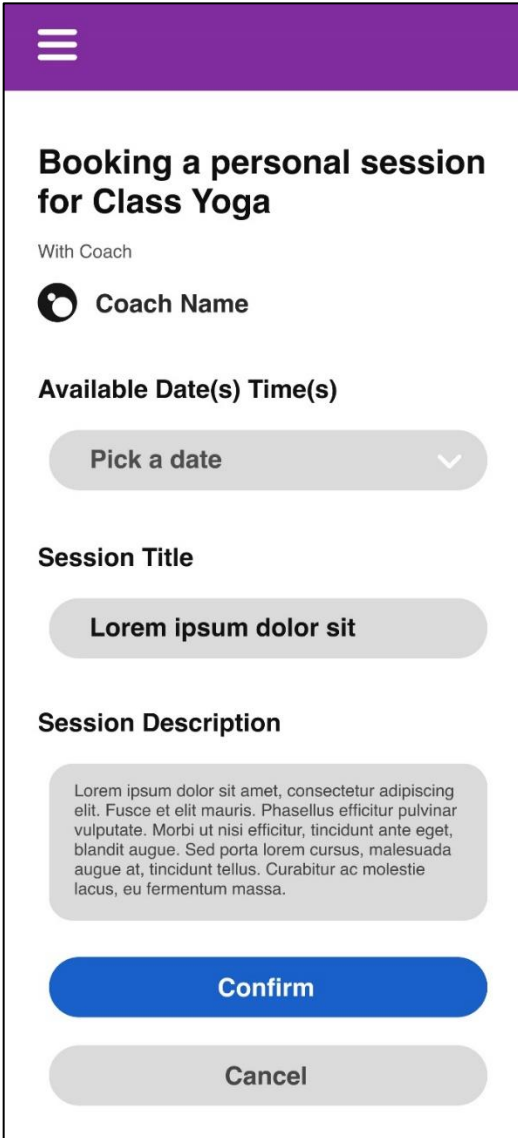


Figure 4.14: View Students User Interface

4.2.2.11 Personal Session Booking


The student user can book a personal session for a class by picking the available date and time set by their coaches, session title and session description. Figure 4.15 shows the personal session booking user interface.




The image shows a mobile application interface for booking a personal session. It features a purple header with a hamburger menu icon. The main content area is white and contains the following elements: a title 'Booking a personal session for Class Yoga', a subtitle 'With Coach', a coach profile icon and name, a section for 'Available Date(s) Time(s)' with a date picker, a 'Session Title' field with placeholder text, a 'Session Description' field with placeholder text, and two buttons at the bottom: 'Confirm' (blue) and 'Cancel' (gray).

Booking a personal session for Class Yoga

With Coach

 **Coach Name**

Available Date(s) Time(s)

Pick a date 

Session Title

Lorem ipsum dolor sit

Session Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce et elit mauris. Phasellus efficitur pulvinar vulputate. Morbi ut nisi efficitur, tincidunt ante eget, blandit augue. Sed porta lorem cursus, malesuada augue at, tincidunt tellus. Curabitur ac molestie lacus, eu fermentum massa.

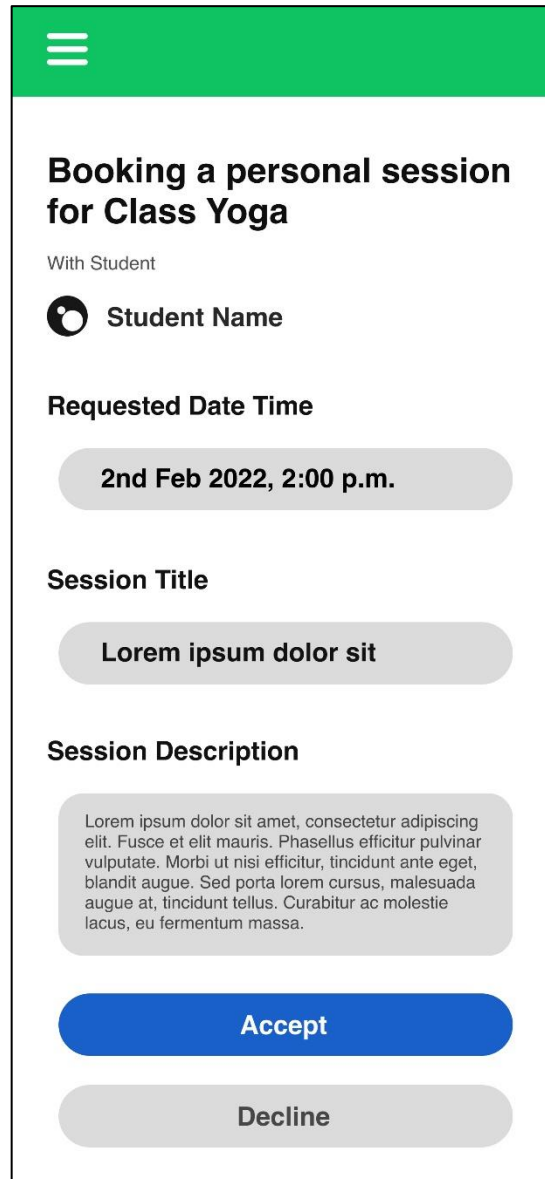
Confirm

Cancel

Figure 4.15: Personal session booking user interface

4.2.2.12 Personal Session Response


The coach can accept or decline a session requested by the student within this user interface. Figure 4.16 displays the personal session user interface.



The image shows a mobile application interface for responding to a session request. It features a green header with a white hamburger menu icon. The main content area is white and contains the following elements: a title 'Booking a personal session for Class Yoga', a subtitle 'With Student', a student profile icon and name, a 'Requested Date Time' field with the value '2nd Feb 2022, 2:00 p.m.', a 'Session Title' field with the value 'Lorem ipsum dolor sit', and a 'Session Description' field with a paragraph of Lorem Ipsum text. At the bottom, there are two buttons: a blue 'Accept' button and a grey 'Decline' button.

Booking a personal session for Class Yoga

With Student

 **Student Name**

Requested Date Time

2nd Feb 2022, 2:00 p.m.

Session Title

Lorem ipsum dolor sit

Session Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce et elit mauris. Phasellus efficitur pulvinar vulputate. Morbi ut nisi efficitur, tincidunt ante eget, blandit augue. Sed porta lorem cursus, malesuada augue at, tincidunt tellus. Curabitur ac molestie lacus, eu fermentum massa.

Accept

Decline

Figure 4.16: Personal Session Response User Interface

4.2.2.13 Personal Session

This user interface is where the personal session is held. Both student and coach user can interact by audio or video. Figure 4.17 displays the personal session user interface.

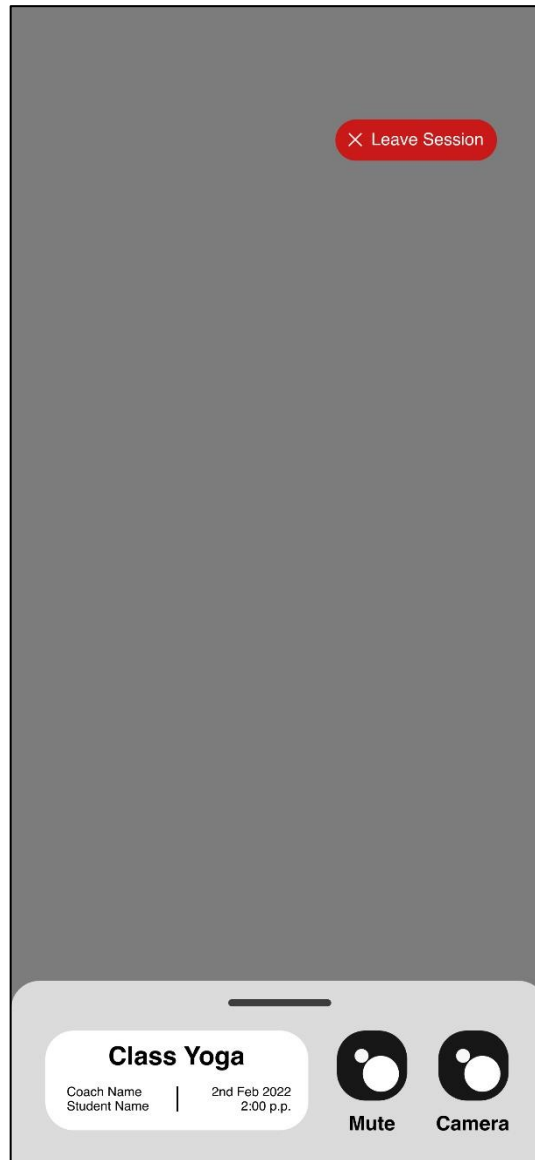


Figure 4.17: Personal Session User Interface

4.2.2.14 Administrator Dashboard

The administrator can only use web browser on desktop in order to use the system.

Figure 4.18 shows the administrator log in user interface.

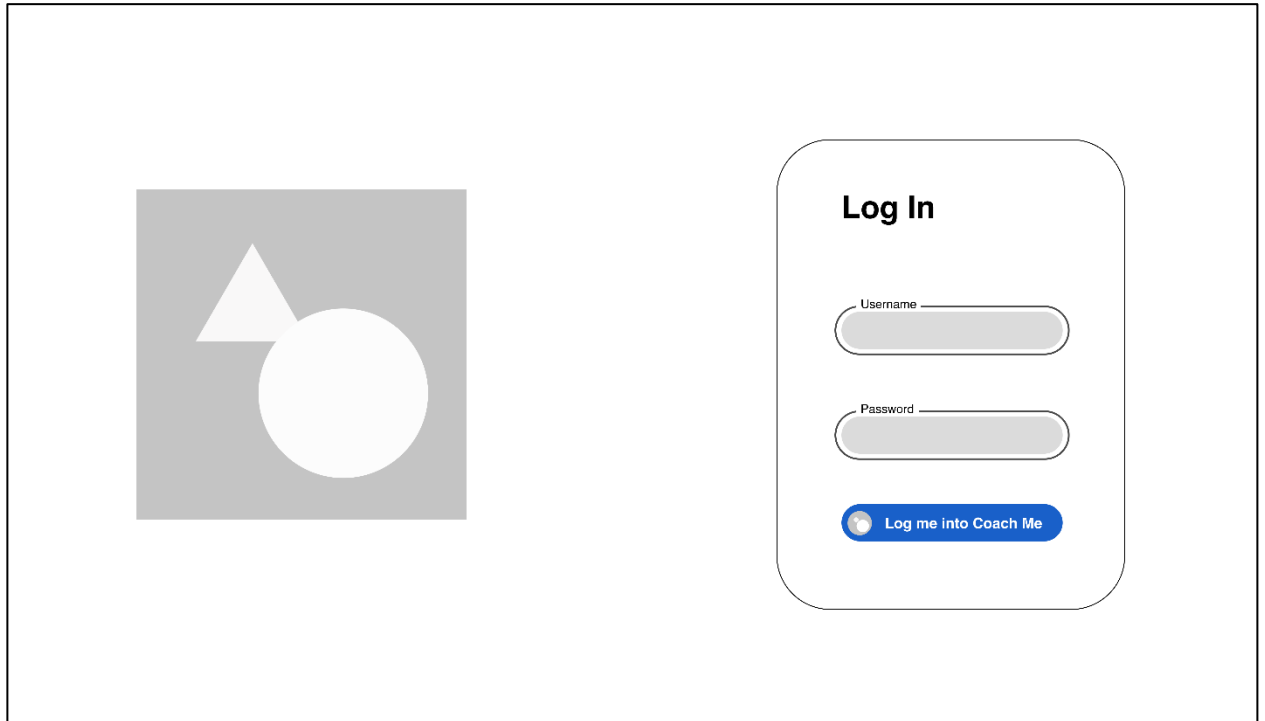


Figure 4.18: Administrator Log In User Interface

4.2.2.15 Administrator Log In

The administrator dashboard shows all the necessary actions for the administrator to perform. Figure 4.19 shows the administrator dashboard user interface.

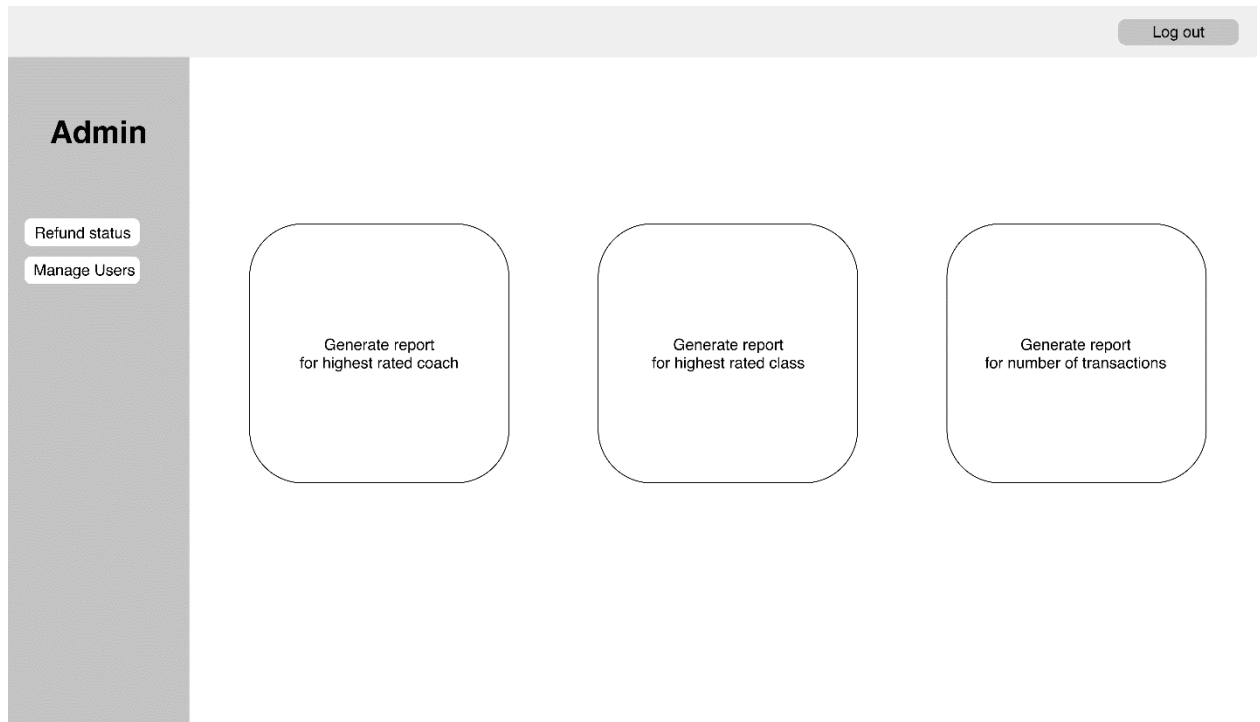


Figure 4.19: Administrator Dashboard User Interface

4.2.3 System Database Design

System design describes the overall structure or flow of the system including the function of the system. Indirectly, it will reduce the gap between the needs and the proposed system. It also transfers the proposed system functionality into the graphical diagram according to specific requirements. In this section, the system design of the proposed system is described. The database schema design is as follows: -

- i. **student**(user_id, username, name, email, telephone, user_type, age, bank_account)
- ii. **coach**(user_id, username, name, email, telephone, user_type, age, field, certificate, bank_account)
- iii. **admin**(user_id, username, name, email, telephone, user_type, age, admin_level)
- iv. **category**(category_id, name, class_id (FK))
- v. **class**(class_id, title, description, category_id (FK), coach_user_id, datetime_creation)
- vi. **lesson**(lesson_id, title, description, class_id (FK))
- vii. **video**(video_id, lesson_id (FK), title, mime, size, resolution)
- viii. **teach**(teach_id, coach_user_id (FK), class_id (FK))
- ix. **enroll**(enroll_id, class_id (FK), student_user_id (FK), datetime_enrolled)
- x. **payment**(payment_id, class_id (FK), coach_user_id (FK), student_user_id (FK), coach_bank_account (FK))
- xi. **receipt**(receipt_id, receipt, date_creation, payment_id (FK))
- xii. **personal_session**(personal_session_id, title, class_id (FK), coach_user_id (FK), student_user_id (FK), date_time)
- xiii. **session_status**(session_status_id, personal_session_id (FK), status)
- xiv. **report**(report_id, report_type, scale, report_date_time)

4.2.4 Data Dictionary

A Data Dictionary can be defined as the collection of names, definitions, and attributes about data elements that are being used or captured in a database, information system. There are 11 tables all together which make up the database of the system.

Table 4.12 shows the student data dictionary. It will store all the necessary records such as the username, name, email, password and etc.

Table 4.12: Student data dictionary

No	Constraint	Attribute	Type	Size
1.	PK	user_id	varchar	12
2.		username	varchar	24
3.		name	varchar	64
4.		email	varchar	64
5.		telephone	int	24
6.		user_type	varchar	12
7.		age	int	24
8.		bank_account	varchar	24

Table 4.13 shows the coach data dictionary. The coaches attribute such as certificate, age, field, telephone and name will be stored.

Table 4.13: Coach Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	user_id	varchar	12
2.		username	varchar	24
3.		name	varchar	64
4.		email	varchar	64
5.		telephone	int	24
6.		user_type	varchar	12
7.		age	int	24
8.		field	varchar	24
9.		certificate	blob	255
10.		bank_account	int	24

Table 4.14 shows the admin data dictionary. The administrator credentials will be stored as records.

Table 4.14: Admin Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	user_id	varchar	12
2.		username	varchar	48
3.		name	varchar	64
4.		email	varchar	64
5.		telephone	int	24
6.		user_type	varchar	12
7.		age	int	24
8.		admin_level	varchar	12

Table 4.15 shows the category data dictionary. Each class has their own category; thus the name of category and its class is stored to be accessed later by the backend Application Programming Interface.

Table 4.15: Category Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	category_id	varchar	12
2.		name	varchar	64
3.	FK	class_id	varchar	12

Table 4.16 shows the class data dictionary. The attributes such as title, description and category are store within the class table.

Table 4.16: Class Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	class_id	varchar	12
2.		title	varchar	64
3.		description	varchar	255
4.		category	varchar	64
5.		datetime_creation	date time	24

Table 4.17 shows the student data dictionary. The lesson table is related to the class by the class_id.

Table 4.17: Lesson Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	lesson_id	varchar	12
2.		title	varchar	64
3.		description	varchar	255
4.	FK	class_id	varchar	12

Table 4.18 shows the video data dictionary. The video is uploaded by using multi/form-part instead of the usual JSON; thus it will be stored in the database as binary large object.

Table 4.18: Video Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	video_id	varchar	12
2.		lesson_id	varchar	12
3.		title	varchar	64
4.		mime	varchar	24
5.		size	varchar	24
6.		resolution	varchar	24
		file	blob	255

Table 4.19 shows the enroll data dictionary. When a student enrolls in a class the data will be stored in this table.

Table 4.19: Enroll Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	enroll_id	varchar	12
2.	FK	class_id	varchar	12
3.	FK	student_user_id	varchar	12
4.	FK	datetime_enrolled	date time	24

Table 4.20 shows the teach data dictionary. When a coach teaches a class, the data will be stored within this table.

Table 4.20: Teach Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	enroll_id	varchar	12
2.	FK	class_id	varchar	12
3.	FK	coach_user_id	varchar	12

Table 4.21 shows the payment data dictionary. This payment table is to record the payment made by the student and store them for future uses.

Table 4.21: Payment Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	payment_id	varchar	12
2.	FK	class_id	varchar	12
3.	FK	coach_user_id	varchar	12
4.	FK	student_user_id	varchar	12
	FK	bank_account	varchar	12

Table 4.22 shows the receipt data dictionary. When the payment is made, receipt will be generated thus it is related with the payment table by payment_id.

Table 4.22: Receipt Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	receipt_id	varchar	12
2.		datetime_creation	date time	24
3.	FK	payment_id	varchar	12

Table 4.23 shows the personal_session data dictionary. Attributes such as the session title, who is attending the session is stored within this table.

Table 4.23: Personal_session Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	personal_session_id	varchar	12
		title	varchar	64
2.	FK	class_id	varchar	12
3.	FK	coach_user_id	varchar	12
4.	FK	student_user_id	varchar	12
		datetime	date time	24

Table 4.24 shows the session_status data dictionary. When a session get decline or accepted, the status is store in the table.

Table 4.24: Session_status Data Dictionary

No	Constraint	Attribute	Type	Size
1.	PK	session_status_id	varchar	12
2.	FK	personal_session_id	varchar	12
3.		status	varchar	12

4.3 Chapter Summary

The analysis and design phase are paramount in the development of the mobile application. UML diagrams were drawn to further understand the structure of the system. The interface was drawn as wireframe diagrams to further understand the flow of the mobile application. The database dictionary and schema depict the overall structure of how the database should be designed. Thus, in developing a system, the analysis and design phase is important.

REFERENCES

- Andreasson, Jesper, and Thomas Johansson. (2015). From exercise to “exertainment”. Body techniques and body philosophies within a differentiated fitness culture. *Scandinavian Sport Studies Forum* 6: 27–45.
- Andreucci, C. A. (2020). Gyms and Martial Arts School after COVID-19: When to come back to train?. *Advances in Physical Education*, 10(2), 114-120.
- Besard, T., Churavy, V., Edelman, A., & De Sutter, B. (2019). Rapid software prototyping for heterogeneous and distributed platforms. *Advances in engineering software*, 132, 29-46.
- Bourque, P., Dupuis, R., Abran, A., Moore, J. W., & Tripp, L. (1999). The guide to the software engineering body of knowledge. *IEEE software*, 16(6), 35-44.
- Bruegge, B., & Dutoit, A. H. (2009). Object--Oriented Software Engineering. Using UML, Patterns, and Java. *Learning*, 5(6), 7.
- Charland, A., & Leroux, B. (2011). Mobile application development: web vs. native. *Communications of the ACM*, 54(5), 49-53.
- Chinazzi, M., Davis, J. T., Ajelli, M., Gioannini, C., Litvinova, M., Merler, S., ... & Vespignani, A. (2020). The effect of travel restrictions on the spread of the 2019 novel coronavirus (COVID-19) outbreak. *Science*, 368(6489), 395-400.
- Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2012). *Non-functional requirements in software engineering* (Vol. 5). Springer Science & Business Media.
- Coronel, C., & Morris, S. (2016). Database systems: design, implementation, & management. Cengage Learning.

- Demirel, S. T., & Das, R. (2018, March). Software requirement analysis: Research challenges and technical approaches. In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)* (pp. 1-6). IEEE.
- Geoffrey H. T., Robert A., Peter K. (2020). Worldwide Effect of COVID-19 on Physical Activity: A Descriptive Study.
- Guida, G., Lamperti, G., & Zanella, M. (2013). Software Prototyping in Data and Knowledge Engineering (Vol. 497). Springer Science & Business Media.
- Hoekman Jr, R. (2010). *Designing the obvious: A common sense approach to Web & Mobile Application Design*. Pearson Education.
- Howley, Edward and Franks, B. Don. (2007). *Fitness Professional's Handbook*, p.ix. 5th Edition.
- Johnson, J. (2010). Designing with the mind in mind: Simple guide to understanding user interface design rules. Morgan Kaufmann.
- Karin A., Jesper A. (2021). Being a Group Fitness Instructor during the COVID-19 Crisis: Navigating Professional Identity, Social Distancing, and Community
- Laredo, E. J. B., Guzman, A. C. D., Manalo, N. P., & Alvarez, R. M. (2021). Monitoring Student Athlete's Personal Wellness and Sports Engagement Amidst COVID-19 Pandemic. *International Journal of Research in Engineering, Science and Management*, 4(8), 95–104. Retrieved from <https://www.journals.resaim.com/ijresm/article/view/1189>
- Lee, S. (2021). The Nike Training Club App. Retrieved 24 December 2021, from <https://medium.com/@by.sharonlee/the-nike-training-club-app-d4d2b028ff26>
- Lehman, M. M., & Belady, L. A. (Eds.). (1985). *Program evolution: processes of software change*. Academic Press Professional, Inc.
- Maguire, M., & Bevan, N. (2002, August). User requirements analysis. In *IFIP World Computer Congress, TC 13* (pp. 133-148). Springer, Boston, MA.
- Malan, R., & Bredemeyer, D. (2001). Functional requirements and use cases. *Bredemeyer Consulting*.
- McAuliffe Jr., T. (2021). Our Full Peloton Workout App Review: Pros, Cons... Is it Worth It?. Retrieved 24 December 2021, from

<https://www.laurelmdchiropractor.com/blog/our-full-peloton-workout-app-review-pros-cons-is-it-worth-it>

- Mulkey, Emily K and Neil, Elizabeth R and Cousins, Trisha A and Gibson, Mark H and Allison, Kaitlyn F and Lovalekar, Mita T and Murray, Mary E. (2021). *The Lived Clinical Experiences of Expatriate Athletic Trainers*. Master's Thesis, University of Pittsburgh.
- Nacheva, R. (2017, June). Prototyping approach in user interface development. In SECOND CONFERENCE ON INNOVATIVE TEACHING METHODS (ITM 2017) 28-29 JUNE 2017, VARNA (Vol. 28, p. 78).
- New York Post. (2021). Openfit app has a 4.95 star rating. Retrieved 24 December 2021, from <https://nypost.com/2021/07/02/openfit-app-has-a-4-95-star-rating-and-is-now-over-45-off/>
- Panhale, M. (2016). *Beginning hybrid mobile application development*. New York: Apress.
- Phongtraychack, A., & Dolgaya, D. (2018). Evolution of mobile applications.
- Pinho, C. S., Caria, A., Aras Júnior, R., & Pitanga, F. (2020). The effects of the COVID-19 pandemic on levels of physical fitness. *Revista da Associacao Medica Brasileira* (1992), 66Suppl 2(Suppl 2), 34- 37. <https://doi.org/10.1590/1806-9282.66.S2.34>
- Tudor-Locke, C., Ainsworth, B. E., Thompson, R. W., & Matthews, C. E. (2002). Comparison of pedometer and accelerometer measures of free-living physical activity. *Medicine and science in sports and exercise*, 34(12), 2045-2051.
- Wang, T., Gan, Y., Arena, S. D., Chitkushev, L. T., Zhang, G., & Rawassizadeh, R. (2021). Advances for Indoor Fitness Tracking, Coaching, and Motivation: A Review of Existing Technological Advances. *IEEE Systems, Man, and Cybernetics Magazine*, 7(1), 4-14.
- Weilkiens, T., Lamm, J. G., Roth, S., & Walker, M. (2015). *Model-based system architecture*. John Wiley & Sons.
- Wright C. J. Wright; Commentary on International Experience Opportunities for Athletic Training Students: A Perspective from the National Athletic Trainers' Association

International Committee. (2019). *Athletic Training Education Journal*; 14 (4): 255–258. doi: <https://doi.org/10.4085/1404255>

Young D. (2013). Software Development Methodologies. Journal White Paper Coronel, C., & Morris, S. (2016). Database systems: design, implementation, & management. Cengage Learning.

Appendix A

Gantt Chart

