

**ONLINE TUTOR MANAGEMENT SYSTEM
USING MOBILE APPLICATION (EZITUTOR)**

**MUHAMMAD KHAIRIL AKMAL BIN
MOHD KHAIRUDDIN**

**BACHELOR OF COMPUTER SCIENCE
(SOFTWARE ENGINEERING)**

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : _____

Date of Birth : _____

Title : _____

Academic Session : _____

I declare that this thesis is classified as:

- CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*
- RESTRICTED (Contains restricted information as specified by the organization where research was done)*
- OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

960710-08-5837

New IC/Passport Number

Date:

(Supervisor's Signature)

Dr. Ahmad Firdaus Bin Zainal
Abidin

Name of Supervisor

Date:

NOTE: * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons	(i)
	(ii)
	(iii)

Thank you.

Yours faithfully,

(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.



SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of *Doctor of Philosophy/ Master of Engineering/ Master of Science in

(Supervisor's Signature)

Full Name :

Position :

Date :

(Co-supervisor's Signature)

Full Name :

Position :

Date :



STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student's Signature)

Full Name : MUHAMMAD KHAIRIL AKMAL BIN MOHD KHAIRUDDIN

ID Number : 960710085837

Date :

ONLINE TUTOR MANAGEMENT SYSTEM USING MOBILE APPLICATION
(EZITUTOR)

MUHAMMAD KHAIRIL AKMAL BIN MOHD KHAIRUDDIN

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Computer Science (Software Engineering)

Faculty of System Computer and Software Engineering
UNIVERSITI MALAYSIA PAHANG

January 2018

ACKNOWLEDGEMENTS

Bismillah, in the name of Allah swt, the Foremost Gracious and the Most Merciful for giving me opportunity and strength in completing this final year project and other task. The health and time that given I use it wisely although there is some obstacle through the way, but I manage to complete this project with success among time, Allahamdulillah. Thank you also to my supervisor Dr Ahmad Firdaus Bin Zainal Abidin, my friend and my family and others that believe in me because give me advices and support in my life, during my studies and in documenting and developing this final year project. May Allah swt bless you all and grant you all in Jannah. Amin

ABSTRAK

Dengan kemajuan teknologi kini, telefon pintar menjadi suatu kemudahan yang dimanfaatkan sebaiknya. Justeru, banyak aplikasi yang dibangunkan untuk mencapai tujuan itu. Antaranya, Sistem pengurusan Tenaga Pengajar di atas talian (EziTutor) yang bertujuan membantu pelajar dan tenaga pengajar bertemu dan berinteraksi sesama satu sama lain untuk membuat kelas tuisyen. Aplikasi ini dibangunkan disebabkan oleh terdapat beberapa masalah yang sering dihadapi masyarakat di Malaysia terutamanya ibu bapa para pelajar sekolah iaitu mereka terpaksa membayar harga yang mahal untuk menghantar anak-anak mereka untuk ke pusat tuisyen atau kelas tambahan. Hal ini membebankan ibu bapa yang ingin melihat anak-anak mereka cemerlang seperti anak orang lain. Selain itu, masalah yang timbul adalah kualiti tenaga pengajar disesetengah pusat tuisyen tidak memuaskan. Hal ini kerana tenaga pengajar tersebut tidak mengetahui dan mengikuti perkembangan silibus yang diajar di sekolah. Ada juga yang mengajar hanya untuk mencari wang saku tanpa mengira ilmu yang disampaikan itu difahami pelajar atau tidak. Oleh yang demikian, pengorbanan ibu bapa para pelajar tampak penuh dengan sia-sia dan tidak mendatangkan manfaat pada anak mereka. Masa dan jarak juga menjadi faktor mengapa para pelajar mempunyai masalah untuk ke kelas tuisyen. Hal ini adalah kerana jadual kelas sudah ditetapkan oleh pihak tuisyen dan tidak dapat diubah mengikut kesesuaian pelajar tersebut. Untuk pelajar yang jauh daripada pusat tuisyen juga menjadi masalah berulang-alik ketempat tersebut. Masalah lain ialah tiada alternatif lain untuk para pelajar mengetahui siapa yang menyediakan perkhidmatan kelas tambahan atau tuisyen diluar kawasan mereka kecuali pusat tuisyen. Oleh sebab masalah-masalah ini aplikasi EziTutor dibangunkan. Aplikasi ini dapat mengurangkan kos pelajar membayar harga tuisyen tersebut kerana tidak ada campur tangan daripada pihak pusat tuisyen. Aplikasi ini dapat menunjukkan lokasi tenaga pengajar yang ingin membuat kelas tuisyen yang berhampiran dengan pelajar tersebut. Pelajar dapat membandingkan harga setiap kelas tuisyen dan mendaftar nama di kelas tersebut. Notifikasi akan diberikan kepada tenaga pengajar. Dengan adanya aplikasi ini diharap dapat membantu pelajar menjadi pelajar yang cemerlang khusus keluarga pelajar yang berpendapatan rendah. Objektif projek ini adalah untuk membangunkan aplikasi berasaskan mudah alih, dan membuat peningkatan ciri-ciri aplikasi pembelajaran sedia ada. Kaedah Rapid Kitaran Pembangunan Aplikasi (RAD) telah dipilih sebagai pembangunan perisian metodologi untuk projek ini.

ABSTRACT

With today's technological advances, smartphones have become a useful facility. Hence, many applications are developed to achieve that goal. Among them, the Online Tutor Management System (EziTutor) which aims to help students and Tutors meet and interact with each other to make tuition classes. This application was developed due to the problems faced by many people in Malaysia especially the parents of that have children in school that they had to pay an expensive price to send their children to a tuition center or an extra class. It is a burden to parents who want to see their children excel like other children. In addition, the problem arises is the quality of the teaching staff and materials is irrelevant. This is because the instructors are not aware of and follow the syllabus taught in schools. There are also those who teach only to find some money regardless of is the knowledge that the student understands or not. As a result, the sacrifice of parents of students seems to be futile and not beneficial to their children. Time and distance are also factoring why students have problems with tuition classes. This is because the class schedule has been set by the tuition and cannot be changed according to the suitability of the student. For students who are away from the tuition center, it is also a matter of commuting to the place. Another problem is that there is no other alternative ways for students to know who provides extra classes or tuition services outside their area except tuition centers. Because of these problems the EziTutor app is developed. This app can reduce the cost of the student to pay the tuition rate because there are no extra charges from the tuition center. This app can detect the location of the tutor who wants to make a tuition class that located near to the student. Students can compare the price of each tuition class and register the name in the class. Notifications will be given to tutor. With this application hopefully, it will help students to be excellent students especially for low-income families. The objective of this project is to develop mobile-based applications, and to improve the features of existing learning applications. Rapid Method of Application Development Cycle (RAD) has been selected as a software development software for this project.

TABLE OF CONTENT

DECLARATION	
TITLE PAGE	
ACKNOWLEDGEMENTS	ii
ABSTRAK	iii
ABSTRACT	iv
TABLE OF CONTENT	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
LIST OF SYMBOLS	x
LIST OF ABBREVIATIONS	xi
CHAPTER 1 INTRODUCTION	1
1.1 Background of study	1
1.2 Problem Statement	2
1.3 Objective	4
1.4 Scope	4
1.5 Significant	5
1.6 Thesis Organization	5
CHAPTER 2 LITERATURE REVIEW	7
2.1 Introduction	7
2.2 Mobile Platform	7
2.3 Differentiate Between Native App Development and Hybrid App Development	8

2.4	Existing Application on Tutor management system	10
2.4.1	Snapask: Instant Homework Help	10
2.4.2	Cambly	11
2.4.3	EMT Tutor NREMT-B Study Guide	13
2.5	Comparing the existing application and current application	15
2.6	Conclusion	16
CHAPTER 3 METHODOLOGY		17
3.1	Introduction	17
3.2	Methodology	17
3.2.1	Rapid Application Development (RAD)	18
3.2.2	Propose User Interface	21
3.2.3	General architecture and package module	24
3.2.4	Software Requirement Specification (SRS)	26
3.2.5	Software Design Document (SDD)	26
3.3	Hardware and Software Tools	26
3.3.1	Hardware Tools	26
3.3.2	Software Tools	27
3.4	Gantt Chart	27
3.5	Testing Plan	28
CHAPTER 4 RESULTS AND DISCUSSION		30
4.1	Introduction	30
4.2	Implementation Data/Model/Process/Hardware	30
4.2.1	Implementation	30
4.2.2	Implementation Testing	36

4.3	Testing and Result Discussion	37
4.4	User Manual	37
CHAPTER 5 CONCLUSION		38
5.1	Introduction	38
5.2	Advantages	38
5.3	Constraints	39
5.4	Future Work	39
5.5	Conclusion	40
REFERENCES		41
APPENDIX A		42
APPENDIX A(A) sequence diagram		57
APPENDIX B		62
APPENDIX C		66
APPENDIX D		75

LIST OF TABLES

Table 1.1 show the Summary of Problem Statement	3
Table 2.1 show comparison between native application developments vs. hybrid platform application	9
Table 2.2 show Comparison between three existing applications and EziTutor	15
Table 3.1 show detail of hardware use and its specification	26
Table 3.2 show detail of software use and its specification	27
Table 3.3 show the example of the test plan	28

LIST OF FIGURES

Figure 2.1 show GUI for Snapask	10
Figure 2.2 show GUI for Snapask	11
Figure 2.3 shows GUI for Cambly for android	12
Figure 2.4 shows GUI for Cambly for Web-based	13
Figure 2.5 show GUI for EMT Tutor	14
Figure 2.6 show GUI for EMT Tutor	14
Figure 3.1 show RAD phases	18
Figure 3.2 show use case diagram for EziTutor	19
Figure 3.3 show context diagram for EziTutor	20
Figure 3.4 show create class page	21
Figure 3.5 show panel	22
Figure 3.6 show manage profile	22
Figure 3.7 show Tutor registration	23
Figure 3.8 show search tutor	23
Figure 3.9 show Basic Architecture for EziTutor	24
Figure 3.10 show package module for EziTutor	25
Figure 3.11 show Gantt chart for EziTutor development by using RAD method	28
Figure 4.1 show Class Welcome	31
Figure 4.2 show Class Login	32
Figure 4.3 show Class Register	33
Figure 4.4 show Class Location	34
Figure 4.4 show Class for Create Class	35
Figure 4.5 show Class for Create Event	36

LIST OF SYMBOLS

LIST OF ABBREVIATIONS

RAD	Rapid Application Development
EziTutor	Online Tutor Management System
MQA	Malaysia Qualification Agency
OS	Operating System
PDA	Personal digital assistant.
GUI	Graphical User Interface
SDLC	Software Development Life Cycle

CHAPTER 1

INTRODUCTION

1.1 Background of study

In the civilization era of information and communication system, people are easily getting information and connect with each other using their smartphone. This happen because it gets support with new and rapidly growing Mobile application sector. This growing sector can give global positive impact or negative impact to its users. Many developers created android application and iOS application that focus on entertainment such as games, music, social media and other to its mobile application user but few of them created mobile application that can help people in their daily routine or in life. For example, Google Maps and manage to help people to travel from one destination to other destination in short distance as possible. Moreover, this help the people make calculate of budget before travelling and can save cost and time. Thus, the role of mobile application is big enough to give impact to society and change the way how we are living.

This EziTutor mobile application use the function of Google Maps and implement at different area which is education in Malaysia. The purpose of this application mainly to help student especially from poverty family to have a better education by providing a platform for student and tutor meet each other. Currently, tuition centre is the ideal place for both student and tutor to meet each other but there are many issues regarding the tuition centre such as highly fees. Thus, student from poverty family cannot afford to have tuition.

EziTutor mobile application is used android as it operating system. This application shows how many tutors available, the distance and cost so that student can select their preferable choice. By doing this student from poverty family can select the lowest fees to have tuition with a standard level of quality tutor. Only qualify people can

be register as tutor in this application by selecting based on their resume and certificate for safety and quality issue. Hopefully this application can change the way of education in Malaysia.

1.2 Problem Statement

With the success development and growth of economic and technology in Malaysia, giving the best education to the people is the main factor behind of it. It should start with our children in their kindergarten, primary and secondary school. Moreover, this new generation is our hope to lead the country in the future. Thus, everyone must take responsible to take care of this new generation especially parents and tutors. One thing that parents that can do to improve and help the education of their children is by sending their children to tuition either conducted by the school itself of outside of the school. (Labour & Survey, 2015)

The tuition centre is place where the students are usually taught how to tackle examination question. Thus, this is the most preferable place to send their children to gain more knowledge and excel in examination among parents in Malaysia. Unfortunately, there are several weakness and issue regarding the tuition centre. First, the issue is not all parents have the financial ability to support their children to get this extra education. The charging fees of tuition centre is expensive because there are many individuals involve which are tutors, staff of tuition centre and the owner of the place. The solution to this problem currently is to gain more financial income by doing more jobs or part time jobs. This become a burden especially poverty family. (Renehan, 2015)

Second, the students do not know who will be teaching them because the tutors are provided by the tuition centre. They do not have flexibility to choose the tutor who are teaching the subject. Furthermore, there is tutor that provided by the tuition centre teaching the subject not in their niche area and they not know the current school syllabus. The tuition centre lack of experience tutor but in same times still doing the business because of money. Thus, the student will not achieve the good result in their examination because this extra class does not give any added value and knowledge to them.(Lumpur, 2016)

Lastly, tutor or people who want to do part time job for teaching are depend with the tuition centre. Tutor need to wait if the tuition centre is not available for them and if they want to do the tuition by their own it is difficult for them to get student. This is because there is no platform provided for gathering the student except the tuition centre.

Table 1.1 the Summary of Problem Statement

Problem	Description	Effect
The tuition centre fees expensive	There are many parties involve such as the owner of the place, tutor and staff that organize the tuition centre	Parent need to find other income to pay this fee especially student that come from poverty family
Unqualified or inexperience tutor	The tutor provide by the tuition centre is not from education background or not expert in the subject that they teach	Student cannot understand what the tutor teaches and will affect the examination
Student and tutor do not have contact each other	The student wants to find the best tutor for the subject and tutor need gather student that want learn the subject, but both cannot contact each other	Student and tutor need to depend on tuition centre to learn and teach the subject respectively

1.3 Objective

The objective of this project is to develop a mobile application that provide platform for student and tutor connect with each other conveniently known as Online Tutor Management System (EziTutor). The details objectives of the project are described as below:

- i. To identify the requirement that useful for the EziTutor.
- ii. To develop EziTutor using hybrid application.
- iii. To test the EziTutor functionality based on user requirement

1.4 Scope

The scope of this project is discussing on boundary and users. The scope details of the project are described as follows:

- i. Scope Objective
 - a. There are things that must bear in mind to achieve the goal of the project when develop an application that will provide access control though mobile which are:
 - user friendly : Can be use by all people and easy to learn the flow of the system
 - conveniently : This system is not complex, and the user do not need any training to become expert
 - feasibility : Able to give accurate result by system function
- ii. User
 - a. Tutor: All graduated people with university level of education.
 - b. Student: Primary, secondary and high level of education but just few will be selected as it is beta version and for testing purpose only.
 - c. Admin: Selected by EziTutor organization staff.
 - d. Client: EziTutor Company
- iii. Boundary

- a. This system will use dummy data and only available within Universiti Malaysia Pahang area for testing purpose.

1.5 Significant

The project is significant to the society and it can change the future. The project significant are described as below:

- i. The project reduces the fees of tuition by eliminates third party which is tuition centre.
- ii. The project gives flexibilities to student to choose their tutor based on qualification, cost and distance`.
- iii. The project provide platform for qualified people that want to be tutor advertise their availabilities to student that need.

1.6 Thesis Organization

Generally, there are total six chapters in discuss in this thesis paper. The chapter details are as follow:

Chapter one discusses the introduction and identification of the mobile application to be develop. This chapter consist of introduction about the system, problem statement in real life situation, objective to develop this system, scope of project overall and thesis organization to briefing what flow on this report.

Chapter two discuss the literature review regarding the mobile application. It includes the review of other mobile based application, the advantages of the system and its advantages.

Chapter three discuss concerning an overall of methodology are implementing to developing the system. This project can develop by exploitation agile software package Development Life Cycle methodology. It additionally includes the approach accustomed develop the applying. Moreover, discuss on analysis and style of the project supported

the need. Overall system should be represented by the developer and therefore the content of this chapter is said by the analysis and style to make a project. the planning should have relation between the info and knowledge collected. The developer ought to show the flow on a way to ensure the system is developed.

Chapter four discuss the implementation of the project, beside the constrains of the system. Developer check the system to judge the practicality and effectiveness of the system. Testing part is completed during this chapter once system already developed. This part can perform testing on every module and carried the result and knowledge analysis. this can be vital to form positive the testing part meet the whole project objective and demand.

Chapter five discuss concerning overall conclusion concerning this project. The outline can conclude the benefits and quality concerning the system. additionally, the longer-term upgrade for this technique can depend upon the effective and economical use by client.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter will cover the literature review on existing online tutor system whether it is web based, mobile based or standalone that has already been used in the market which covers the feature provided, advantage, disadvantages of the application. Any gaps can be identified when analyse the existing works which can be addressed in the proposed application known as Online Tutor management system using mobile application

Education plays a big role for a people to success in their life. The jobs vacancy offers by huge company in Malaysia have their own standard level of education such as diploma, degree and other that certified by Malaysia Qualification Agency (MQA) or certified by Public Service Department (PSD). The student nowadays very competitive among each other and want to become the best among the best. Thus, ease to continue their studies in University and can have better jobs.

An Online Tutor Management using mobile application (EziTutor) is created to help people especially student gaining more knowledge by providing a platform to search the best Tutor based on their qualification, payment and distance. EziTutor also provide function for Tutor to manage the environment of study that they prefer such as the place, type of tutoring and limit number of students.

2.2 Mobile Platform

The mobile platform is the mobile operating system (OS) which is running on mobile devices, for example tablets, smartphones and personal digital assistant (PDA).

The hardware of mobile devices is installing with mobile OS to make its functioning. There are three type of mobile OS that are common to user and widely used which is Android, iOS and Windows. Without a mobile OS, all the mobile applications cannot run on the smartphone. It is the core software in a mobile device. There are three advancement phases involving the mobile OS such as PC-based OS, embedded OS and the latest one which is present-day smartphone-oriented OS.

2.3 Differentiate Between Native App Development and Hybrid App Development

A native application is a program that has been coded in a programming language, for gadgets having OS. For example, Objective C is utilized for iOS or Java for Android Phones. These applications can exploit OS highlights and other programming devices that are introduced on that platform. They also have access different telephones function, for example, camera, sound, contact list and so forth. For the most part Native applications are ordered into machine code.

A hybrid application is a program that is constructed utilizing HTML 5, CSS and JavaScript and wrapped in native compartment. The native compartment loads maximum data on the page when the client explores and navigates the application. There are a lot of local web systems accessible, which can enable you to fabricate a hybrid application.

Clarifications about advantages and limitation of each and design which approach ought to be chosen relying upon the versatile experience and usefulness intend to convey and furthermore take an analysis at how these methodologies can be joined to convey the most ideal experience inside tight requirements to the user. Table 2.1 below show the comparison between these two platforms. (Multidots, 2014)

Table 2.1 comparison between native application developments vs. hybrid platform application

NATIVE APP DEVELOPMENT	CRITERIA	HYBRID APP DEVELOPMENT
Long	Time of development	Short
Full support	Can access all hardware on the device	Partial support
Excellent	User experience and responsiveness. Speed of interaction	Poor
Full support	Used of latest functions of device	Not support
More expensive	Cost of the development	Less expensive
Full support	Developers can use the benefit of new or open source software libraries	Not support
High	Reliability	Low
Support online and offline	Transactions	Support online
Not Support	Integrate with device file system	Support

2.4 Existing Application on Tutor management system

2.4.1 Snapask: Instant Homework Help

Snapask is online tutoring application that founded by Timothy Yu in January 2015 to connect students to tutors for homework help and academic support allows every student ask question anytime and anywhere. Snapask is mobile application that helps you get personalized learning help from great tutors and encourages you to build learning habit. Thus, help you overcome challenges with confidence. Snapask functions like a mobile crowdsourcing platform that involve artificial intelligence, in which students are required to snap a photo to their queries and submit it to the platform to be addressed by tutors. To use this application, user need to register their profile first and select their category either student or tutor. The level of education for student category limited to primary school and secondary school only. Snapask application is a free mobile application and available for iOS and Android. Figure 2.1 and Figure 2.2 shows the graphical user interface (GUI) for Snapask mobile application. The details of features of this system can be refer to Table 2.2. (Snapask, 2018)

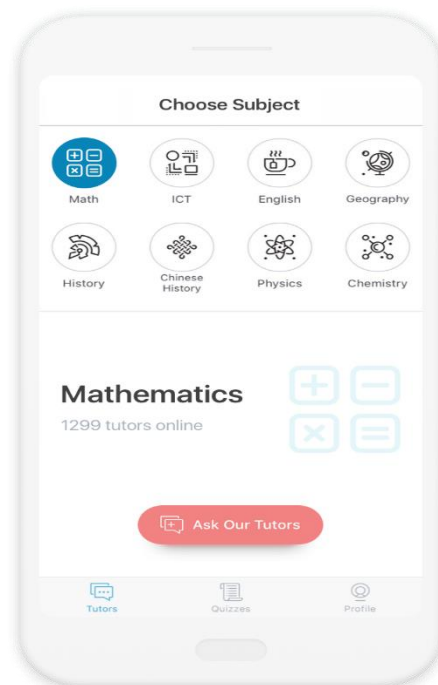


Figure 2.1 GUI for Snapask

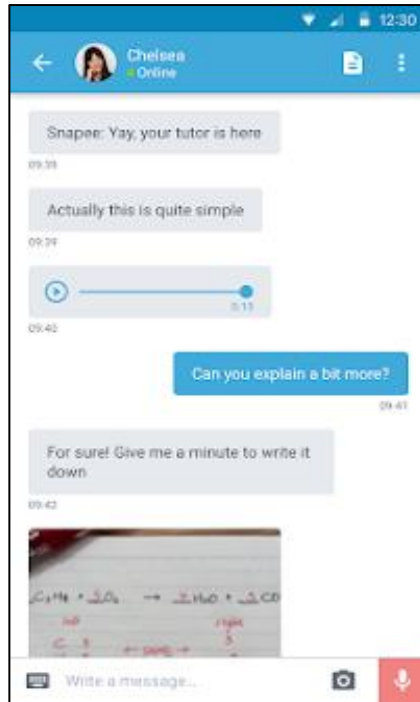


Figure 2.2 GUI for Snapask

2.4.2 Cambly

Cambly is an online English teaching and tutoring service that connects tutors to students all over the world through their own video chat platform. These apps do not require the people how want to apply as tutor any prior teaching experience and it's a great way to earn money online. Cambly tutors are dynamic native speakers from the United States, Canada, UK and Australia, etc. User can browse tutor profiles to find tutors with relevant experience or interests. Cambly tutors have a wide range of education and professions. User can also search for interesting topics to read and discuss with your tutors User also can access Cambly from your mobile device, or from a computer with a web camera. Furthermore, this application connects with user perfect tutor in under 5 seconds by simply pressing a button. This app provides as platform to practice English conversation, take an English course, or learn to ace the International English Language Testing System (IELTS) or Test of English as a Foreign Language (TOEFL). There is function in Cambly that can learn with a private tutor for customized classes. It pays \$0.17 per minute spent chatting, which adds up to just over \$10 per hour. A tutor will earn \$20 if new tutor signs up with Cambly link that provided by another tutor and have tutor 10 priority hours. This application provides a Video Recording capability for user

to record video of every class and allow to watch and re-watch every new vocabulary or concept user learn. Reservations function in this application let user book classes with their favourite tutors, so user always get the tutor they want at the time user want. Cambly application is a free mobile application and available for iOS and Android. Figure 2.3 shows GUI for Cambly mobile application and figure 2.4 show GUI for web-based application. The details of features of this system can be refer to table 2.2. (Cambly, 2018)(Babeesh, 2018)

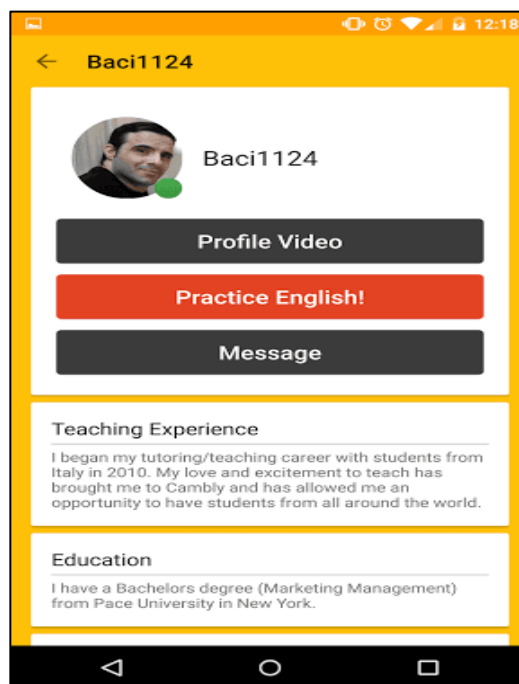


Figure 2.3 GUI for Cambly for android

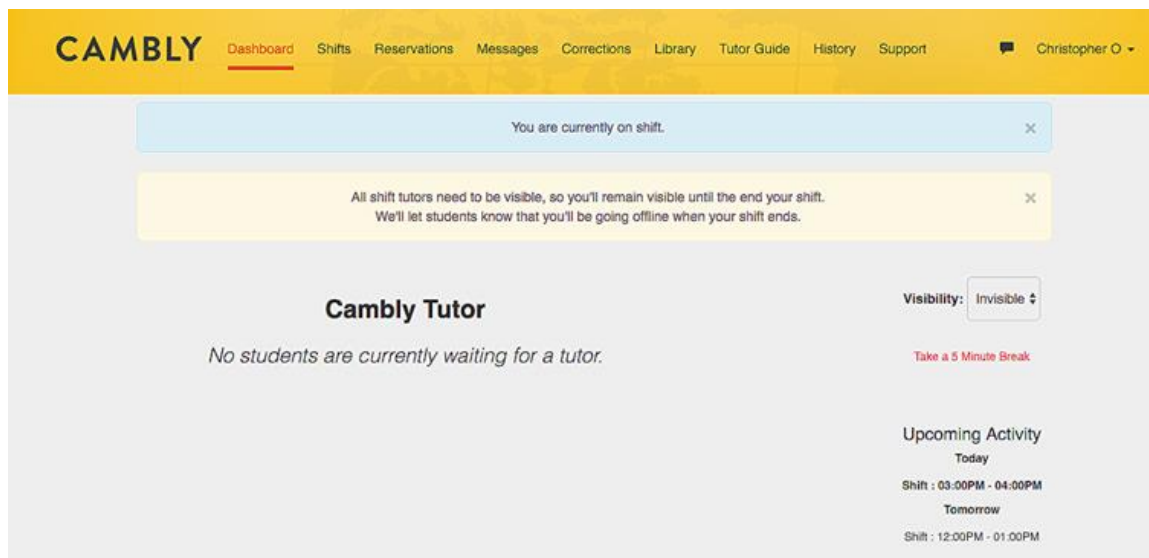


Figure 2.4 GUI for Cambly for Web-based

2.4.3 EMT Tutor NREMT-B Study Guide

The EMT tutor could be a comprehensive coaching tool for the Emergency Medical Technician (EMT) Basic. This application is made by Code3Apps company that develops mobile apps for Emergency Responders to reinforce their coaching and development as skilled crisis managers. during this application there, square measure over a thousand doable quiz queries and flashcards on the market. The functions on the market is Bookmarking feature that enables the user to save lots of tough queries, flashcards, or eventualities for later study. this permits the user to tailor the expertise to their wants. Explanations section so the user learns from each question and placard. Over a thousand Quiz queries and a thousand Flashcards ready by Emergency Medical Services (EMS) instructors that square measure concerned in each EMS education and running calls within the streets. during this application even have a scenario-based section to help in taking text book material from page to street. Over fifty eventualities square measure given here, with additional to come back, that a user will undergo on their own or with a field pedagogue to stimulate the thinking processes that occur on a 911 decision. Material that has been developed by EMS educators that square measure current on latest National written account of Emergency Medical Technicians paramedical developments. EMT Tutor application could be a free mobile application and on the market for iOS and mechanical man. Figure 2.5 and Figure 2.6 shows GUI of EMT Tutor for mechanical

man mobile application. the small print of options of this method will be talk to Table 2.2. (code3apps, 2013)

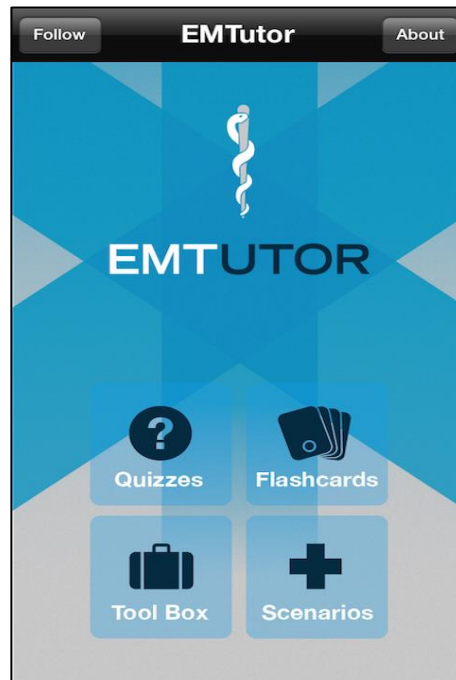


Figure 2.5 GUI for EMT Tutor

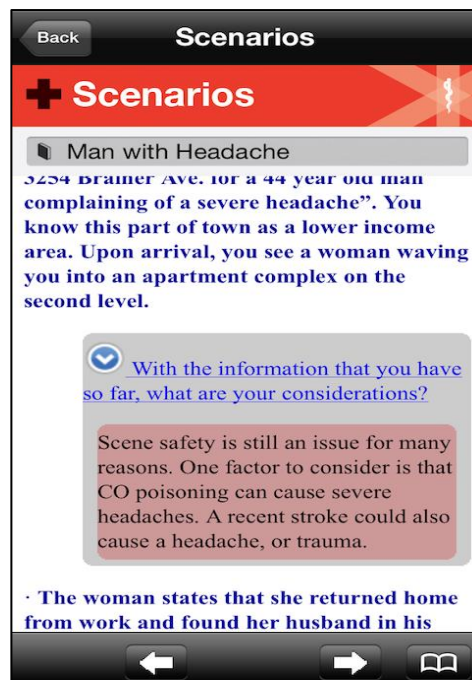


Figure 2.6 GUI for EMT Tutor

2.5 Comparing the existing application and current application

This topic will discuss the comparison between three existing application. Between these three existing applications, Snapask is a suitable application compared to Cambly, EMT Tutor because Snapask can access to Android, iPhone, and Web-based application and for student that need tutor this app is the best choice compare too Cambly and EMT Tutor that just focus on certain subject. Snapask also has the interactive and simple interface that can guide the user to use this application friendly.

Table 2.2 Comparison between three existing applications and EziTutor

Name	Snapask	Cambly	EMT Tutor	EziTutor
Access to device	Android, iPhone, web	Android, iPhone, web	Android, iPhone	Android
Purchased	No	No	Yes	No
Offline capabilities	No	No	No	Yes
Provide Location for tutor	No	Yes	No	Yes
Provide customize subject	Yes	No	No	Yes
Provide platform for student and tutor to register	Yes	Yes	No	Yes
Provide platform to earn income	Yes	Yes	No	Yes
Detect nearest Tutor	No	No	No	Yes
User experience	Moderate	High	High	Moderate

Type of teaching format	chat, video call	chat and video call	not support	chat, video call, home tutoring
-------------------------	------------------	---------------------	-------------	---------------------------------

2.6 Conclusion

To sum up this literature review, EziTutor mobile application is an application that provide tutoring platform for tutor and student and meet each other. Most of the tutor apps nowadays are based on mobile application. This application is used to view Tutor availability and find the nearest tutor from location of user since the main problem faced nowadays and by comparing three existing applications is there no platform for student select their tutor based on payment that they can afford and deal themselves about the tutoring price. So, with this application, user can view tutor availability and the nearest tutor from their location and make it is easy for them to book the tutor. As for the conclusion, EziTutor is chosen to be a mobile based application since people likely to use smartphone rather that another device.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter is discussing about the methodology used in developing EziTutor application. In software engineering, a software development methodology or system development methodology is a framework used to structure, plan and control the process of developing an information system(Professionals, 2018). This chapter also will explain about the hardware and software that will be uses for development.

3.2 Methodology

This term is used to promote phases, techniques, mechanisms and rules. Documentation, processes, management and coaching to improve the software development standard. It is also a procedure or analysis of every field's principles. There are certain steps that need to follow when using style of models of methodologies which will be used to the event of any system like System Development Life Cycle (SDLC) and Rapid Application Development (RAD) (Professionals, 2018).

To select the foremost appropriate model to be used for the event of this project, numerous software package development life cycle models are researched. In results, Rapid Application Development (RAD) methodology has been hand-picked to be employed in this project since it will facilitate to lock, shorten the SDLC and turn out an improved result. RAD methodology is chosen as a result of it used least coming up with in favour of fast prototyping, this permits the system to be developed quicker and makes it straightforward to alter the need.

3.2.1 Rapid Application Development (RAD)

This section discusses about the strategy that may be utilized in the event of this project. RAD model refers to a development life cycle that describes a method of necessities coming up with, design, implementation, verification and maintenance of the system. The computer code model that may be utilized in the event of EziTutor application is Rapid Application Development (RAD) model.

The ultimate purpose of RAD is to shorten the SDLC and during this approach respond earlier to dynamic data necessities of organizations as a result of the progression of the system will be measured, and time of development will be reduced. RAD will facilitate the systems run in no time with a decent quality. RAD can also manufacture high-quality system and primarily victimization unvaried prototyping at any stage of development and computerised development tools. RAD phases are shown in Figure 3.1 below:

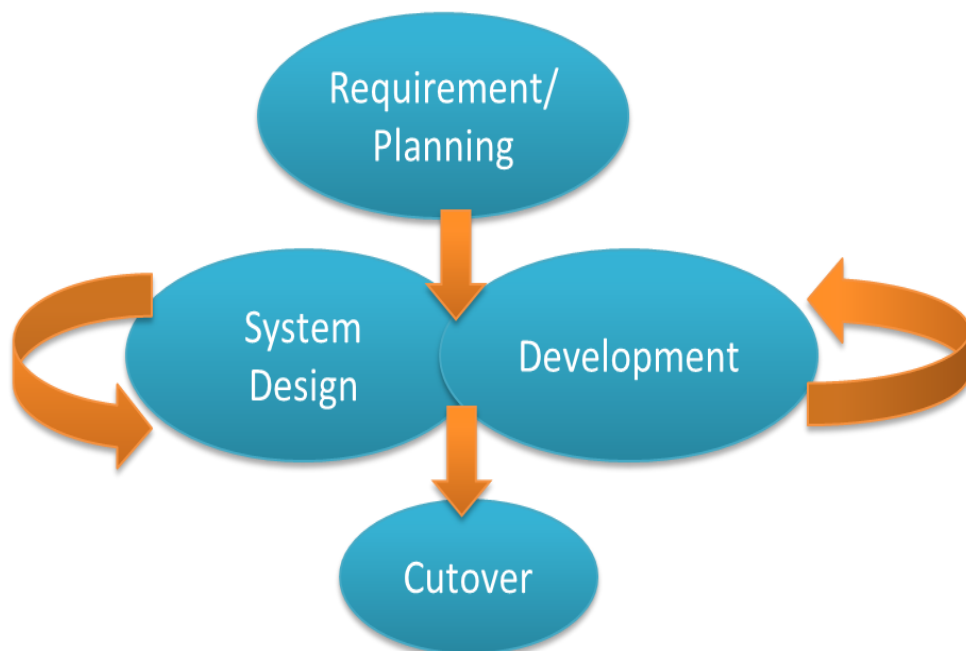


Figure 3.1 RAD phases

3.2.1.1 Requirement planning phase

Also called the conception Definition Stage, this stage defines the business functions and information subject areas that the system can support and determines the

system's scope. The developers ought to discuss and agree on business desires, project scope, constraints, and system needs. It ends once the team agrees on the key problems and obtains management authorization to continue. this can be done to document the initial needs for the system All the specified practicality for the system is collected from the consumer and therefore the end-users.

3.2.1.2 User design phase

In this part, information regarding additional regarding the user design for this project like users interact with systems analysts, develop models and prototypes that represent all system processes, inputs, and outputs. User design phase can rework the client needs into details and clarify the functionality of the diagram as well as the layout of the project platform. User design is interactive method that enables users to modify, understand, and approve an operating model of the system that meets user's demand. This part conjointly describes the input, output and interface of the event of EziTutor application.

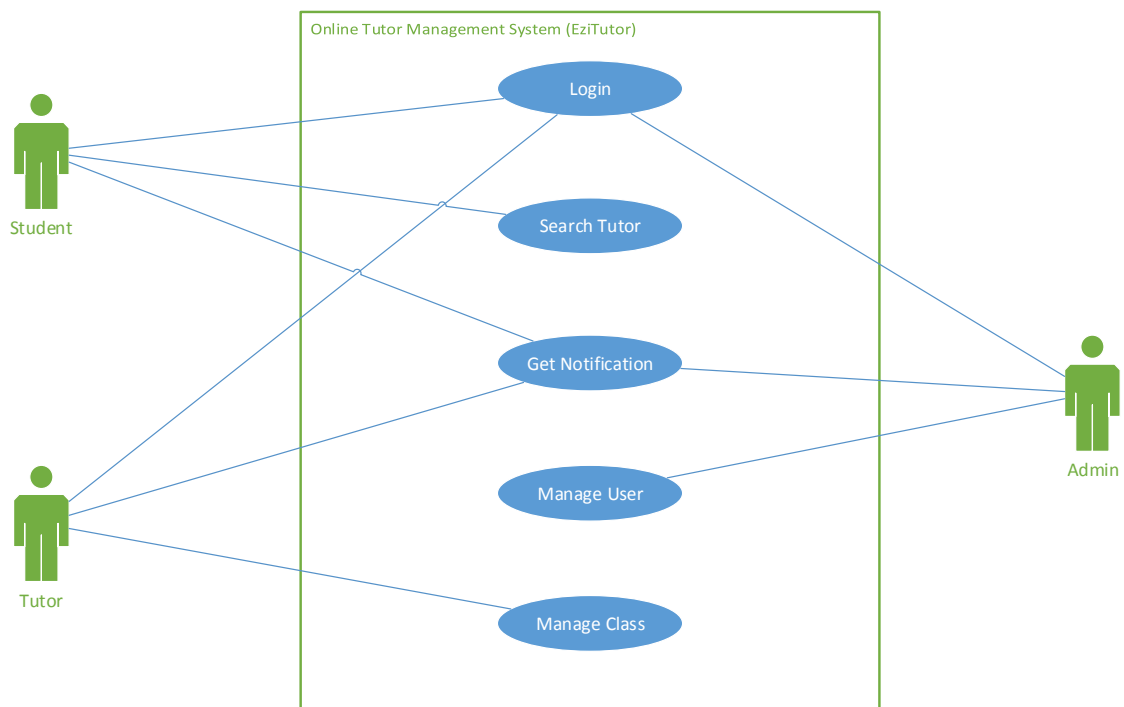


Figure 3.2 use case diagram for EziTutor

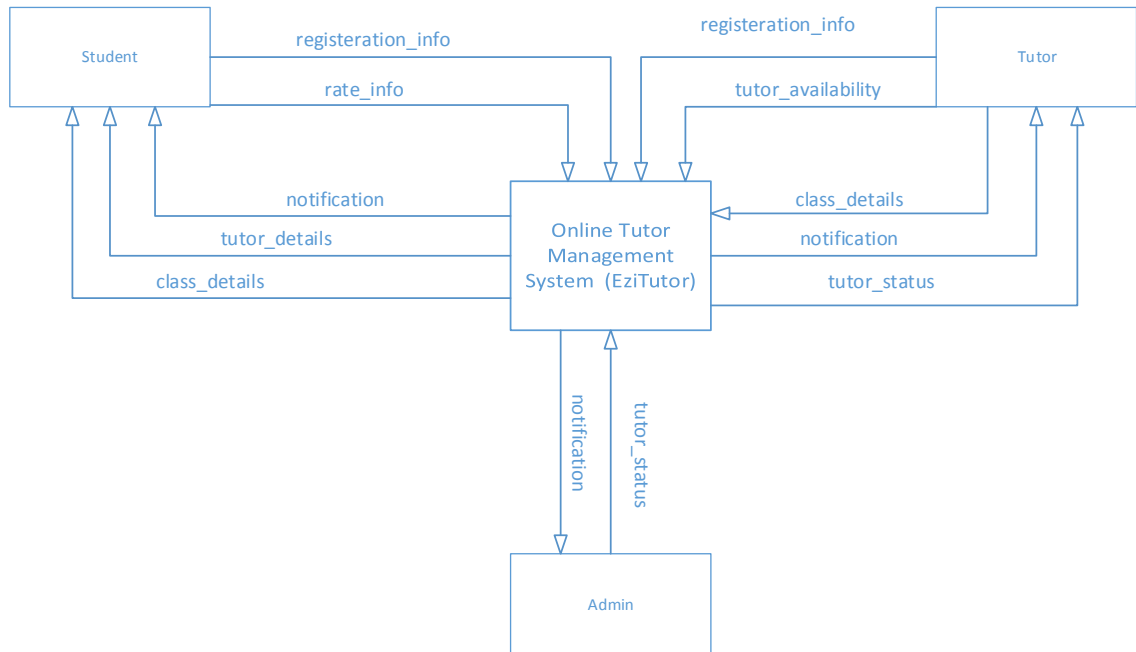


Figure 3.3 context diagram for EziTutor

From the use case diagram and context diagram, the actor that involved in the application are Student, Tutor and Admin. There are 5 things that will be discuss in detail in Appendix A which is login, search tutor, get notification, manage user and manage class.

3.2.1.3 Construction phase

This stage also defines the business functions and the data subject areas that the system will support and determine the scope of the system. The patron and thus the end users collect all the specified utility for the system. The developers should be forced to discuss and agree on business needs, scope of the project, constraints and system requirements. It ends when the team agrees on the key issues and gets management permission to carry on. This can often be done to document the system's initial needs.

3.2.1.4 Cutover phase

Cutover half is that the last a part of RAD methodologies. among the Cutover half, the entire technique is compressed. throughout this stage, the system needs to be complete with cryptography and technique testing of EziTutor application. As a result, the new system is created, delivered, and placed operative a great deal of sooner. The core performance of the device and to boot the installation, this type of testing is checked properly to ensure that the device remains fully sensitive once it is installed in the machine. By doing this project, sort of testing that has been used is sensible testing where it helps to verify that the system or application will perform and functions properly in line with the interface that already vogue.

3.2.2 Propose User Interface

This section shows the proposed designs for EziTutor Apps. This design is only to illustrate how EziTutor Apps would be. The designs presented are not the final system designs and will be modified according to client requirements and satisfaction.

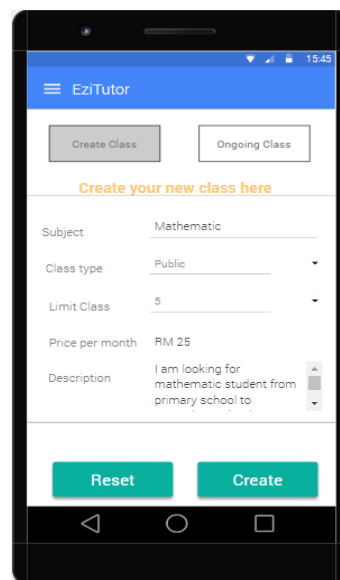


Figure 3.4 create class page

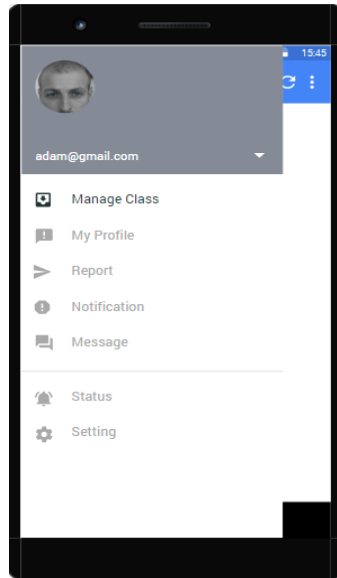


Figure 3.5 panel

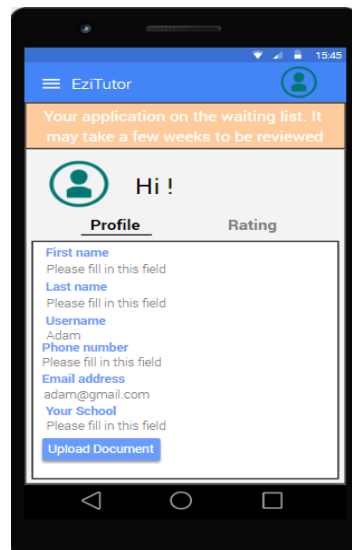


Figure 3.6 manage profile

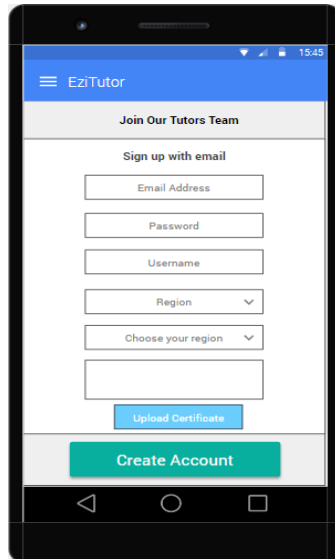


Figure 3.7 Tutor registration

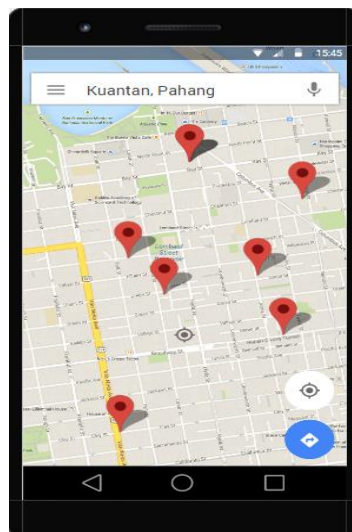


Figure 3.8 search tutor

3.2.3 General architecture and package module

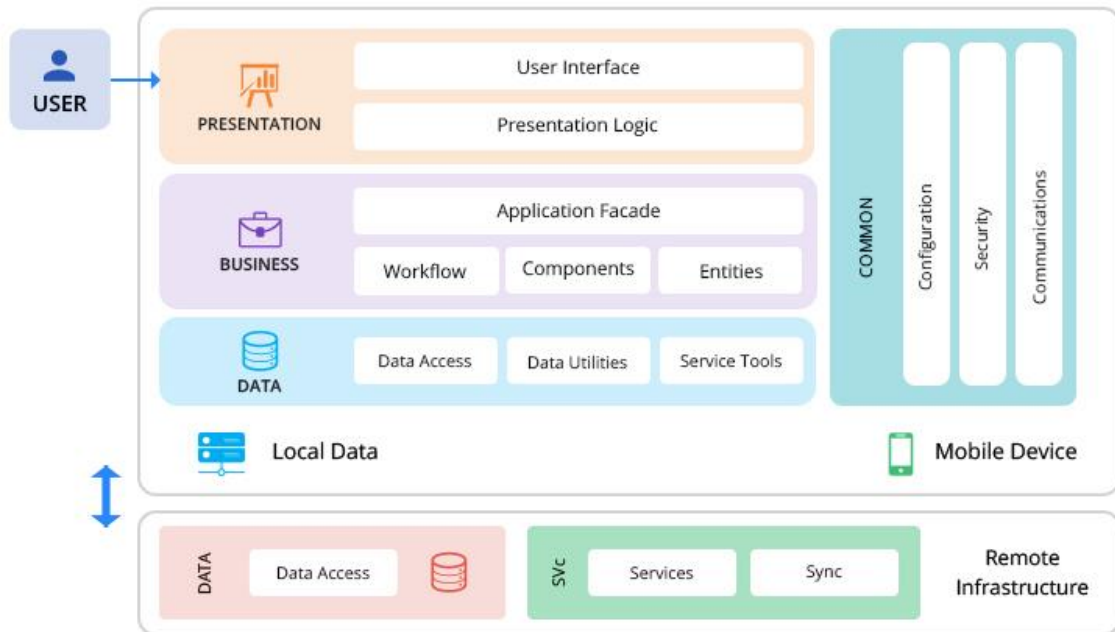


Figure 3.9 Basic Architecture for EziTutor

There are 3 layers within the Online Tutor Management System design. Start from the beginning, that is users, its needed services. Users can run the applying of the system. After that, the application information is on the server and all information can be saved on native data and information. That's why every layer can start with user. The system users are a student, a tutor and a Administrator. In Business Layer, server aspect data is called primarily.

All the activities like application, services and message that created are delivered and employed in real time system over the net. For any request from user are sent from programme to business interface layer. Business layer additionally concerned association between the info accesses. Business Layer additionally sanction native customers to access application remotely via the net. the information in application services can pull through the net, and everyone information can save in server. In server, this layer can take all the knowledge and can be saving at info layer. (Ubaid Pisuwala, 2018)

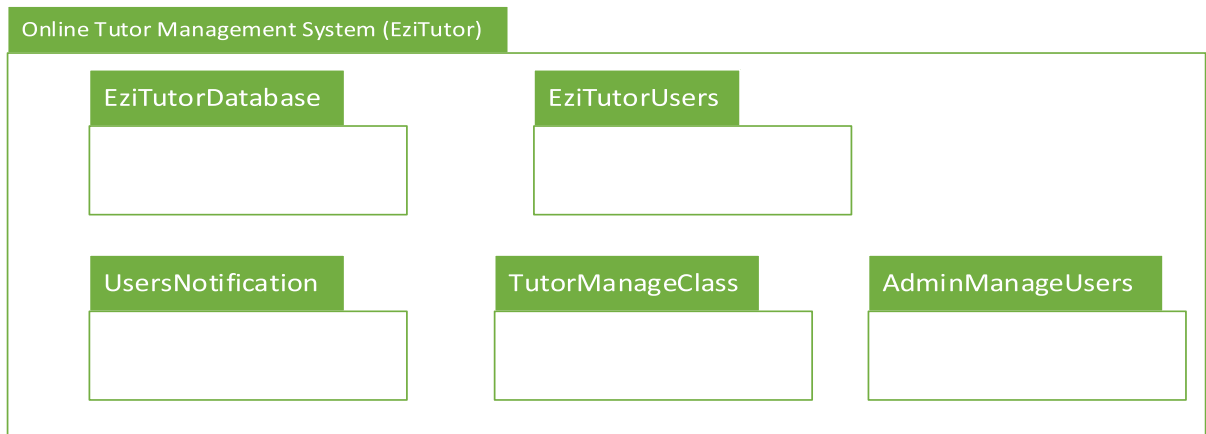


Figure 3.10 package module for EziTutor

i. EziTutorDatabase

The package does not contain any classes. This package just acts as reuse database that consist many tables.

ii.EziTutorUsers

The package is responsible to display all Student, Tutor and Admin information.

a) viewUser Class

iiiUserNotification

The package is responsible to handle notification send by the user and system.

a) handleNotification Class

iv.TutorManageClass

The package is responsible to handle management of class create by Tutor.

a) tutorManage Class

v.AdminManageUser

The package is responsible to handle management of User process by Admin.

a) adminManage Class

3.2.4 Software Requirement Specification (SRS)

Refer Appendix A

3.2.5 Software Design Document (SDD)

Refer Appendix B

3.3 Hardware and Software Tools

There are a few hardware and software that have been using in the development of EziTutor application.

3.3.1 Hardware Tools

Table 3.1 detail of hardware use and its specification

Items	Specification	Purpose
-Smartphone (Oppo A57)	-Android Marshmallow	-To test the application during development
-Inspiron N4050	-Windows 7 -Intel® Core™ i3, 3110M -CPU @ 2.40 GHz -RAM 8.00GB -64-bit OS	-For development and documentation of the Project
-Transcend StoreJet 25M3 Blue USB 3.0 -1TB Portable Hard Drive	-1TB -USB 3.0	-For save and backup data and files
-HP DeskJet 2130 series	-4800(horizontal) x 12000(vertical) dpi	- To print the documentation and full report

3.3.2 Software Tools

Table 3.2 detail of software use and its specification

Software	Purpose
-JustInMind	-To design interface
-Microsoft word 2016 -Microsoft Visio 2016 -Microsoft Project 2016 -Microsoft power point 2016	-To write documentation -To create use case diagram, context diagram -Slide presentation -To create Gantt chart
-Visual Studio code -Ionic	-To develop the mobile application
-Firebase	-Database server for the system

3.4 Gantt Chart

The Gantt chart is provided for the EziTutor application to know the timeline for developing this application.

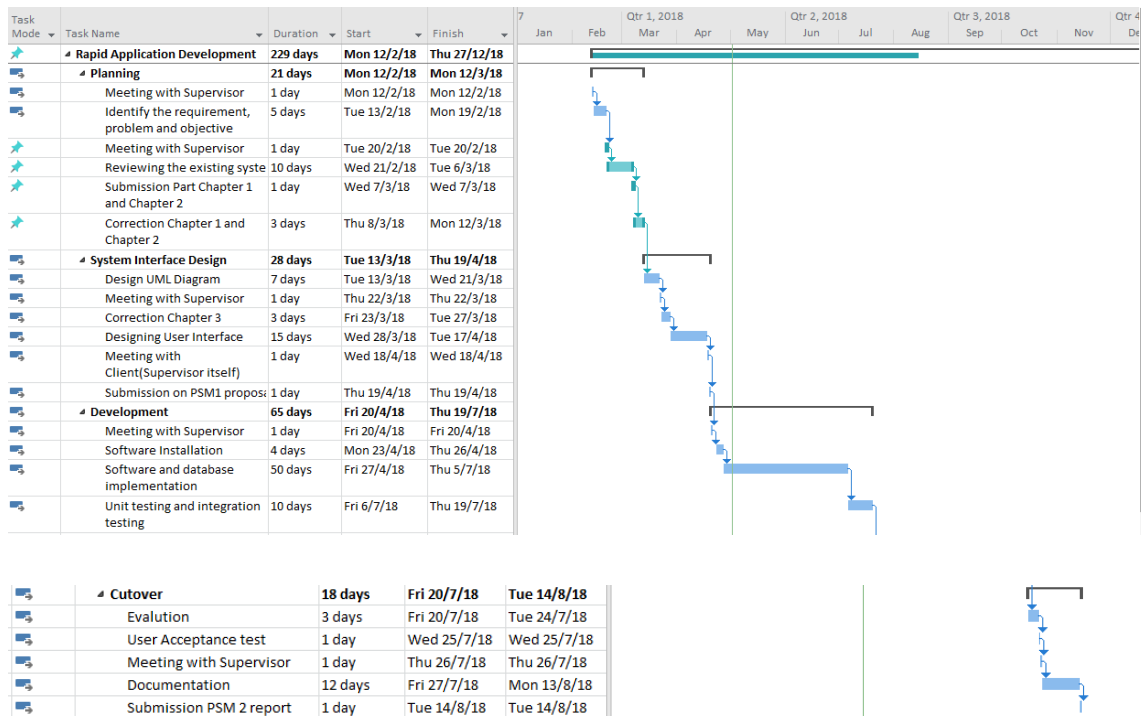


Figure 3.11 Gantt chart for EziTutor development by using RAD method

3.5 Testing Plan

This half describe regarding a way to check the appliance of the project, by designing on how to do the check. the main points of the check are recorded and documented as follows wherever the check embody event, condition, expected result, actual result, output (pass/fail), and comment. The designed check cases are accustomed proceed with the testing part. The check cases can collect and record all errors occur throughout testing part for application correction. Table 3.3 shows the example of the test plan.

Table 3.3 the example of the test plan

	Event	Condition	Expected Result	Actual Result	Output (Pass/Fail)	Comment
1	Click Search Tutor	Connect with gps user and System	Pass	Pass	Pass	Success

2	Click Search Tutor	Connect with gps user and System	Pass	Failed	Failed	Failed
---	--------------------	----------------------------------	------	--------	--------	--------

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter explain about the result that have been collected from EziTutor mobile application and discussion. These results produced by implementing and testing the application in certain condition and phase. Implementing and testing is very important in every project development to meet the user requirement, control the quality of the application and have a reliability, high performance application deliver to the client and user. In addition, it explains how the EziTutor mobile application will be tested based on certain condition to pass to production including the result from the testing process.

Section 4.2 shows the implementation steps for the application project. In section 4.3, the section will have discussed about the testing process of the mobile application to the user. The user will test the mobile application and comment from the user will be recorded in Table 4.1 and 4.2. Next, section 4.4 gives a brief description about the result that are obtained from the application. It included the design and layout of the application that's being used by the user. While, the advantages and disadvantages of the mobile application are stated in section 4.5. Lastly, section 4.6 will conclude all the content in the chapter.

4.2 Implementation Data/Model/Process/Hardware

4.2.1 Implementation

This subsection will show the code that has been implement in the EziTutor mobile application. The software that used to create this application are Node.js,

Github, Ionic Framework and Visual Studio Code. Language that used is HTML 5, Typescript, Angular and RxJS.

a) Class Welcome

```
1 import { Component } from '@angular/core';
2 import { IonicPage, NavController, NavParams } from 'ionic-angular';
3 import { AlertController } from 'ionic-angular';
4 import { StdLoginPage } from '../std-login/std-login';
5 import { TutorLoginPage } from '../tutor-login/tutor-login';
6 import { AdminLoginPage } from '../admin-login/admin-login';
7
8 @IonicPage()
9 @Component({
10   selector: 'page-welcome',
11   templateUrl: 'welcome.html',
12 })
13 export class WelcomePage {
14
15   constructor(public navCtrl: NavController, public navParams: NavParams,
16     | private alertCtrl: AlertController) {
17   }
18
19   presentAlert() {
20     let alert = this.alertCtrl.create({
21       | title: 'Welcome',
22       | subTitle: 'Together we learn',
23       | buttons: ['OK']
24     | });
25     alert.present();
26   }
27
28   tutorLogin(){
29     | this.navCtrl.push(TutorLoginPage);
30   }
31
32   adminLogin(){
33     | this.navCtrl.push(AdminLoginPage);
34   }
35
36   stdLogin(){
37     | this.navCtrl.push(StdLoginPage);
38   }
39
40   ionViewDidLoad() {
41     | this.presentAlert();
42   }
43
44 }
45
```

Figure 4.1 Class Welcome

Figure 4.1 is the class where user need to select whether they are Tutor, Student or Admin. This class control the type of the user account.

b) Class Login

```
constructor(private navParams: NavParams, private navCtrl: NavController, private auth: AuthService, fb: FormBuilder,
private googlePlus: GooglePlus, private http: HttpClient, public loadingCtrl: LoadingController) {
  this.loginForm = fb.group({
    email: ['', Validators.compose([Validators.required, Validators.email])],
    password: ['', Validators.compose([Validators.required, Validators.minLength(6)])]
  });
}

login() {
  let data = this.loginForm.value;

  if (!data.email) {
    return;
  }

  let credentials = {
    email: data.email,
    password: data.password
  };
  this.auth.signInWithEmail(credentials)
    .then(
      () => this.navCtrl.setRoot([StdSideMenuPage]),
      error => this.loginError = error.message
    );
}
```

Figure 4.2 Class Login

Figure 4.2 is the class to Login to class Home. This class control the authorize of valid user by referring their email and password.

c) Class Register

```
constructor(
  fb: FormBuilder, public navCtrl: NavController, public navParams: NavParams, private auth: AuthService,
  private afAuth: AngularFireAuth, private afDatabase: AngularFireDatabase
) {
  this.form = fb.group({
    firstname: ['', Validators.compose([Validators.required, Validators.required])],
    lastname: ['', Validators.compose([Validators.required, Validators.required])],
    email: ['', Validators.compose([Validators.required, Validators.email])],
    password: ['', Validators.compose([Validators.required, Validators.minLength(6)])],
    gender: ['', Validators.compose([Validators.required, Validators.required])],
    country: ['', Validators.compose([Validators.required, Validators.required])],
    state: ['', Validators.compose([Validators.required, Validators.required])],
    school: ['', Validators.compose([Validators.required, Validators.required])]
  });
}

signup() {
  let data = this.form.value;
  let credentials = {
    email: data.email,
    password: data.password
  };
  this.auth.signInWithEmailAndPassword(
    () => this.navCtrl.setRoot(StdLoginPage),
    error => this.signupError = error.message
  )
}
```

Figure 4.3 Class Register

Figure 4.3 is the class to Register for new user and to Login into the application. This class get the user details and save in database.

d) Class Location

```
loadmap() {
  var mapboxgl = require('mapbox-gl/dist/mapbox-gl.js');
  mapboxgl.accessToken = 'pk.eyJ1IjoibXVraGFpIiw1IjI6ImNqc6JlMjdocTBkejgza292amo4NjRnb2kiZQ.CBakBZywTt5FQfWdWz79Lg';
  //initialize the map
  var map = new mapboxgl.Map({
    container: 'map',
    style: 'mapbox://styles/mukhai/cjpd9kaai09s22rmmzoeoye13',
    center: [2.317600, 48.866500],
    zoom: 12.0
  });

  var positionOptions;

  let geoTracker = new mapboxgl.GeolocateControl({
    positionOptions: {
      enableHighAccuracy: true,
      timeout: 5000,
      maximumAge: 0
    },
    trackUserLocation: true,
  },
  this.updateLocation(positionOptions)
  );

  this.loadDummyTutor(map, mapboxgl);
  map.addControl(geoTracker);
}
```

Figure 4.4 Class Location

Figure 4.4 is the class where Mapboxgl is applying to put marker of event and class marker. User can click the marker and get the detail of it. Moreover, they can reserve their place if they interested.

e) Class TutorMyClass

```
myEvent = {} as TutorMyEvent;

constructor(private afAuth: AngularFireAuth, private afDatabase:AngularFireDatabase,
  public navCtrl: NavController, public navParams: NavParams, public viewCtrl: ViewController,
  public alertCtrl:AlertController, private camera:Camera) {

saveMyEvent(){
  this.afAuth.authState.subscribe(auth => {
    const pushId = this.afDatabase.createPushId();

    if(this.eventImage!=null)
      this.eventImage = storage().ref(`eventsPicture/${auth.uid}/${pushId}`);

    const item = {
      title: this.myEvent.title,
      description: this.myEvent.description,
      id: pushId,
      venue: this.myEvent.venue,
      payment: this.myEvent.payment,
      startDate: this.myEvent.startDate,
      endDate: this.myEvent.endDate,
      dueDate: this.myEvent.dueDate
    };

    this.afDatabase.list(`tutorMyEvent/${auth.uid}`).set(item.id,item);
    //this.calendar.createEvent(this.myEvent.title);
    alert("Event save");
    // this.save();
    this.navCtrl.pop();
  });
}
```

Figure 4.5 Class for Create Class

Figure 4.4 is the class for user create class for a subject and post it on the application.

f) Class TutorMyEvent

```
myEvent = {} as TutorMyEvent;

constructor(private afAuth: AngularFireAuth, private afDatabase:AngularFireDatabase,
  public navCtrl: NavController, public navParams: NavParams, public viewCtrl: ViewController,
  public alertCtrl:AlertController, private camera:Camera) {

saveMyEvent(){
  this.afAuth.authState.subscribe(auth => {
    const pushId = this.afDatabase.createPushId();

    if(this.eventImage!=null)
      this.eventImage = storage().ref(`eventsPicture/${auth.uid}/${pushId}`);

    const item = {
      title: this.myEvent.title,
      description: this.myEvent.description,
      id: pushId,
      venue: this.myEvent.venue,
      payment: this.myEvent.payment,
      startDate: this.myEvent.startDate,
      endDate: this.myEvent.endDate,
      dueDate: this.myEvent.dueDate
    };

    this.afDatabase.list(`tutorMyEvent/${auth.uid}`).set(item.id,item);
    //this.calendar.createEvent(this.myEvent.title);
    alert("Event save");
    // this.save();
    this.navCtrl.pop();
  });
}
```

Figure 4.6 Class for Create Event

Figure 4.5 is the class for user create event of interesting topic and post it on the application.

4.2.2 Implementation Testing

The mobile application through many phases of testing to implement the application process on this project. First, design the test cases based on the requirement function and interface so it suited with the process of application and can run testing smoothly. This is also to ensure that no error occur on the main process flow of the user interface application. At the second phase, based on the test cases from phase one, the functionality testing, test design specification and other testing is conducted. For

example, button, pop-up alert, navigation and other is functioning without any bug. At the last phase, the process and function of the application will be tested to ensure that all the function such as the server connection and smartphone camera working fine. The implementation of the application development was done by using Visual Studio as IDE in this project. All the data from the above phase will be recorded on User Acceptance Test (UAT) to collect all the data.

4.3 Testing and Result Discussion

The respondent for this project is the targeted user which are students and tutor. This project was develop using Visual Studio Code by help of ionic framework and Android Emulator to test the application codes and function on smartphone. Firebase was used as database/server for this application.

The testing phase consist of few steps which first testing run by developer itself to ensure that all the code and interface are running smoothly. Second steps, the application will be tested on a client to ensure all the system are functioning according to application requirement. On this testing steps also will help the developer whether the function on the application suitable with the targeted user working environment. The technique uses to develop, test and implement this project has met the objective of this project which are:

- I. To identify the requirement that useful for the EziTutor.
- ii. To develop EziTutor using hybrid application.
- iii. To test the EziTutor achieved the objective and met the requirements.

The user test result will be recorded on User Acceptance Test (UAT) that will be provided at appendices C.

4.4 User Manual

The User Manual is provided for the EziTutor mobile application to know how to perform the functions offered in this application. For this documentation, kindly refer to appendices D.

CHAPTER 5

CONCLUSION

5.1 Introduction

The application is unsuccessfully meet the user satisfaction for this beta version. Although it has all the requirement/module from the use case design that mention in previous topic but from the user sight it does not enough satisfaction as the application does not have a good user interface design and some functionality that still need to fix the logic part. However, this version of application can be use in but have some limit which is can only support small number of user login in one time, can show the location but cannot find the nearest one and admin need to verify the resume by its own capabilities. To make it clear based on the objective that have state in previous chapter, this project manages to cover all the objective which is gathering the user's requirement, developing the application using the hybrid platform which is Ionic framework and test the application so that it met the user requirement.

5.2 Advantages

The advantages that occurred during this development of project is as below:

- I. Readymade Ionic Framework online
By using the Ionic Framework, the design become easier and enhance the development time. The framework is easy to use and available online.
- II. No lagging
During the development using the Dell Inspiron, there is fewer lagging compare using the android studio IDE. Because of hardware limitation Ionic Framework give advantages for the developer.

5.3 Constraints

The constraint that occurred during this development of project is as below:

III. Design and developing database using Firebase

Using the Firebase database that implement the NoSql structure of database have some limitation to this project. EziTutor application use the Create, Read, Update and Delete (CRUD) method and its need more understanding because NoSql implement the parent child method. The structure is of database is more complex as it uses the realtime database compare to MySql and SQLite. In this project also use the Mapbox gl javascript that need to store the data in GeoJSON format as it uses object and array format which not suitable in firebase.

IV. Integrating maps using Google Maps API

Starting July 2018, the google maps implement the new condition for their google maps API. There is limitation to use the google maps in this project, so the design has been changing from using the Google Maps to Mapbox gl. There a lot to be learn in short time as when develop using Google Maps cannot fit the project. The Google Maps become blank and cannot be use.

V. Integrating maps using Mapbox gl js

The maps are nice and suitable for the project but has also limitation such as the popup message for the marker cannot/hard to be customize. There are no alternative ways for developer to make changes in pop up message for marker in the Mapbox gl js. The navigation cannot be done well. There also limitation of function and method that can be use compare to Google Maps API.

5.4 Future Work

In the future, this project is needed to be upgrade as it has a good application and have a good potential to be market outside. The new version will be release after the maintenance and new module has been create that meet the users need and satisfaction. The design as mention need to be improve and modify to attract more user and get the good feedback. The Industrial Evolution (IR) 4.0 encourage developer to have better skill in programming and other such as Artificial Intelligent and Arduino. To make this

application more complex and meet the IR 4.0, it will an implement Artificial Intelligent in it in the future. Thus, hopefully can this new version of application can be helpful for those who need and as for the application itself can be more capable, smooth, applicable, user-friendly and easy to be use by all users in different background.

5.5 Conclusion

As for the conclusion for the document of the project, this project is a system-based project which mainly targeting the student and tutor interaction for meeting each other. Admin role as support system in this application as it need to verify the quality of tutor that use this EziTutor application so that the student gets the qualify tutor for teaching them. Hopefully this application can be use in all region.

REFERENCES

- Babeesh. (2018). 1000 Chats Later: My Experience Tutoring English Online with Cambly. Retrieved May 4, 2018, from <http://beyondbabeesh.com/my-experience-tutoring-english-online-with-cambly/>
- Cambly. (2018). English Tutors Online - Cambly. Retrieved May 4, 2018, from <https://www.cambly.com/english?lang=en>
- code3apps. (2013). Home - Code3 Apps. Retrieved May 4, 2018, from <http://www.code3apps.com/>
- Labour, T., & Survey, F. (2015). Department of Statistics Malaysia Press Release, 264(June), 5–9. <https://doi.org/10.1017/CBO9781107415324.004>
- Lumpur, K. (2016). Researching Private Supplementary Tutoring, 49–59. <https://doi.org/10.1007/978-3-319-30042-9>
- Multidots. (2014). Difference Between Native vs Hybrid Android App Development. Retrieved April 15, 2018, from <https://www.multidots.com/difference-native-vs-hybrid-android-app-development-2/>
- Professionals, A. of M. T. (2018). Software Development Methodologies. Retrieved April 17, 2018, from <http://www.itinfo.am/eng/software-development-methodologies/>
- Renehan, S. (2015). Rising Tuition in Higher Education : Should we be Concerned ? *Visions for the Liberal Arts*, 1(1), 1–16.
- Snapask. (2018). Snapask: Learn confidently and efficiently. Retrieved May 2, 2018, from <https://snapask.com/en-my/>
- Ubaid Pisuwala. (2018). All about App architecture for efficient mobile app development. Retrieved May 4, 2018, from <https://www.peerbits.com/blog/all-about-app-architecture-for-efficient-mobile-app-development.html>

APPENDIX A

**SOFTWARE REQUIREMENT SPECIFICATION
(SRS)**

For

**Online Tutor Management System
(EziTutor)**

Generated by:

Muhammad Khairil Akmal Bin Mohd Khairuddin CB15128

1.0 PRODUCT DESCRIPTION

This section describes the detail of user characteristic of the system and the constrain of the system.

1.1 User Characteristic

The user of this system consists of 3 users; Student, Tutor and admin, where the admin is EziTutor Staff.

Table 1.1 User Characteristic

User	Description	Education Level	Background Experience
Student	Student from Malaysia and have register in any school.	Primary School Secondary School	The experience of the student is not required to use this app.
Tutor	Tutor can be anyone that have certificate and approve by the admin	Diploma and above	The experience of the student is not required to use this app.
Admin	Admin is the one who control the system	Diploma and above	The user has learnt the application in overall aspect to become admin

1.2 Constrain

- i. **Internet Limitation**
Smartphone that have slow internet connection will experience poor performance when using the application.
- ii. **Policy**
Only student and tutor from Malaysia can use this application.
- iii. **Hardware Limitation**
The smartphone must have Global Positioning System embedded in their phone architecture.

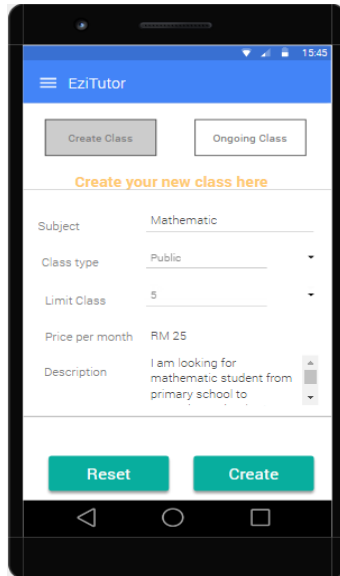
2.0 INTERFACE REQUIREMENT

This section describes the detail of user interface of the system.

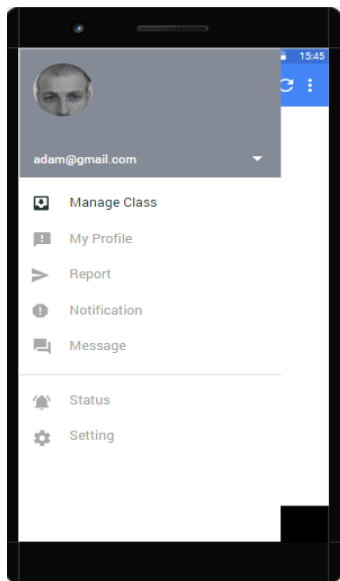
2.1 User Interface

This section explains about basic interface of the system

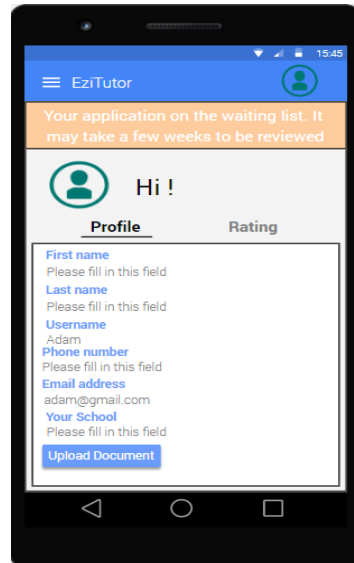
2.1.1 Create Class Page



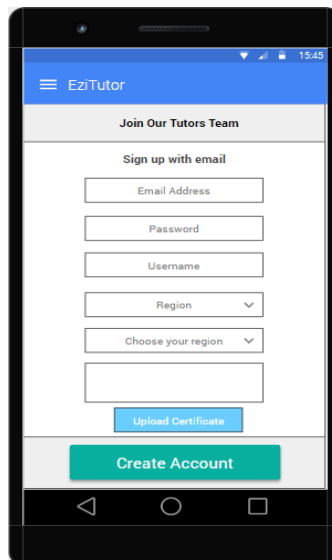
2.1.2 Side Bar Page



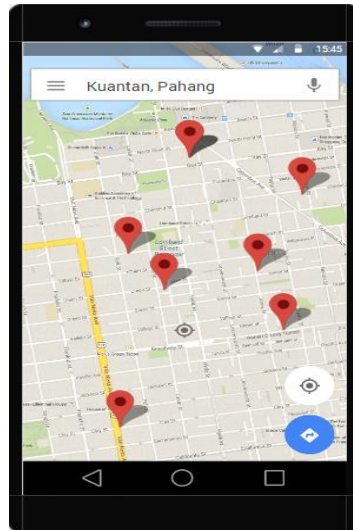
2.1.3 Manage Profile page



2.1.4 Registration page



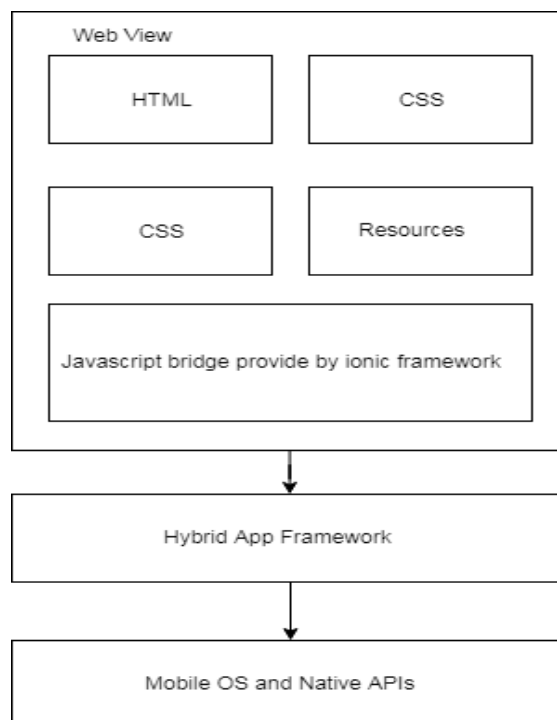
2.1.5 Search Tutor Page



2.2 Hardware Interface

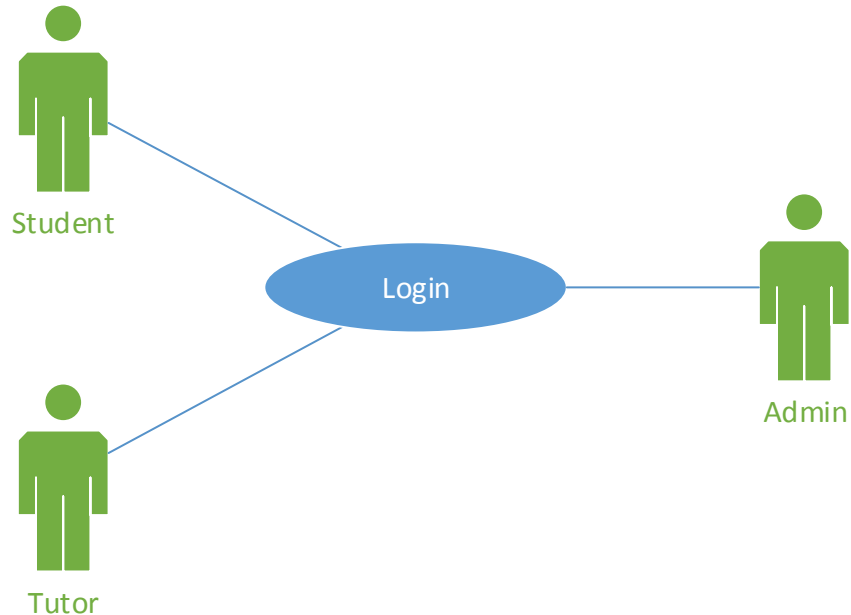
Not applicable

2.3 Software Interface



3.0 Software Product Features

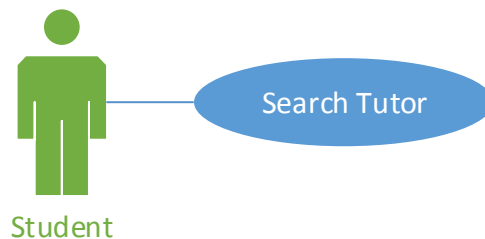
3.1 Use case Login



Use Case ID	TC_02_00
Brief Description	This use case is about user login to the system and what the terms and conditions that need to follow.
Actor	Student, Tutor, Admin
Pre-Conditions	1. All user need to be register first into the system and have a valid email. 2. All user has verified their email address.
Basic Flow	1 User enter their username, password and type of user. 2. System will validate with EziTutor database. [E1] [A1] 3. User enter the EziTutor main page based on their user type.
Alternative Flow	A1: 2alf in the basic flow the system cannot find the

	name, or the password is invalid, an error message is displayed. The actor can type in a new name or password or choose to cancel the operation, at which point the use case ends.
Exception Flow	E1: 2a The users not enter username or password correctly. -2a1 The EziTutor system will display error message and redirect to login page.
Post-Conditions	None
Rules	None
Constraints	Users can have one email for one username in the system.
Sequence Diagram	Refer Appendix A(A)

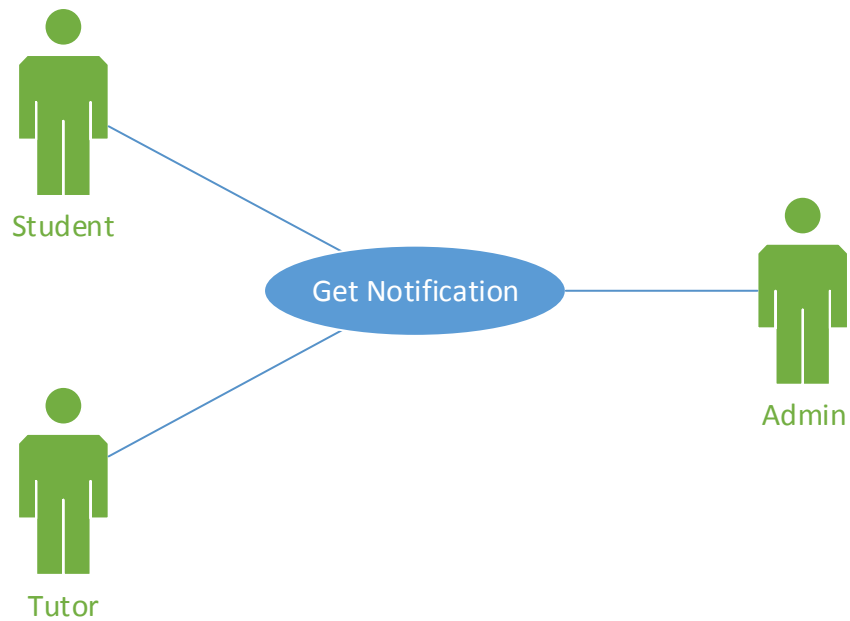
3.2 Use case Search Tutor



Use Case ID	TC_07_00
Brief Description	This use case is about Student find the Tutor using google maps and GPS function based on the nearest location to the user.
Actor	Student
Pre-Conditions	<ol style="list-style-type: none"> 1. Student must have the internet connection to access this feature 2. Student must have GPS function in their device. 3. The application must be in a ready state.

Basic Flow	<ol style="list-style-type: none"> 1. The Student activated the system. 2. The system open the google maps and connect with GPS function. [E1] [A1] 3. The system detect location of the student 4. The system check in the database location of the Tutor near with Student. 5. The system show the location of the Tutor with their details. 6. The student select the Tutor location 7. System will calculate distance between Tutor and Student.
Alternative Flow	<p>2a: The system shows in list form of Tutor registered in the system.</p> <p>-2a1 Student filter the result based on state.</p>
Exception Flow	<p>2a: The system cannot detect any GPS function in Student device.</p> <p>-2a1 The EziTutor system will display error message.</p>
Post-Conditions	<ol style="list-style-type: none"> 1. Student booking the Tutor class 2. The system give notification to Tutor. 3. The system update booking class into the Student activity section
Rules	None
Constraints	<p>Student can check details of the tutor one in a time.</p> <p>Tutor that approve by the EziTutor admin only will show in the system</p>
Sequence Diagram	Refer Appendix A(A)

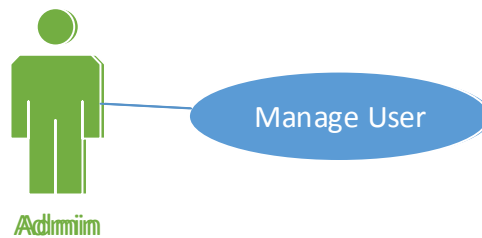
3.3 Use case Get Notification



Use Case ID	TC_08_00
Brief Description	This use case is about the display of the notification from a specific process such as registration, message, approve Tutor status, booking class and warning alert.
Actor	Student, Tutor, Admin
Pre-Conditions	<ol style="list-style-type: none"> 1. The Student, Tutor and Admin must install EziTutor application in their device. 2. Student, Tutor and Admin must have the internet connection to access this application. 3. The application must be in a ready state. 4. Student, Tutor and Admin has login into the system
Basic Flow	<ol style="list-style-type: none"> 1. Student or Tutor give report to admin about each other about ethic problem based on the rule of Ezi Tutor. 2. The system give notification to the admin about the report. 3. Admin give warning message to Tutor or Student. <p>[E1]</p> <ol style="list-style-type: none"> 4. The Tutor or Student get the warning message.
Alternative Flow	None
Exception Flow	3a: The system calculates the warning has exceeded the

	<p>limit.</p> <p>-2a1 The system displays warning limit message to the admin.</p> <p>-2a2 Admin confirm to remove the user.</p> <p>-2a3 The system remove the user and send email to user.</p>
Post-Conditions	1. The Student, Tutor and Admin successfully received the notifications based on their specific activities.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A(A)

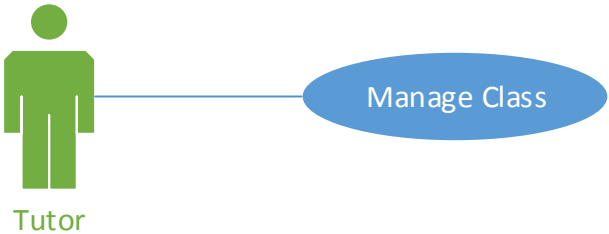
3.4 Use case Manage User



Use Case ID	TC_04_00
Brief Description	This use case is about admin approve Tutor applicant based on document uploaded and verify it with their education department.
Actor	Admin
Pre-Conditions	<ol style="list-style-type: none"> 1. The Admin must install EziTutor application in their device. 2. Admin must have the internet connection to access this application. 3. The Admin has clicked the manage user section. 4. Admin has login into the system
Basic Flow	1. Admin verify the document upload by the Tutor.

	<ol style="list-style-type: none"> 2. Admin approve the Tutor applicant. [E1] 3. System give notification to the Tutor. 4. System enable all EziTutor features for Tutor to use. 5. Tutor location added to the google maps.
Alternative Flow	None
Exception Flow	<p>E1: 4a The information of the document is not correct and reject by the Admin.</p> <p>-4a1 The EziTutor application will give notification to Tutor and display message “Please upload new document”</p>
Post-Conditions	<ol style="list-style-type: none"> 1. Tutor get notification and green light to use EziTutor as Tutor. 2. Tutor can use all features in EziTutor. 3. System give notification to Admin about the process of approval.
Rules	None
Constraints	1 Limit for unapproved Tutor applicant to inactive is 1 weeks without upload new document or certificate and will be remove automatically by the system.
Sequence Diagram	Refer Appendix A(A)

3.5 Use case Manage Class



Use Case ID	TC_05_00
Brief Description	This use case is about Tutor manage class and subject and open it to the Student.
Actor	Tutor
Pre-Conditions	<ol style="list-style-type: none"> 1. The Tutor must install EziTutor application in their device. 2. Tutor must have the internet connection to access this application. 3. The Tutor has clicked the manage user section. 4. Tutor has login into the system.
Basic Flow	<ol style="list-style-type: none"> 1. Tutor click the manage class button. 2. Tutor click the create class section. [A1] 3. Tutor fill in the form of create class features. 4. Tutor enter create class button. [E1] 5. System display confirmation message. 6. Tutor confirm the process 7. The class created and added to ongoing class section.
Alternative Flow	<p>A1: 2a The Tutor click the ongoing class section.</p> <p>-2a1 The Tutor closed or delete the class</p> <p>-2a2 System update in database.</p>
Exception Flow	<p>E1: 4a The Tutor do not insert the information completely.</p> <p>-4a1 The EziTutor system will display error message and redirect to the create class section.</p>
Post-Conditions	The Tutor get notified by the EziTutor system changes that have been made and open the new class to public.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A(A)

4.0 Requirement Traceability

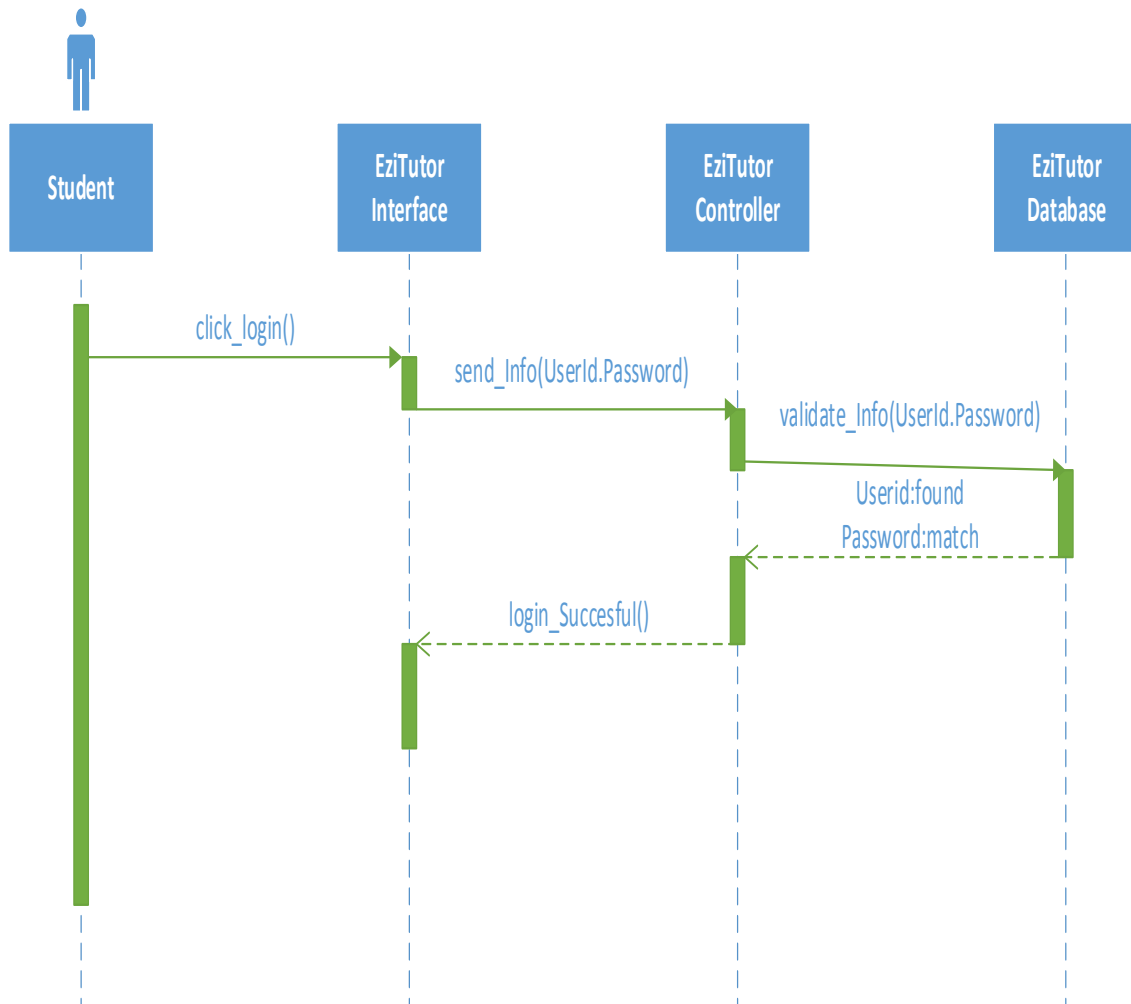
Requirement	Description
Manage User	Allow Admin to accept and reject the Tutor applicant
Manage Class	Allow Tutor to create, remove and modify the class.

5.0 System Requirement Approval

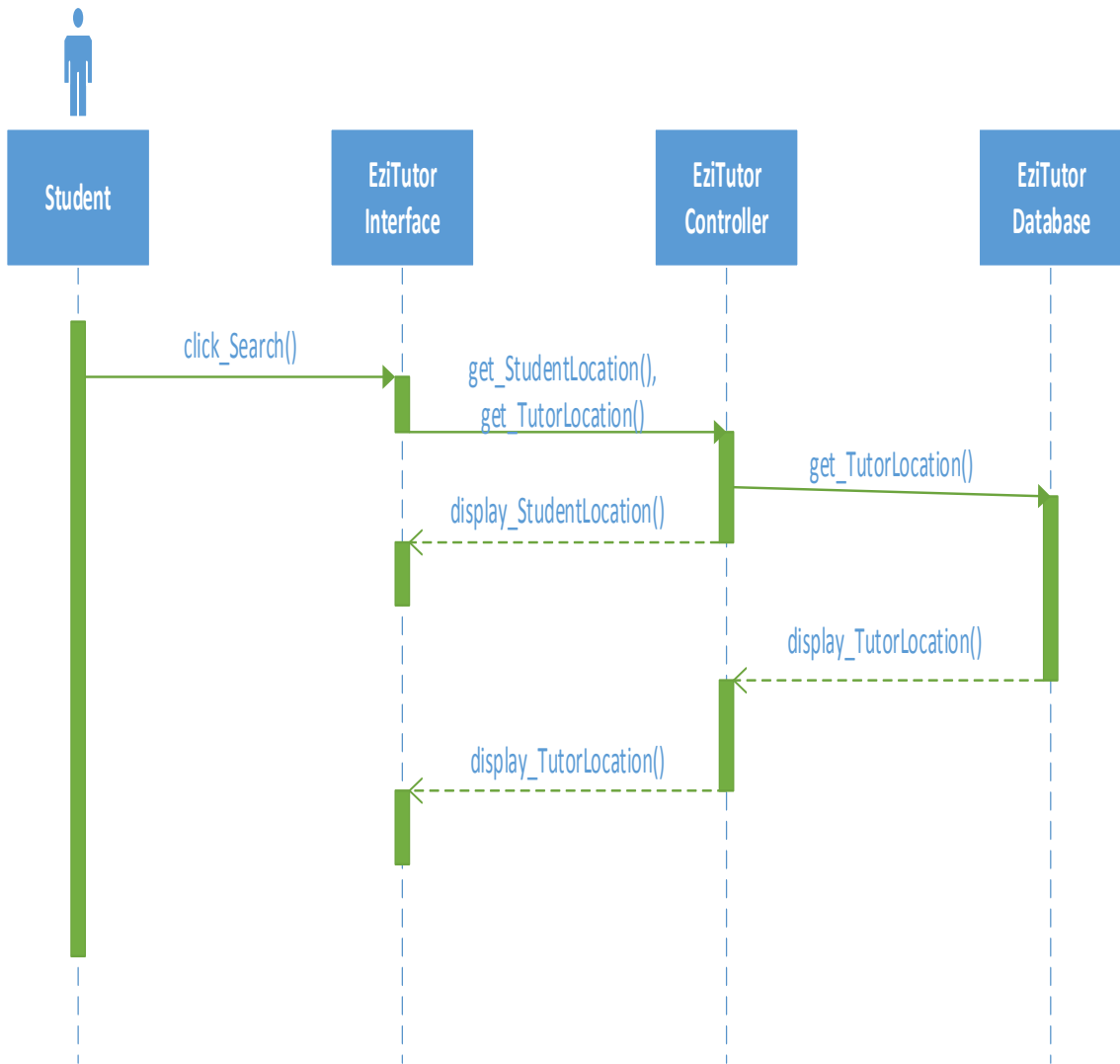
	Name	Date
<p>Authenticated by:</p> <p>_____</p> <p>Developer</p>		
<p>Approved by:</p> <p>_____</p> <p>Client</p>		

APPENDIX A(A)
SEQUENCE DIAGRAM

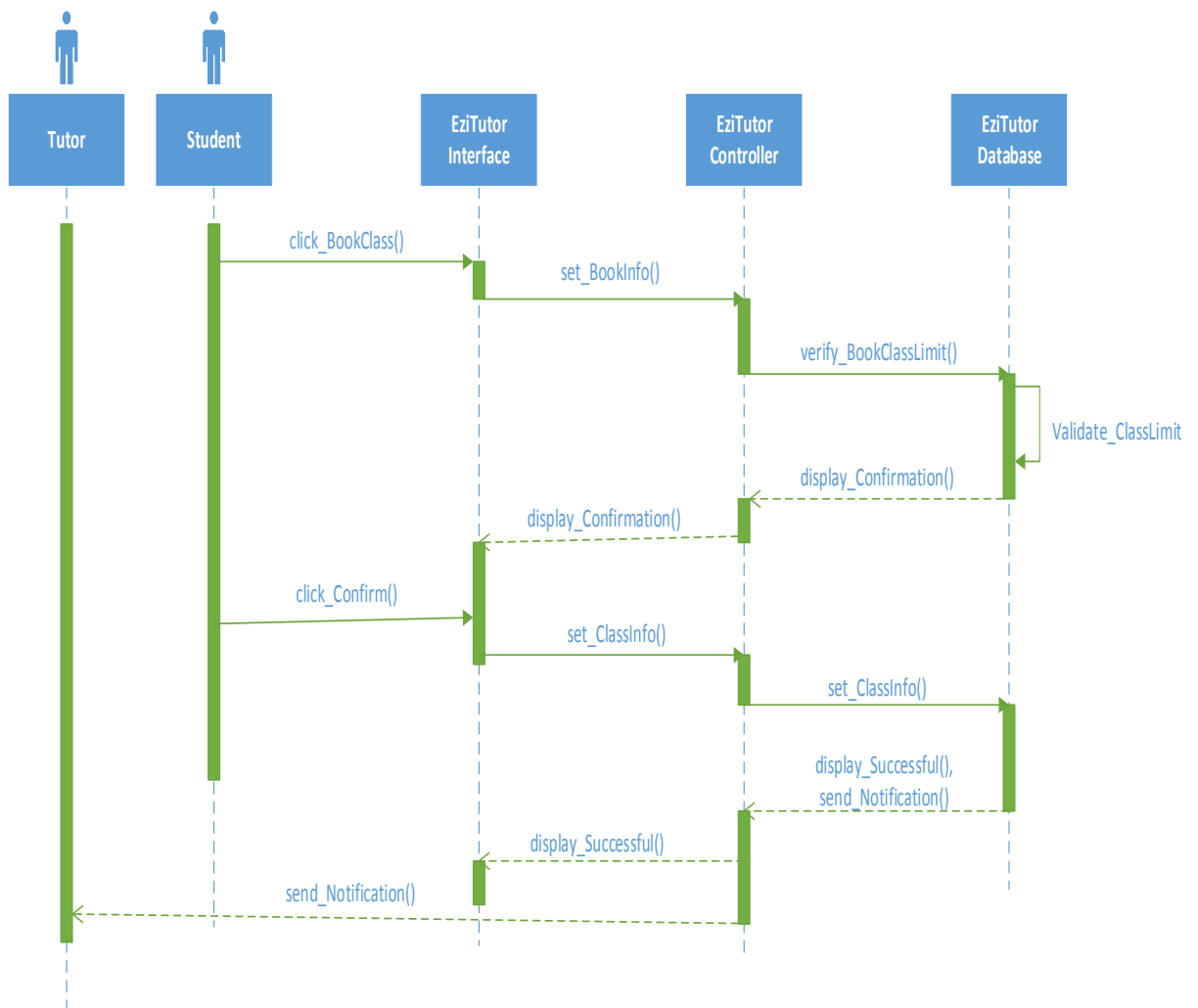
Appendix A-1



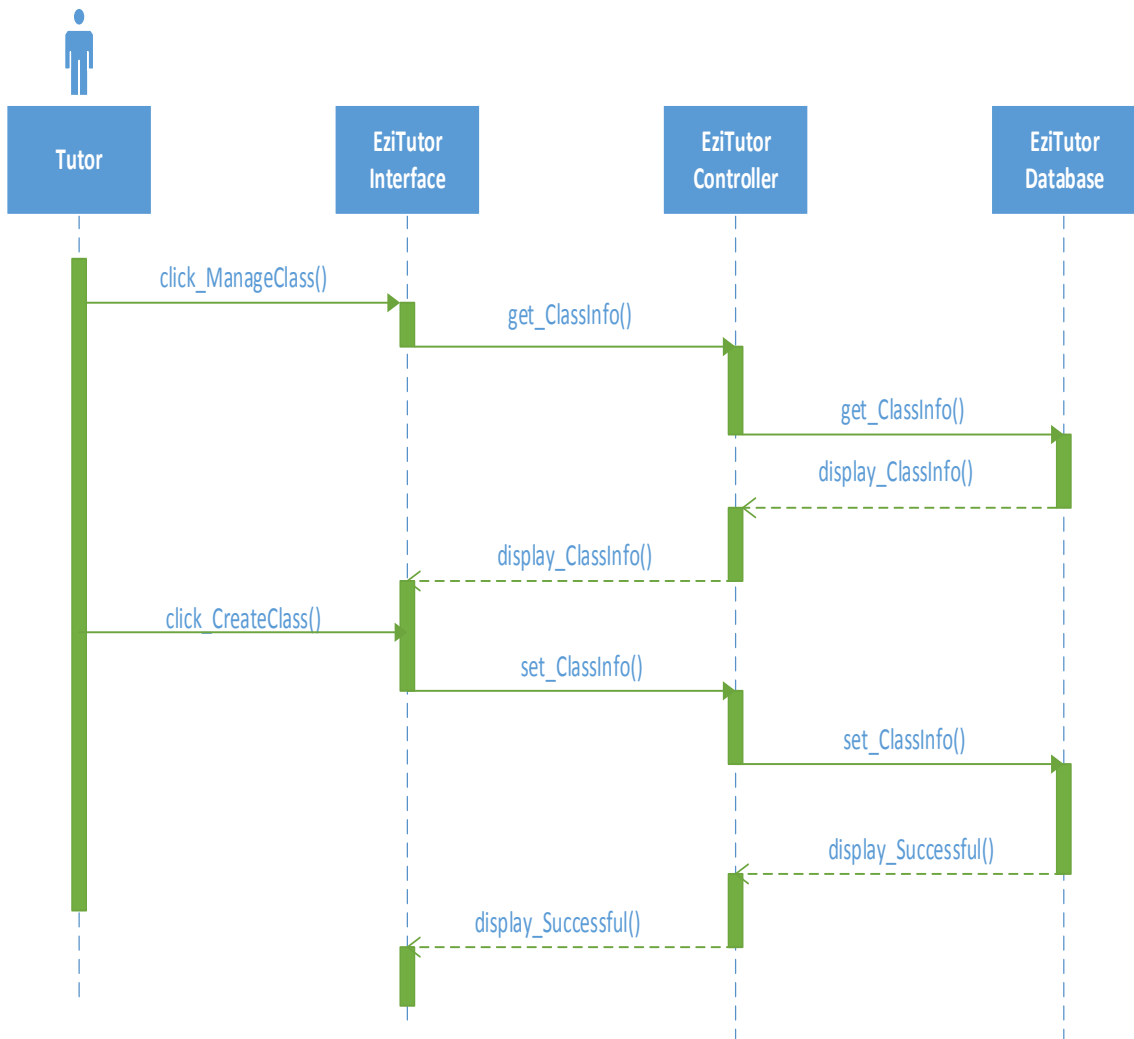
A-1.1: Sequence Diagram-Login



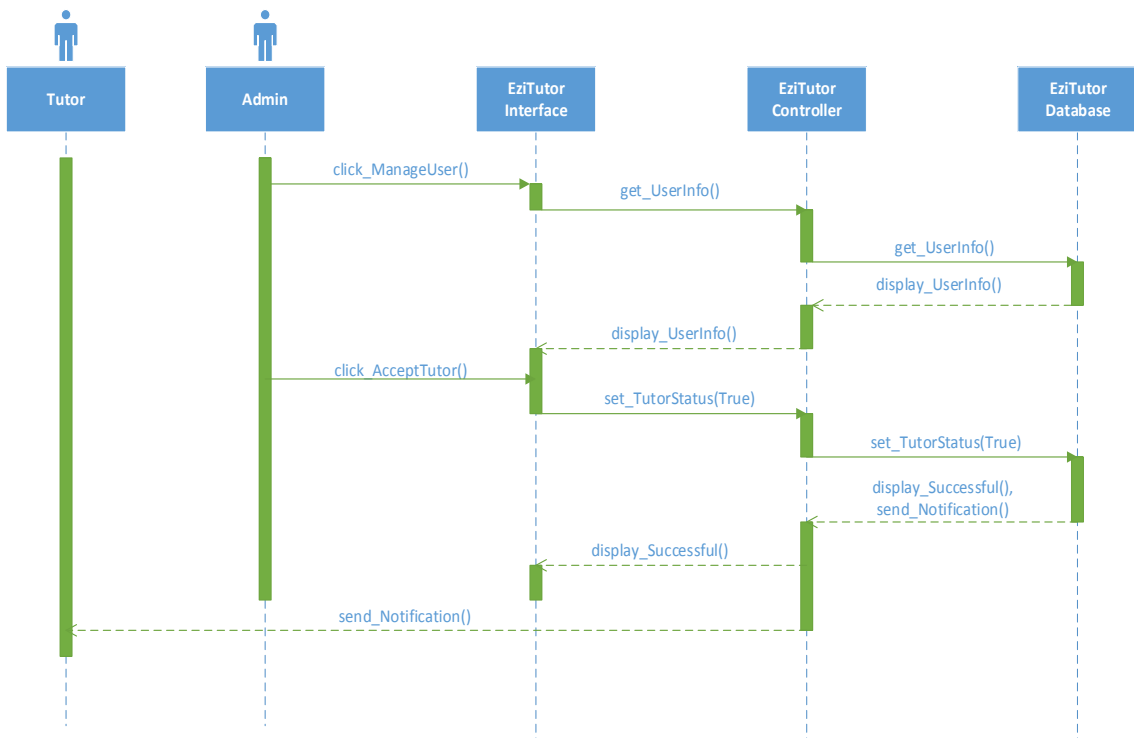
A-1.2: Sequence Diagram-Search Tutor



A-1.3: Sequence Diagram-Book Class



A-1.4: Sequence Diagram-Manage Class



A-1.1: Sequence Diagram-Manage User

APPENDIX B

**SOFTWARE DESIGN DOCUMENT
(SDD)**

For

**Online Tutor Management System
(EziTutor)**

Generated by:

Muhammad Khairil Akmal Bin Mohd Khairuddin CB15128

1.0 Data Dictionary

Table 1.1 Student Data Dictionary

Field name	Description	Data Type	Constraint
username	EziTutor username	varchar(20)	Primary Key
password	EziTutor password	varchar(20)	
student_Id	Student ID	varchar(20)	
student_Name	Student name	varchar(20)	
student_Phone	Student phone number	varchar(20)	
student_Email	Student Email	varchar(20)	Secondary Key

Table 1.2 Tutor Data Dictionary

Field name	Description	Data Type	Constraint
username	EziTutor username	varchar(20)	Primary Key
password	EziTutor password	varchar(20)	
Tutor_Id	Student ID	varchar(20)	
Tutor_Name	Student name	varchar(20)	
Tutor_Phone	Student phone number	varchar(20)	
Tutor_Email	Student Email	varchar(20)	Secondary Key

Table 1.3 Admin Data Dictionary

Field name	Description	Data Type	Constraint
username	EziTutor username	varchar(20)	Primary Key
password	EziTutor password	varchar(20)	
Admin_Id	Student ID	varchar(20)	
Admin_Name	Student name	varchar(20)	
Admin_Phone	Student phone number	varchar(20)	
Admin_Email	Student Email	varchar(20)	Secondary Key

2.0 Preliminary Design

2.1 System Architecture

2.1.1 Static Organization

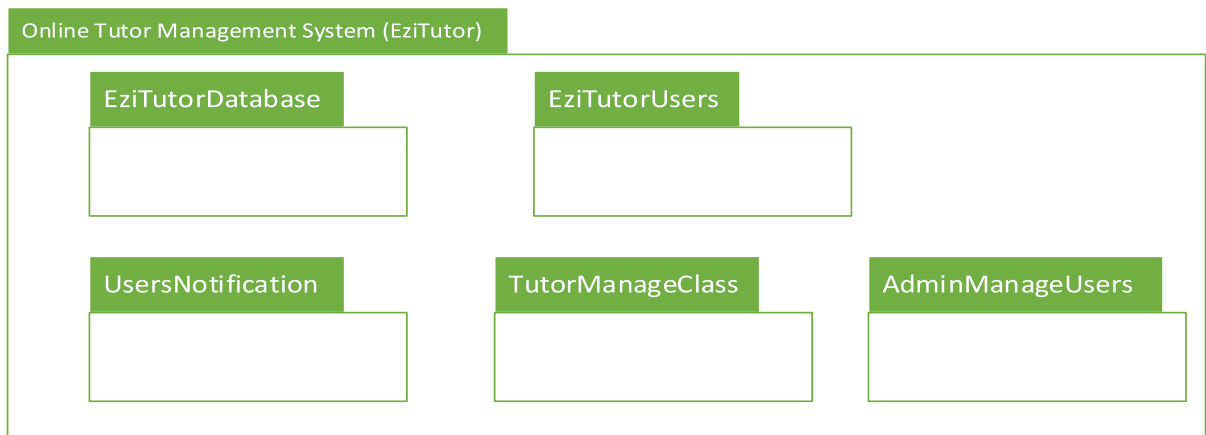


Figure 2.1 show package module for EziTutor

i. EziTutorDatabase

The package does not contain any classes. This package just acts as reuse database that consist many tables.

ii.EziTutorUsers

The package is responsible to display all Student, Tutor and Admin information.

a) viewUser Class

iiiUserNotification

The package is responsible to handle notification send by the user and system.

a) handleNotification Class

iv.TutorManageClass

The package is responsible to handle management of class create by Tutor.

a) tutorManage Class

v.AdminManageUser

The package is responsible to handle management of User process by Admin.

a) adminManage Class

APPENDIX C

**USER ACCEPTANCE TEST
(UAT)**

For

**Online Tutor Management System
(EziTutor)**

Generated by:

Muhammad Khairil Akmal Bin Mohd Khairuddin CB15128

1.0 TESTING REPORT

A In software development, user acceptance testing (UAT) - also called beta testing, application testing, and end user testing - is a phase of software development in which the software is tested in the like a real world by the intended audience. A User Acceptance Test (UAT) is designed to test the functionalities of EziTutor. The main functionalities will be tested are sign up, login, manage account, admin manage tutor, Tutor manage event or class, Student manage event or class, search tutor and notification. Users involved to test EziTutor will follow the user manual and signed the document as an evidence. The form below shows the UAT of EziTutor.

Test Case ID	Event	Test Data	Test Type	Expected Result	Actual Result	Pass/Fail	Comment
TC_01_00	Sign up new account						
TC_01_01	Sign up with new email	Email: testing@gmail.com Password:123456 Firstname: test Lastname: test Phone:012-3456789	Normal	Sign up successfully			
TC_01_02	Sign up with existed email	Email: testing@gmail.com Password:123456	Erroneous	Sign up failed			

		Firstname: tester Lastname: tester Phone:012-3456789					
TC_01_03	Sign up but no connection internet	Email: testing@gmail.com Password:123456 Firstname: tester Lastname: tester Phone:012-3456789	Erroneous	Sign up failed			
TC_02_00	Login to Account						
TC_02_01	Login with valid email and password	Email testing@gmail.com Password:123456	Normal	Sign up successfully			
TC_02_01	Login with invalid email and password	Email: testing@. com Password:123	Erroneous	Sign up failed			
TC_02_01	Login with email and invalid password	Email: testing@gmail.com Password:123	Erroneous	Sign up failed			
TC_03_00	Manage Account						
TC_03_01	Update Information with all valid information and all require field is fill and save into firebase	firstname: tester lastname: tester country: Malaysia state: Perak school: Univerisiti Malaysia Pahang age: 22	Normal	Update Information successfully			

		<p>address: Universiti Malaysia Pahang, Gambang Campus Lebuhraya Tun Razak phone: 012-3456789 Profile Picture: XYZ.jpeg</p> <p>Additional for Tutor resume: String; picture: String; experience: String;</p>					
TC_03_02	Delete Information with invalid authentication and update into firebase	Compare info based on email: testing@gmail.com	Erroneous	Delete Information failed			
TC_03_03	Get Information from firebase and Display Account Information	Load info based on email: testing@gmail.com	Normal	Display Information successfully			
TC_03_04	Add Information and save into firebase	<p>firstname: tester lastname: tester country: Malaysia state: Perak school: Univerisiti Malaysia Pahang age: 22 address: Universiti Malaysia Pahang, Gambang Campus Lebuhraya Tun Razak</p>	Normal	Add Information successfully			

		phone: 012-3456789 Profile Picture: XYZ.jpeg					
TC_04_00	Admin Manage Tutor						
TC_04_01	Download Tutor Resume	variable string/blob requested HTTP url to firebase	Normal	Download successfully			
TC_04_02	Display Waiting/Unapproved Tutor Account	request data from admin	Normal	Display successfully			
TC_04_03	Approve Waiting Tutor Account and change the Account into valid one	firstname: tester lastname: tester country: Malaysia state: Perak school: Univerisiti Malaysia Pahang age: 22 address: Universiti Malaysia Pahang, Gambang Campus Lebuhraya Tun Razak phone: 012-3456789 Profile Picture: XYZ.jpeg resume: String; picture: String; experiance: String	Normal	Approve Tutor Account successfully			
TC_05_00	Tutor Manage Event/Class						

TC_05_01	Add Event/Class	Picture Event: XYZ.jpeg Title: Mathematic Description: Syllabus for UPSR Start Date: 11/12/2018; End Date: 11/12/2018; Due Date: 10/12/2018; Venue: Kuantan; Limit: 10; Payment: RM50; Location: marker from Maps;	Normal	Add Event/Class successfully			
TC_05_02	Delete Event/Class	TutorId:123abc Email: testing@gmail.com EventId:123abc	Normal	Delete Event/Class successfully			
TC_05_03	Update Event/Class	Picture Event: XYZ.jpeg Title: Mathematic Description: Syllabus for UPSR Start Date: 11/12/2018; End Date: 11/12/2018; Due Date: 10/12/2018; Venue: Kuantan; Limit: 10; Payment: RM50; Location: marker from Maps;	Normal	Update Event/Class successfully			
TC_05_04	Add Event/Class and pass the data into Calendar with invalid passing value	Picture Event: XYZ.jpeg Title: Mathematic Description: Syllabus for UPSR Start Date: 11/12/2018;	Erroneous	Add Event/Class to calendar failed			

		End Date: 11/12/2018; Due Date: 10/12/2018; Venue: Kuantan; Limit: 10; Payment: RM50; Location: marker from Maps;					
TC_05_05	Add Event/Class and pass the data into Calendar with valid passing value	Picture Event: XYZ.jpeg Title: Mathematic Description: Syllabus for UPSR Start Date: 11/12/2018; End Date: 11/12/2018; Due Date: 10/12/2018; Venue: Kuantan; Limit: 10; Payment: RM50; Location: marker from Maps;	Normal	Add Event/Class to calendar successfully			
TC_06_00	Student Manage Class/Event						
TC_06_01	Booking New Event/Class	First name: Muhammad Last name: Abdullah Email: Muhammad@gmail.com Phone:012-456789	Normal	Add Event/Class successfully			
TC_06_02	Delete Event/Class	TutorId:123abc Email: testing@gmail.com EventId:123abc	Normal	Delete Event/Class successfully			
TC_07_00	Search Tutor						

TC_07_01	Search Tutor by invalid first name or last name or email or location.	first name: tester12 last name: tester12 email: testing@.com Location: Singapore	Erroneous	Search Tutor failed			
TC_07_02	Search Tutor by valid first name or last name or email or location.	first name: tester12 last name: tester12 email: testing@.com Location: Singapore	Normal	Search Tutor Successfully			
TC_08_00	Notification						
TC_08_01	Get Notification after doing invalid Student booking Event/Class	First name: xyz Last name: xyz Email: xyz@gmail.com Phone:012-456789	Erroneous	Notification Tutor failed			
TC_08_02	Admin Send Notification	Message: Your application has been approved	Normal	Notification Successfully			

2.0 SYSTEM TESTING APPROVAL

	Name	Date
Authenticated by: _____ Developer		
Approved by: _____ Client		

APPENDIX D

User Manual

For

**Online Tutor Management System
(EziTutor)**

Generated by:

Muhammad Khairil Akmal Bin Mohd Khairuddin CB15128

1.0 General Information

1.1 System Overview

Online Tutor Management System (EziTutor) is application for student and tutor connecting with each other. In this application, the quality of tutor is verifying by EziTutor admin which insuring the student get the best experience of tutoring when using this application. This application provides the location of tutor, event and class that has been create by the tutor that is near to the student. Furthermore, student also can create their own tutoring schedule by contacting the tutor who available in the tutor list section. Objective of this application is hopefully can help people to earn money as part time job and helping other to learning who cannot afford such things as tuition which need commitment to pay monthly by doing free tutoring session in this application.

2.0 System Summary

2.1 System Configuration

EziTutor is based on hybrid application that using Ionic 3 as its framework and supported by Angular 2 language. Ionic its self is using typescript language which is wrapper for JavaScript language. This application is built in android 6.0 which is Marshmallow. This application required this Android Operating System to run this application.

2.2 User Access Level

- a. Android dan iOS Smartphone user that install this application and have register to this application
- b. Authorize EziTutor Admin to manage and control the database which is firebase

2.3 Contingencies and alternate modes of the operation

a. In the case of no internet connection available, data cannot be retrieving from backend and saved into firebases.

b. The location of tutor, event and class cannot be render or load if the global position system is not on by the user itself.

3.0 Get Started

3.1 Installing and logging In

First, search in Google Play and install EziTutor in your android device. The first page that will show up is Welcome page. After that there are two button which lead to different login system as shown as Figure 3. Choose whether you want to login as Tutor or Login as Student. As for admin there are secret path that led to the admin accessibility.

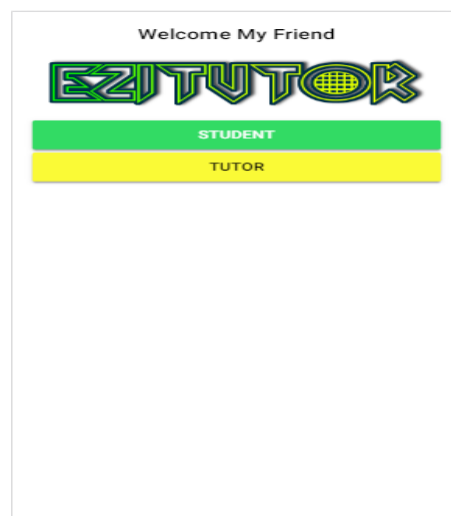


Figure 3.1 the Welcome Page for the EziTutor

If student category has been selected form the application, Figura 3.2 show the Student Login for the Application. If you have the google account, you can use it for login. You also can sign up into EziTutor mail via Sign Up button as show in the Figure 3.2. After the Signup has been successfully you can login by entering valid email and password.

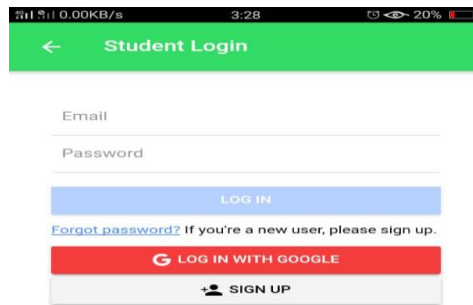


Figure 3.2 the Student Login Page for the EziTutor

If Tutor category has been selected from the application, Figure 3.3 shows the Tutor Login for the Application. If you have the Google account, you can use it for login. You also can sign up into EziTutor mail via Sign Up button as shown in Figure 3.3. After the Sign up has been successfully you can login by entering valid email and password.

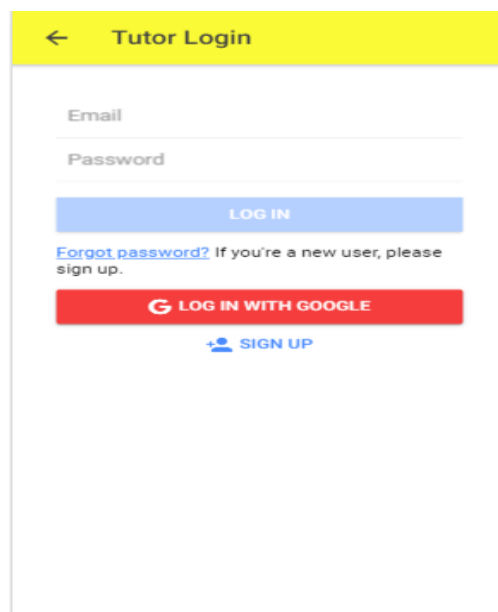


Figure 3.3 the Tutor Login Page for the EziTutor

0.00KB/s 5:01 10%

← Tutor Sign Up

First Name

Last Name

Email

Password

Gender ▼

Country ▼

State ▼

School Name

Phone Number

SIGN UP

Figure 3.4 the Tutor Signup Page for the EziTutor

1.13KB/s 5:01 10%

← Student Sign Up

First Name

Last Name

Email

Password

Gender ▼

Country ▼

State ▼

School Name

SIGN UP

Figure 3.5 the Student Signup Page for the EziTutor

The difference between signup Tutor and Student are the Student after this page will redirect to EziTutor Main Page and been accepted but for Tutor it is different. They need to upload their resume and other related document and after that they need to wait until Admin of EziTutor has been verify and validate Tutor resume. If there is no problem Admin will approve Tutor Account, so it can be use as usual.

3.2 Manage Account

This figure 3.6 shows the user interface for tutor account management and it has the same variable as Student account, but Student does not have My resume section as it only for Tutor. In here There are two button which is can update information and logout. To update just simply click Update Information and will go to Update page. User can update their current information and upload new user profile picture. If logout button is click, then the application will prompt confirmation message then will direct to welcome page back.

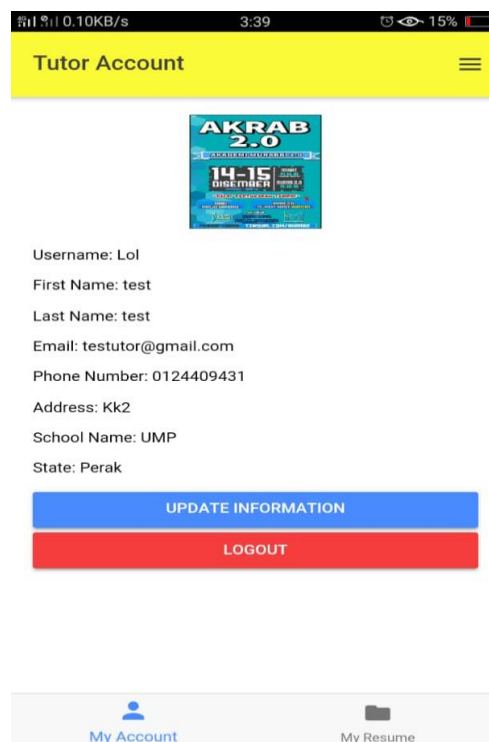


Figure 3.6 the Tutor Account Page for the EziTutor

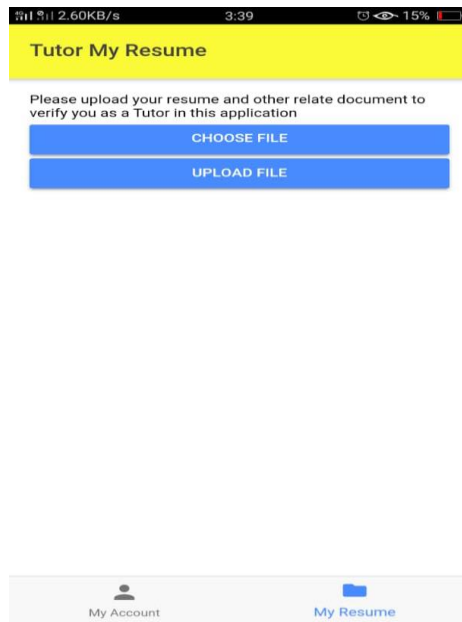


Figure 3.7 the Tutor My Resume Page for the EziTutor

Figure 3.7 show the interface that show after user click My Resume button. There are two button which is Tutor need to choose File which currently available is pdf extension file only and then the application will temporary hold it in the application. After Tutor click upload file, it will upload into the firebase storage. Admin will get notification about it.

3.3 Side menu and Tab

Figure 3.8 shows the side menu for the student section. Each content that has in the side menu will lead to other page that has Tab in it. Example is when Student tap home button, student will navigate to three tab which is location page, event/class list page and Tutor list page. The side menu can be access at top right corner of the application.

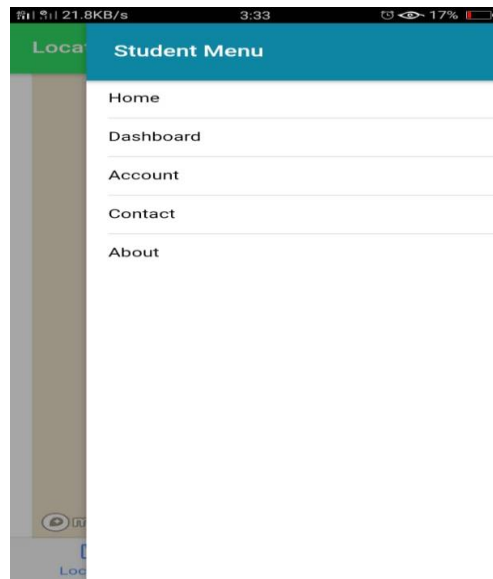


Figure 3.8 the Student Side Menu Page for the EziTutor

3.4 Booking Event/Class

There are two ways to know the Event/Class in this application. First based on the figure 3.9 which show the location of Tutor and the Event/Class that has been created by the Tutor. The marker of position shows the Event/Class which will held on that place. The Student can select the marker and book the place if they interested. The student can locate the nearest targeted place to join in.

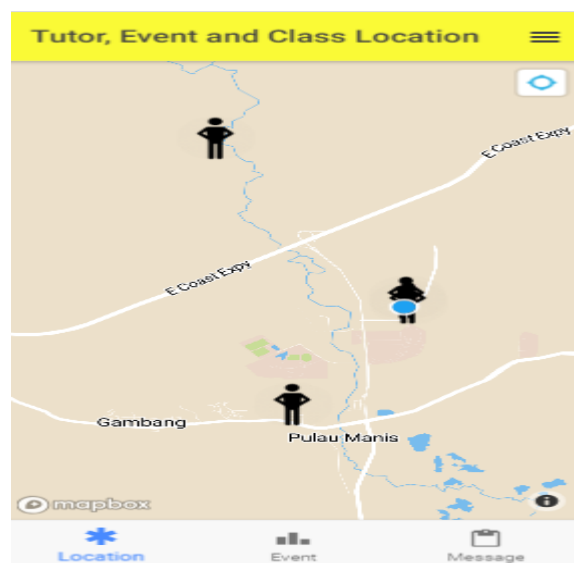


Figure 3.9 the Location of Tutor, Event and Class Page for the EziTutor

Second, based on the figure 3.10 which show the of the Event/Class in List form that has been created by the Tutor. The student can select the list and book from this section.

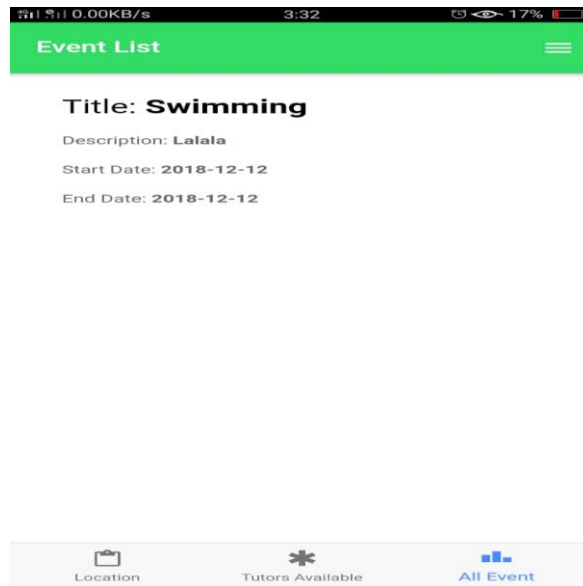


Figure 3.10 the Event and Class in List form Page for the EziTutor

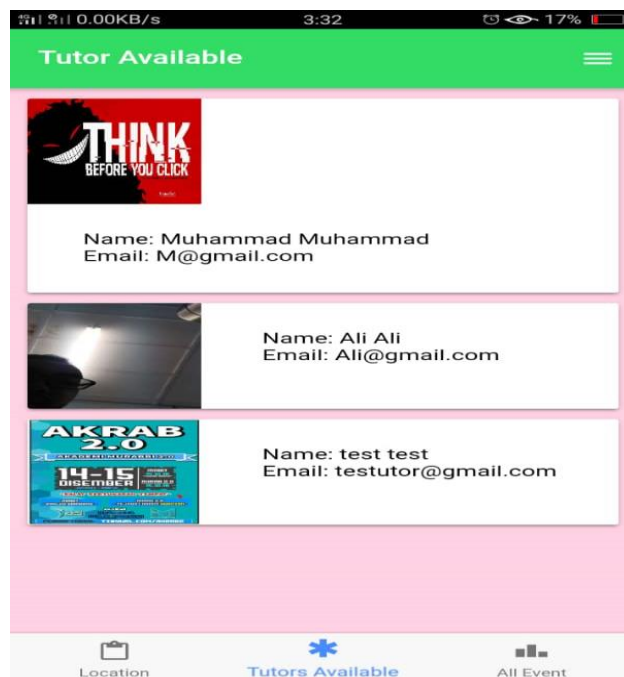


Figure 3.11 the Tutor Available Page for the EziTutor

The student can customize the scheduling of tutoring section and what subject that they want by contacting the available Tutor. When student tap the list, they will redirect to Tutor info and can send message from there.

3.5 Admin manage Tutor

Figure 3.12 and 3.13 show the activities that done by the admin. Figure 3.12 show the new list that request to become Tutor in EziTutor application. After admin tap the selected list, the page will direct to figure 3.13 page. All the information of the Tutor is displayed. Admin can view and download their resume for further action and if it is ok admin will approve the Tutor account.

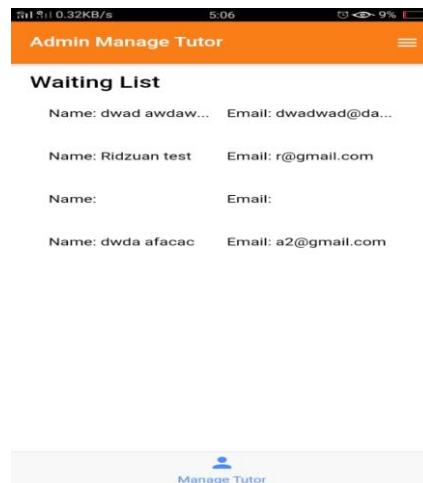


Figure 3.12 the Admin Manage Tutor Page for the EziTutor

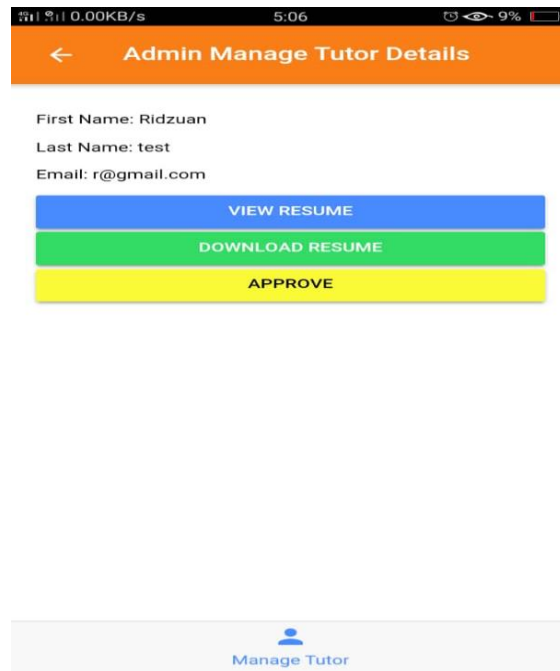


Figure 3.13 the Admin Manage Tutor Details Page for the EziTutor

3.6 Tutor manage Event

This section where Tutor can add, delete or update new event/class for tutoring. There is add button at bottom right corner which will led to create new event/class page. In this section also provide the calendar which show the date of Tutor Event and Class. At the bottom of the calendar show the event that has been created by the Tutor. Refer figure 3.14.

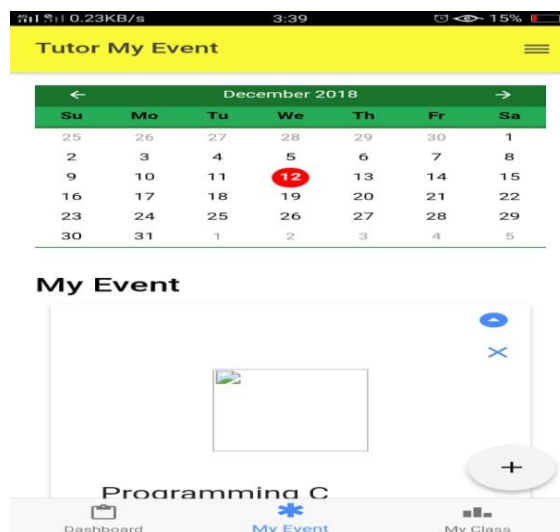


Figure 3.14 the Tutor Manage Event/Class Page for the EziTutor

The screenshot displays the 'Tutor Add New Event' page. At the top, there is a yellow header bar with a back arrow on the left, the text 'Tutor Add New Event' in the center, and a close 'x' icon on the right. Below the header, there are seven text input fields, each with a label above it: 'Title', 'Description', 'Venue', 'Price', 'Start Date', 'End Date', and 'Last Register'. A blue button with the text 'SAVE' is located below the input fields. At the bottom of the screen is a navigation bar with three items: 'Dashboard' with a folder icon, 'My Event' with a gear icon and highlighted in blue, and 'My Class' with a bar chart icon.

Figure 3.15 the Tutor Manage Event/Class details Page for the EziTutor

Figure 3.15 show the input that Tutor need to fill in to create new event. After it has created successfully the event will show on the event list of the student section also.