UTAR SOCIETIES' EVENTS INFORMATION SYSTEM

TEOH WAN CHING

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Hons.) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

April 2019

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature :

Name : Teoh Wan Ching

ID No. : 15UEB02834

Date :

APPROVAL FOR SUBMISSION

I certify that this project report entitled **“UTAR SOCIETIES’ EVENTS INFORMATION SYSTEM”** was prepared by **TEOH WAN CHING** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Hons.) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : ______________________________

Supervisor : Ooi Ean Huat

Date : ______________________________

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my research supervisor, Mr. Ooi Ean Huat for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents for giving me encouragement throughout the process of preparing and completing this report.

Moreover, I would also like to express my gratitude to my friends who had given me advices, supports and feedbacks throughout the process of developing this project.

ABSTRACT

Nowadays, every educational institution has a website for their own university or college to display all detailed information about their campus and educational resources. However, a formal website to display detailed information about co-curricular activities is not found in most of the universities and colleges. Statistic has shown that this phenomena is causing a lot of issues to students every day to get themselves involving in co-curricular activities. Hence, the objective of this project is to develop an online co-curriculum portal to manage all of the co-curricular staffs such as societies and events organized by various societies in order to solve all of the existing problems faced by students. A literature review on existing similar system had been conducted to determine the common features which are necessary for students and to find out some useful key features that are not provided by most of other existing systems. The software development methodology which is used in developing this system is Evolutionary Prototyping model. By using this approach, prototypes will be produced from time to time to send for user's evaluation in order to proceed to the next improvement. The final outcome of this project is a Progressive Web Application which is a standard platform for managing co-curricular stuff. Lastly, the evaluation phase of this project is to let user to test this system and then complete a survey form.

TABLE OF CONTENTS

DECLARATION **ii**
APPROVAL FOR SUBMISSION **iii**
ACKNOWLEDGEMENTS **v**
ABSTRACT **vi**
TABLE OF CONTENTS **vii**
LIST OF TABLES **xi**
LIST OF FIGURES **xii**
LIST OF SYMBOLS / ABBREVIATIONS **xi**
LIST OF APPENDICES **xv**

CHAPTER

1 INTRODUCTION **16**
1.1 Background 16
1.2 Problem Statement 17
1.3 Project Objectives 19
1.4 Project Goal 19
1.5 Final Solution 19
1.6 Final Approach 21
1.7 Project Scope 22
1.7.1 Target Users 22
1.7.2 Modules Covered 22
1.7.3 Modules Not Covered 23

2 Literature Review **25**
2.1 Existing Co-curriculum Information System Review 25
2.1.1 UTAR Portal 25

2.1.2 General common features 27

2.1.3 Particular features 29

2.1.4 Comparison among existing systems 39

2.1.5 Additional included features in system 39

2.2 Software Development Methodology Review 41

2.2.1 Waterfall Methodology 41

2.2.2 Agile Methodology 42

2.2.3 Evolutionary Prototyping Methodology 44

2.2.4 Rapid Application Development Methodology 45

2.2.5 Comparison among Software Development Methodologies 46

2.2.6 Evaluation of the Selected Software Development Methodology 47

2.3 Databases Review 48

2.4 Front-end Frameworks Review 48

2.5 Cloud Services Review 50

2.6 Native and Web Applications Review 51

2.7 Recommender System Review 52

3 Methodology and Project Management 54

3.1 Chosen Software Development Methodology 54

3.1.1 Requirements Gathering 55

3.1.2 Prototype Development 56

3.1.3 Testing 57

3.2 Research Method 57

3.3 Chosen Development Tools 58

3.3.1 Programming Languages 58

3.3.2 Frameworks 59

3.3.3 Cloud Services 59

3.3.4 Database 60

3.3.5 Version Control 60

3.4 Preliminary User Interface Design 60

3.5 Project Plan 69

4 Project Specification 70

4.1 Requirement Specification 70

4.1.1 Functional Requirements 70

4.1.2 Non-Functional Requirements 72

4.2 Use Case Modelling 73

4.2.1 Use Case Diagram 73

4.2.2 Use Case Description 73

5 Design 91

5.1 Software Architecture Design 91

5.2 Software Component Design 92

5.2.1 Client Components 92

5.2.2 Server Components 93

5.3 Detailed Architecture Design 95

5.3.1 Recommender System Architecture 95

5.3.2 Microservices Architecture 96

5.4 Database Design 97

5.4.1 Entity Relational Diagram 97

5.4.2 Logical Entity Relational Diagram 98

5.4.3 Data Dictionary 98

5.5 User Interface Design 104

5.5.1 View 104

5.5.2 Management 109

5.5.3 Others 112

6 Implementation 114

6.1 Process Flow Diagram 114

6.1.1 Recommender System 114

6.1.2 Push Notification 115

6.1.3 Real Time Implementation 116

6.2 Activity Diagram 117

6.3 Component Hierarchy 117

6.4 RESTful Route Design 118

6.5 Role-based Access Control Diagram 124

6.6 Implementation of cloud services in Amazon Web Services 125

7 Testing and Evaluation **128**

7.1 Functional Testing 128

7.2 Non-Functional Testing 132

8 Conclusion and Discussions **135**

8.1 Conclusion 135

8.2 Limitations 135

8.3 Recommendations 136

REFERENCES **137**

APPENDICES **140**

LIST OF TABLES

Table 2.1: Comparison between online co-curriculum information systems in different universities and colleges
Table 2.2: Advantages and disadvantages of Waterfall methodology
Table 2.3: Advantages and disadvantages of Agile methodology
Table 2.4: Advantages and disadvantages of Evolutionary Prototyping methodology
Table 2.5: Advantages and disadvantages of Rapid Application Development methodology
Table 2.6: Comparison among software development methodologies
Table 2.7: Comparison between React and Angular frameworks
Table 2.8: Comparison between services provided by Google Cloud Platform and Amazon Web Service

Table 6.1: Route Design for *GET* method - 1
Table 6.2: Route Design for *GET* method - 2
Table 6.3: Route Design for *GET* method - 3
Table 6.4: Route Design for *POST* method - 1
Table 6.5: Route Design for *POST* method - 2
Table 6.6: Route Design for *PUT* method - 1
Table 6.7: Route Design for *PUT* method - 2
Table 6.8: Route Design for *DELETE* method - 1
Table 6.9: Route Design for *DELETE* method - 2

Table 7.1: Usability Test Average Score

LIST OF FIGURES

Figure 1.1: Client-Server Architecture of the system
Figure 1.2: Overall System Architecture of the system
Figure 1.3: Evolutionary Prototyping model

Figure 2.1: Student society announcement section in UTAR Portal
Figure 2.2: List of societies registered by student
Figure 2.3: List of committee board of the selected society
Figure 2.4: Societies List of National Yang-Ming University
Figure 2.5: Societies List of HELP University
Figure 2.6: Societies List of Universiti Sains Malaysia
Figure 2.7: Societies List of University of Melbourne
Figure 2.8: Events List of University of Melbourne
Figure 2.9: Events Description of University of Melbourne
Figure 2.10: Society Description of University of Melbourne
Figure 2.11: Events List of UCSI University
Figure 2.12: Societies List of UCSI University
Figure 2.13: Home page of ELE Portal of UCSI University (Proposal submission feature)
Figure 2.14: Home page of ELE Portal of UCSI University (Attendance taking feature)
Figure 2.15: Club Enrolment section of ELE Portal of UCSI University
Figure 2.16: QPAY web application of University of Melbourne
Figure 2.17: QPAY mobile application of University of Melbourne
Figure 2.18: Search result of Societies List of University of Melbourne
Figure 2.19: Overview of Waterfall Model

Figure 2.20: Overview of Agile Methodology
Figure 2.21: Overview of Evolutionary Prototyping methodology
Figure 2.22: Overview of Rapid Application Development model
Figure 2.23: Bar graph of International interest based on React and Angular framework

Figure 3.1: Overview of Evolutionary Prototyping Model

Figure 4.1: UI - Show Home page
Figure 4.2 : UI - Login
Figure 4.3: UI - Show list of available societies
Figure 4.4: UI - Show list of available events
Figure 4.5 : UI - Create Society Profile
Figure 4.6: UI - Show all registered societies
Figure 4.7: UI - Show all registered events
Figure 4.8 : UI - Show detailed information of each society
Figure 4.9: UI - Show detailed information of each event
Figure 4.10: UI - Manage member registration of society
Figure 4.11: UI - Show list of registered participants for each event
Figure 4.12: UI - Manage registered crew for each event
Figure 4.13: UI - Register booth for society's event
Figure 4.14: UI - Register for interested event
Figure 4.15: UI - Show recommendations of suitable societies and events
Figure 4.16: UI - Show list of FAQ

Figure 5.1: Client-Server Architecture of the system
Figure 5.2: System Architecture Design
Figure 5.3: Recommender System Architecture
Figure 5.4: Entity Relational Diagram
Figure 5.5: Logical Entity Relational Diagram

Figure 6.1: Recommender System using Amazon Web Service
Figure 6.2: Notification using Amazon Web Service
Figure 6.3: Socket Implementation
Figure 6.4: Student registration for society / event
Figure 6.5: Component Hierarchy
Figure 6.6: Role-based Access Control Diagram

Figure 7.1: Test script for redux action using Jest and react-thunk
Figure 7.2: Test script for redux reducer using Jest and react-thunk
Figure 7.3: Test script for component rendering using Enzyme
Figure 7.4: Unit Test Result for React
Figure 7.5: Test script for *GET* method
Figure 7.6: Test script for *POST* method
Figure 7.7: Test script for *PUT* method
Figure 7.8: Test script for *DELETE* method
Figure 7.9: Test result for Flask
Figure 7.10: Audited result using Lighthouse

LIST OF APPENDICES

APPENDIX A: Work breakdown structure and Gantt chart 140

APPENDIX B: Use Case Diagram 145

APPENDIX C: Feedback Survey Form 147

APPENDIX D: Feedback Result 154

APPENDIX E: Flow Chart 160

APPENDIX F: Interview Result 170

APPENDIX G: Integration Test Cases 172

APPENDIX H: Performance Testing Report (Lighthouse) 176

APPENDIX I: User Acceptance Test Agreement Form 181

CHAPTER 1

INTRODUCTION

This chapter provides introduction about the basic information of this whole project that will be done at the end of this report. Section 1.1 is about the background of the project. Section 1.2 gives a description of the problem statement of this project. Section 1.3 is about the project objectives while Section 1.4 is about the project goal. Section 1.5 briefly describes about the final solution. Section 1.6 is the approach used to complete this project. Finally, the last section which is Section 1.7 is about the overall project scope of this project.

1.1 Background

The internet is a very important and popular aspect in everyone's daily life nowadays and almost everyone uses it every day at any time, any place and under any condition. The impact that internet brings to everyone's life is very great because it makes everything to be able to work in a very efficient and productive manner. From the perspective of business, mostly everything is turning into online management system rather than still remain in implementing paperwork management system. Today, a lot of online websites are used in many various kinds of fields such as business, education and health because of the existence of internet.

Little, G (2018) stated that a lot of advantages can be found if using an online system in managing business. Everything is better to be integrated into one system which can manage everything well in one central location if you want to analyse your business in a bigger picture. By using this centralized online management system, things can be done in a more simple, cheap, efficient and effective way. Besides, all data can be managed better in a more organized and structured way if using an online system because everything will not be messed up easily. Online management system also helps in reducing workload and the needs of manpower because the online system will do everything for you.

In the aspect of education, the concept is the same as in the aspect of business. An online management system is also beneficial to the related educational institution. Nowadays, an academic portal is definitely necessary for every education institution in managing academic stuff. However, co-curriculum also becomes an important aspect in education nowadays. An online co-curriculum portal is also needed in order to manage all of the co-curriculum stuff efficiently using an online managing system instead of using paperwork system.

1.2 Problem Statement

According to the survey result of UTAR students as shown in Appendix D: Feedback Survey Result, when there is a standard platform for students to have a clear overview about UTAR societies and events and also acknowledge about all latest information about co-curricular activities, 92.2% of the students will be highly encouraged and motivated to involve in co-curricular activities. Besides, when society committees are able to manage their own societies and events through a standard platform, all tasks can be done more effectively and efficiently in a productive manner such as handling registration of new members, bidding for event's booth location, and promoting upcoming events. Furthermore, when staffs are able to always acknowledge about all updates of their respective societies through a standard platform, they can always provide appropriate guidance at all time. When the system is able to provide recommendations to students regarding suitable events for them to join based on their interested past events, this will increase students' interest to join co-curricular activities frequently. In addition, according to Appendix D: Feedback Survey Result, when there is notification sent to remind students about the upcoming events that they had registered, 70% of the students think that this will help to solve their problem by reducing the chances they forget to attend the events.

However, in reality, there is a messy overview regarding information about UTAR societies and events. According to the survey result of UTAR students as shown in Appendix D: Feedback Survey Result, a few problems have been identified as shown below.

1.2.1 Scattered information in social media

Most of the students obtain the co-curriculum information from their friends around or obtain through social media application such as Facebook and WhatsApp. 51% of the students said that these phenomena cause inconvenience to them. Hence, some students are not aware of most of the upcoming events organized by various societies and some existing but non-popular societies due to the ineffective promoting approaches.

1.2.2 Ineffective management

Some staffs are not able to receive latest updates from their respective societies about their recent activities. Society committees are not able to manage their respective societies and events well in a standard platform such as registration of new members and promoting events effectively.

1.2.3 Lack of recommendation for students

There are too many different natures of events organized by various kinds of societies in UTAR. Sometimes, different kinds of events are often coming together at a particular peak period and students might missed out some interesting events or they know about it too late.

1.2.4 Lack of notification for students

54.9% of the students always tend to forget to attend the events that they have registered before or they remember it only on one day before the actual date of the event. There are 61.4% of them have forgotten about the upcoming registered events for 1 to 3 times based on the past experience.

Hence, the solution to overcome all of the problems stated above is to develop a UTAR Societies' Events Information System. When all of the features as stated above are combined and incorporated into a centralized online system, this will become an ideal situation that required by all UTAR students and staffs. Lastly, 70.6% of the students have agreed that a co-curriculum portal is necessary for managing co-curricular activities in order to solve their problems.

1.3 Project Objectives

The objectives of this project are:

1. To plan the methodology to be used to conduct this project.
2. To design framework or model of this project.
3. To implement the project according to the designed framework or model.
4. To test and evaluate the effectiveness and efficiency of the completed solution.

1.4 Project Goal

The goal of this project is to provide a standard platform for UTAR students and staffs to have a clear overview towards societies or events and to improve the efficiency and effectiveness in managing co-curricular information.

1.5 Final Solution

In order to solve the problems that are stated above, some solutions have been determined. An overall system architecture has been designed in order to solve all problems using this system. The concept used to build this system is a client-server architecture. This system is be built as a Progressive Web Application (PWA) which is able to provide an app-like experience to users using modern web capabilities.

The frontend of this system is built using React framework which is easy to learn, use and implement (Willoughby, 2017). React is used to handle the logic flow of the system, screenflow for user interface and the conduct the updates to backend. Cloud services in Amazon Web Services is used in developing this system in order to achieve high scalability and high availability.

The front-end is hosted in Firebase while the backend of this system is deployed and maintained in Amazon Elastic Beanstalk. Flask framework is used to create the RESTful API for the server side which is used to make connection between front-end of the application and the database. The database used to store all data and information of this system is Amazon Relational Database Service which is implementing MySQL.

Next, the recommender system is built using various kinds of services in Amazon Web Service such as Amazon RDS, Amazon EMR, Amazon Lambda and

Amazon S3 to analyse data. The client-server architecture and the overview of the entire system architecture is shown in the Figure 1.1 and Figure 1.2 below.

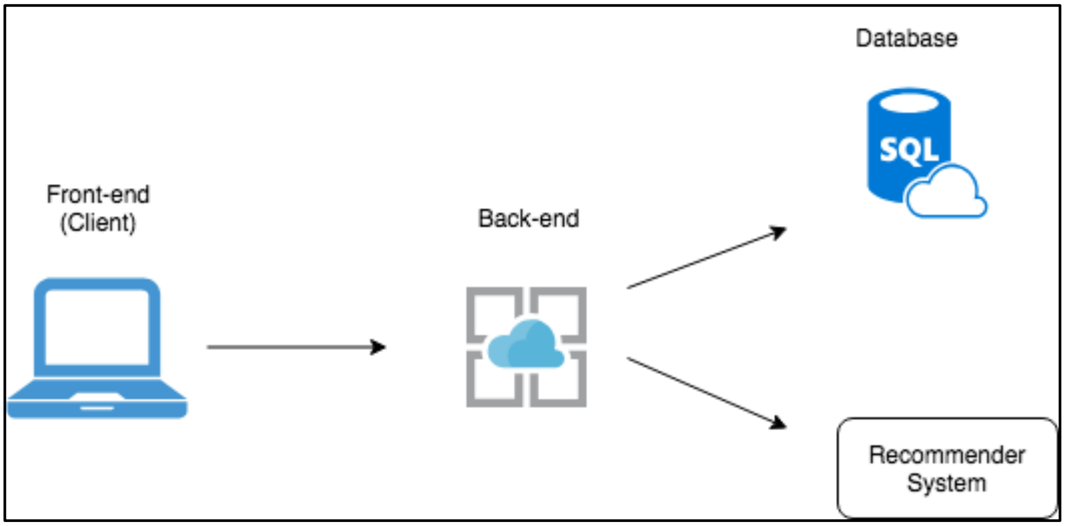


Figure 1.1: Client-Server Architecture of the system

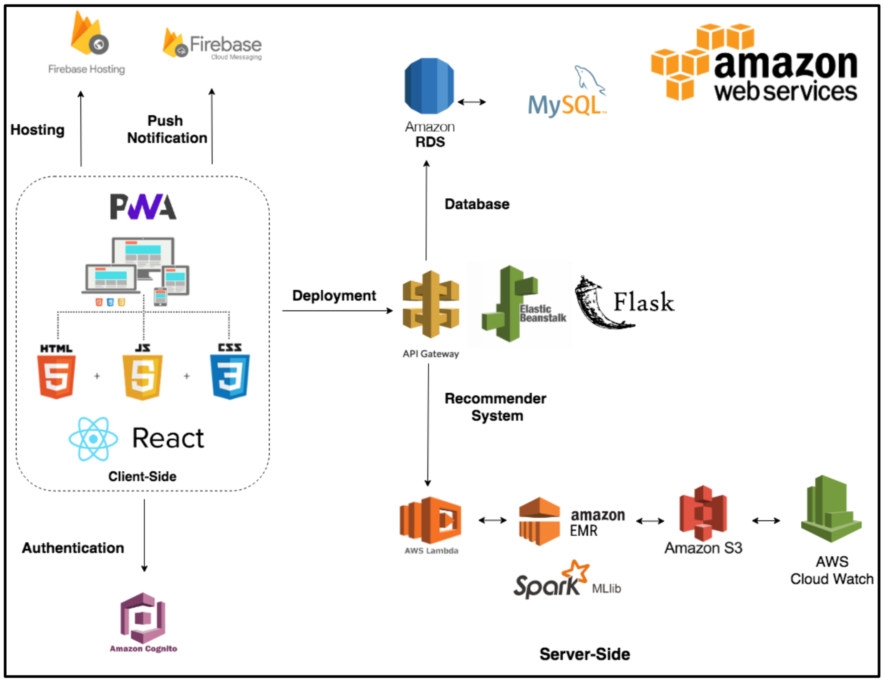


Figure 1.2: Overall System Architecture of the system

1.6 Final Approach

The software development approach that was used to develop this system is Evolutionary Prototyping model. This approach is used to develop a complete and mature system by undergoing a series of iterations and production of prototypes. The initial state of this model is requirements gathering. Next, there is a continuous looping among four phases which are Design, Prototyping, Customer Evaluation and Review. This four phases will be repeated continuously until the final prototype is approved by the end user. The final prototype will then be used as reference to develop the final system. Finally, the final product will be sent for testing and maintaining. (Rouse, 2005). The final product of this project is an information system about UTAR co-curriculum activities such as various kind of societies and events. The overview of the Evolutionary Prototyping model is shown in the Figure 1.2 below.

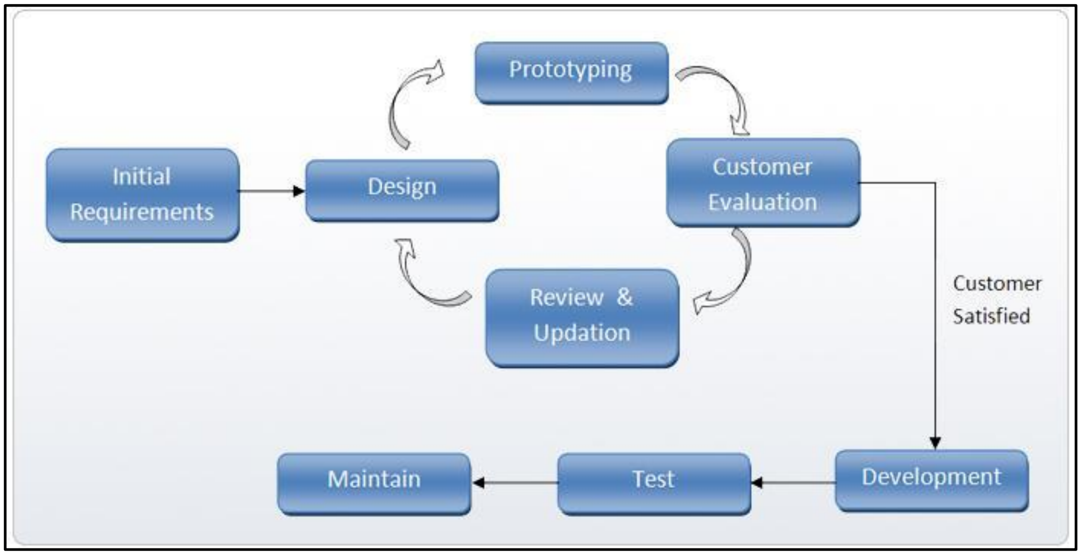


Figure 1.3: Evolutionary Prototyping model

1.7 Project Scope

1.7.1 Target Users

The target users of this information system are all UTAR students who have interest to involve in co-curricular activities and UTAR staffs who are responsible in managing co-curricular information. The users will be able to acknowledge about the latest information of all UTAR co-curricular activities and also manage the information through this standardized information system.

1.7.2 Modules Covered

The modules stated below shall be provided by this information system in order to achieve the project objectives.

1.7.2.1 Displaying list of societies and events

This information system will display a list of existing societies in UTAR and various kinds of events organized in UTAR. Detailed information of each society and event will be displayed when user selects it.

1.7.2.2 Management of societies and events

The specific authorized users such as society committees (Chairperson and Vice Chairperson) are able to manage their own societies and the events organized by their respective societies. For example, society committees are able to manage event participant registration, event crew registration, bid for booth location of events and create their society profile. Besides, staffs are able to manage society membership registration for the respective societies.

1.7.2.3 Society and event registration

The students are able to register for their interested societies and events through this information system. The user are able to fill in their personal details such as name, phone number, course and age for the registration purpose.

1.7.2.4 Event rating

Students are able to submit rating for every participated events. A status will be displayed to indicate whether the user have provided rating for the particular event. Rating system is important as a statistical analysis to all of the event organizers because they need it for future improvement and enhancement. Besides that, the result of the rating system will be used to conclude the performance of every event and will be displayed in the system as an overview for students.

1.7.2.5 Notification

This information system shall be able to provide push notification for students regarding the students' upcoming registered events and updated events from registered societies if the student selected the option of "Allow Notification" during the event registration. The notification will be provided on the actual date of the event at every morning.

1.7.2.6 Recommender System

This information system shall be able to provide recommendation for students regarding the suggested societies to be participated based on the interest of other students due to the ratings provided. Besides, recommendations are provided to students based on the overall rating provided by students in UTAR.

1.7.2.7 Cancellation of event registration and event crew registration

The information system shall be able to allow user to cancel the event registration and event crew registration that had made previously. There are some restrictions on cancelling the registration. Firstly, the registration will not allowed to be cancelled three days before the actual date of the event. Besides, the registration as event crew is not able to be cancelled if the society committees already approved for it.

1.7.3 Modules Not Covered

Due to time constraint and the wide scope of this system, some functions are not able to be delivered on time. For example, payment feature is not provided for students during societies and events registration. Chatbot feature is very convenient and common nowadays and it is often be found in most of the online website for enquiry purpose. However, this feature will not be provided in this system due to limited time given. Both of these features are quite useful and essential for this system and they will be included in the future improvement. In addition, this system will not be built upon native mobile application due to the inconvenience for users to install mobile application. Since the Progressive Web Application is able to provide the mostly similar mobile application experience to users, so it is chosen for this system instead of native mobile application.

CHAPTER 2

LITERATURE REVIEW

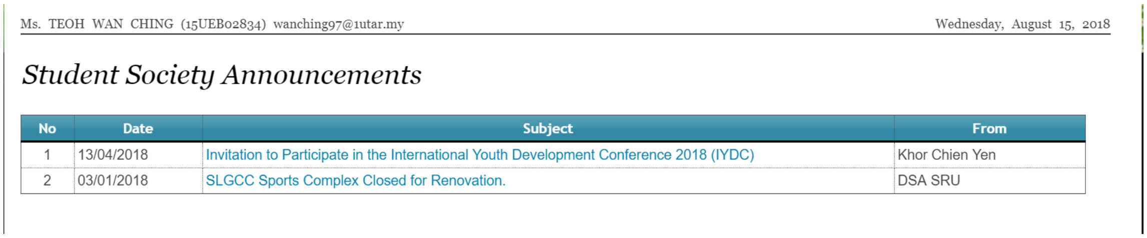
This chapter will introduce about review on existing similar systems in section 2.1. In section 2.2, it is about review of software development methodologies. Review on software development tools such as database, frontend framework, cloud services, native and web application will be covered from section 2.3 to section 2.6.

2.1 Existing Co-curriculum Information System Review

In order to develop an online co-curriculum portal which is able to ensure that all of the basic user requirements and features are fulfilled, a review is done based on the existing online co-curriculum information system in UTAR and also other universities and colleges. There are 10 universities and colleges (included local and overseas) have been analysed and evaluated. The 10 universities and colleges included University of Melbourne, National Yang-Ming University, University of Malaya, Universiti Sains Malaysia, Universiti Teknologi Petronas, Multimedia University, UCSI University, Tunku Abdul Rahman University College, HELP University and Aimst University. Due to the variant degree of emphasis on the aspect of co-curriculum in these universities and colleges, their co-curriculum information system have different features. These 10 universities and colleges are categorized into 3 groups based on the similarity in their system features as stated below.

2.1.1 UTAR Portal

UTAR Portal is mainly used to manage students' academic information such as their respective courses, examination, billing, course registration and academic announcements. Co-curriculum information also involved in this portal, but the information is very limited and insufficient. There is a small section for co-curriculum information which is about society announcement and society committee member. For the society announcement section, only limited announcements will be uploaded, and it is not fully utilized by the management. The figure 2.1 below shows the society announcement section in UTAR Portal.



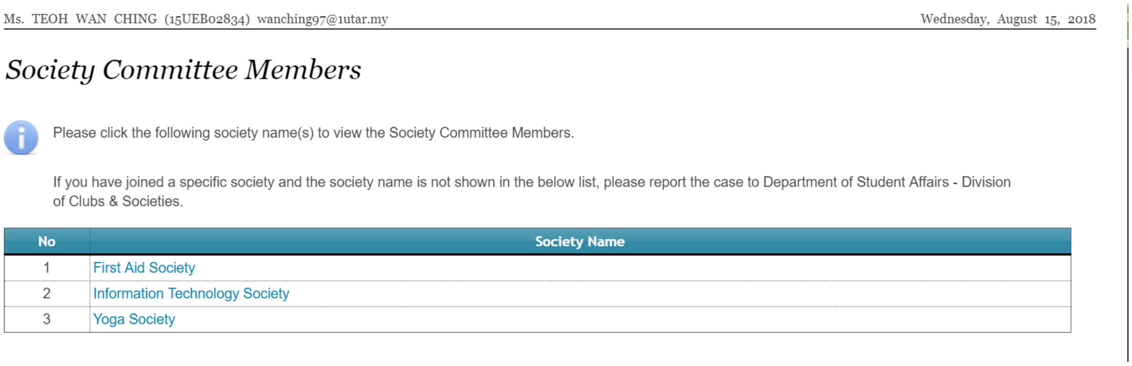
Ms. TEOH WAN CHING (15UEB02834) wanching97@1utar.my Wednesday, August 15, 2018

Student Society Announcements

No	Date	Subject	From
1	13/04/2018	Invitation to Participate in the International Youth Development Conference 2018 (IYDC)	Khor Chien Yen
2	03/01/2018	SLGCC Sports Complex Closed for Renovation.	DSA SRU

Figure 2.1: Student society announcement section in UTAR Portal

Besides, another section about co-curriculum information is society committee members section. This section will only display all societies which are registered by the student and the list of current committee board of the respective societies. The information included in this section is described in Figure 2.2 and Figure 2.3.



Ms. TEOH WAN CHING (15UEB02834) wanching97@1utar.my Wednesday, August 15, 2018

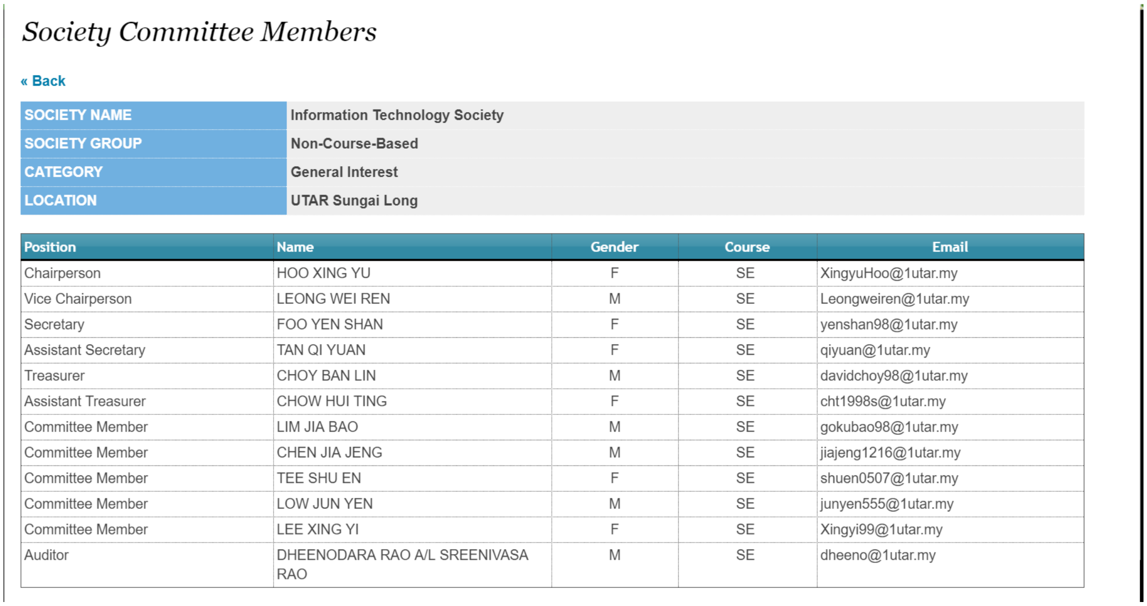
Society Committee Members

Please click the following society name(s) to view the Society Committee Members.

If you have joined a specific society and the society name is not shown in the below list, please report the case to Department of Student Affairs - Division of Clubs & Societies.

No	Society Name
1	First Aid Society
2	Information Technology Society
3	Yoga Society

Figure 2.2: List of societies registered by student



Society Committee Members

« Back

SOCIETY NAME	Information Technology Society
SOCIETY GROUP	Non-Course-Based
CATEGORY	General Interest
LOCATION	UTAR Sungai Long

Position	Name	Gender	Course	Email
Chairperson	HOO XING YU	F	SE	XingyuHoo@1utar.my
Vice Chairperson	LEONG WEI REN	M	SE	Leongweiren@1utar.my
Secretary	FOO YEN SHAN	F	SE	yenshan98@1utar.my
Assistant Secretary	TAN QI YUAN	F	SE	qiyuan@1utar.my
Treasurer	CHOY BAN LIN	M	SE	davidchoy98@1utar.my
Assistant Treasurer	CHOW HUI TING	F	SE	cht1998s@1utar.my
Committee Member	LIM JIA BAO	M	SE	gokubao98@1utar.my
Committee Member	CHEN JIA JENG	M	SE	jiajeng1216@1utar.my
Committee Member	TEE SHU EN	F	SE	shuen0507@1utar.my
Committee Member	LOW JUN YEN	M	SE	junyen555@1utar.my
Committee Member	LEE XING YI	F	SE	Xingyi99@1utar.my
Auditor	DHEENODARA RAO A/L SREENIVASA RAO	M	SE	dheeno@1utar.my

Figure 2.3: List of committee board of the selected society

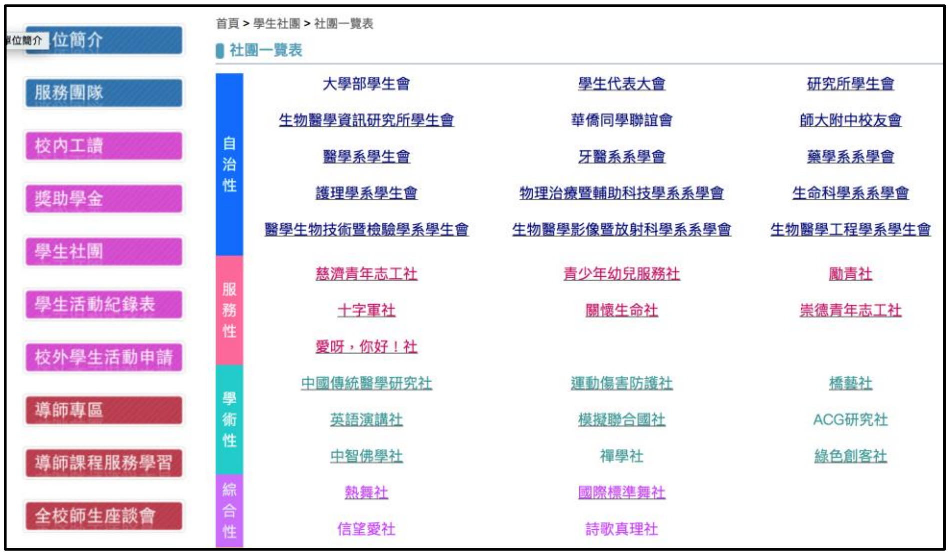
2.1.2 General common features in all universities and colleges

2.1.2.1 Simple list of societies is displayed (exclude societies' events)

All of the universities and colleges have the feature of displaying list of societies in their official website. Some of these universities and colleges link the Facebook account of each society in their official website for convenience purpose. On the other hand, some of the universities and colleges only provide a simple list of societies without additional details or information about the societies. Every university and college display the list of societies according to categories such as Culture-based societies, Course-based societies, Sport-based societies and etc.

2.1.2.1.1 National Yang-Ming University

(i) List of societies according to categories



首頁 > 學生社團 > 社團一覽表

社團一覽表

自治性	大學部學生會	學生代表大會	研究所學生會
	生物醫學資訊研究所學生會	華僑同學聯誼會	師大附中校友會
	醫學系學生會	牙醫系系學會	藥學系系學會
	護理學系學生會	物理治療暨輔助科技學系系學會	生命科學系系學會
	醫學生物技術暨檢驗學系學生會	生物醫學影像暨放射科學系系學會	生物醫學工程學系學生會
服務性	慈濟青年志工社	青少年幼兒服務社	勵青社
	十字軍社	關懷生命社	崇德青年志工社
	愛呀，你好！社		
學術性	中國傳統醫學研究社	運動傷害防護社	橋藝社
	英語演講社	模擬聯合國社	ACG研究社
	中智佛學社	禪學社	綠色創客社
綜合性	熱舞社	國際標準舞社	
	信望愛社	詩歌真理社	

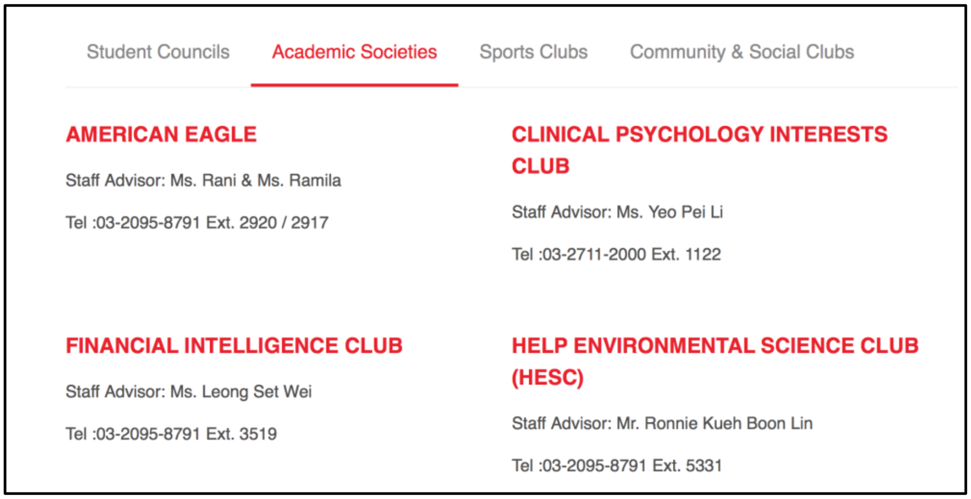
Figure 2.4: Societies List of National Yang-Ming University

Source taken from:

NYMU, n.d. *Extracurricular Activities Section.* [online] Available at: <https://ead.ym.edu.tw/files/11-1204-242.php> [Accessed 30 June 2018].

2.1.2.1.2 HELP University

(i) List of societies according to categories



AMERICAN EAGLE

Staff Advisor: Ms. Rani & Ms. Ramila

Tel :03-2095-8791 Ext. 2920 / 2917

CLINICAL PSYCHOLOGY INTERESTS CLUB

Staff Advisor: Ms. Yeo Pei Li

Tel :03-2711-2000 Ext. 1122

FINANCIAL INTELLIGENCE CLUB

Staff Advisor: Ms. Leong Set Wei

Tel :03-2095-8791 Ext. 3519

HELP ENVIRONMENTAL SCIENCE CLUB (HESC)

Staff Advisor: Mr. Ronnie Kueh Boon Lin

Tel :03-2095-8791 Ext. 5331

Figure 2.5: Societies List of HELP University

Source taken from:

HELP University Sdn Bhd, n.d. *Academic Societies.* [online] Available at: <https://www.help.edu.my/campus-extra-curriculum.html> [Accessed 30 June 2018].

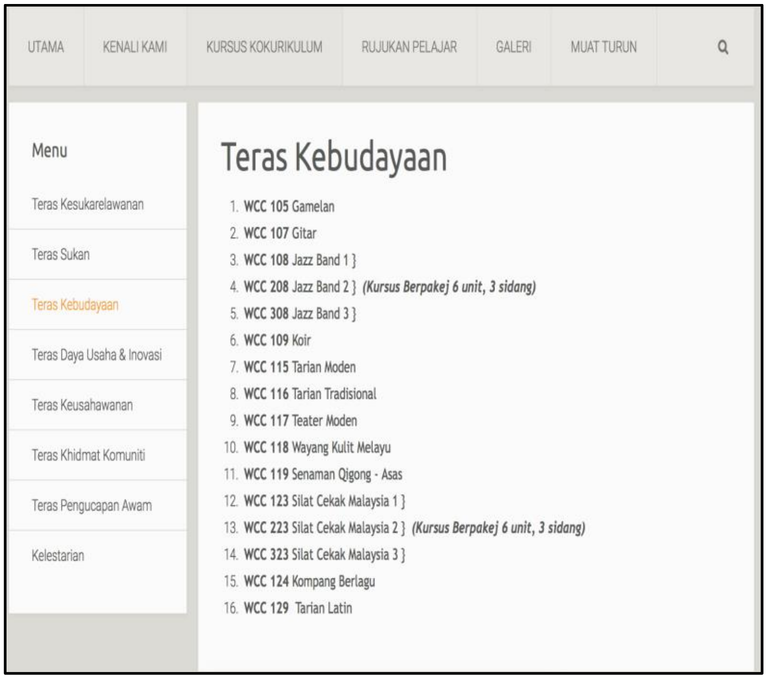
2.1.3 Particular features in other universities and colleges

2.1.3.1 Detailed list of both societies and events are displayed

University of Melbourne, University of Malaya, Universiti Sains Malaysia and Multimedia University are using their official website to list out all existing societies and upcoming events. Some of these universities and colleges provide a clear introduction and information of their societies and events in their website but some of them only provide a simple list without extra information. Since the website is used only for viewing purpose, society staffs and society committees are not allowed to manage their own societies and events in the website. Hence, login is not required for authorization purpose.

2.1.3.1.1 Universiti Sains Malaysia

(i) List of events according to categories (without extra information or details)



Teras Kebudayaan

1. **WCC 105** Gamelan
2. **WCC 107** Gitar
3. **WCC 108** Jazz Band 1 }
4. **WCC 208** Jazz Band 2 } ***(Kursus Berpakej 6 unit, 3 sidang)***
5. **WCC 308** Jazz Band 3 }
6. **WCC 109** Koir
7. **WCC 115** Tarian Moden
8. **WCC 116** Tarian Tradisional
9. **WCC 117** Teater Moden
10. **WCC 118** Wayang Kulit Melayu
11. **WCC 119** Senaman Qigong - Asas
12. **WCC 123** Silat Cekak Malaysia 1 }
13. **WCC 223** Silat Cekak Malaysia 2 } ***(Kursus Berpakej 6 unit, 3 sidang)***
14. **WCC 323** Silat Cekak Malaysia 3 }
15. **WCC 124** Kompang Berlagu
16. **WCC 129** Tarian Latin

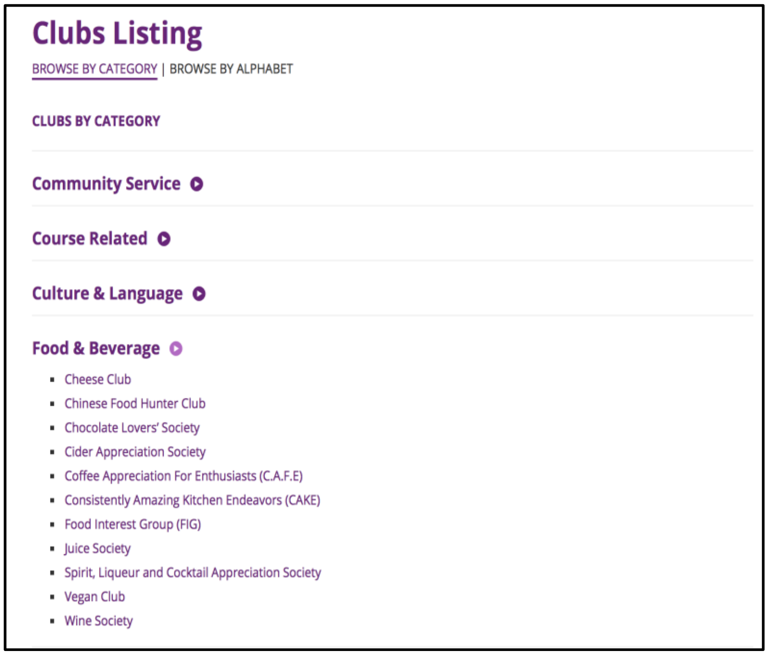
Figure 2.6: Societies List of Universiti Sains Malaysia

Source taken from:

Universiti Sains Malaysia, n.d. *Kursus Kokurikulum.* [online] Available at: <http://kok.usm.my/index.php/kursus-kokurikulum/teras-kebudayaan> [Accessed 30 June 2018].

2.1.3.1.2 University of Melbourne

(i) List of societies according to categories



Clubs Listing

BROWSE BY CATEGORY | BROWSE BY ALPHABET

CLUBS BY CATEGORY

Community Service

Course Related

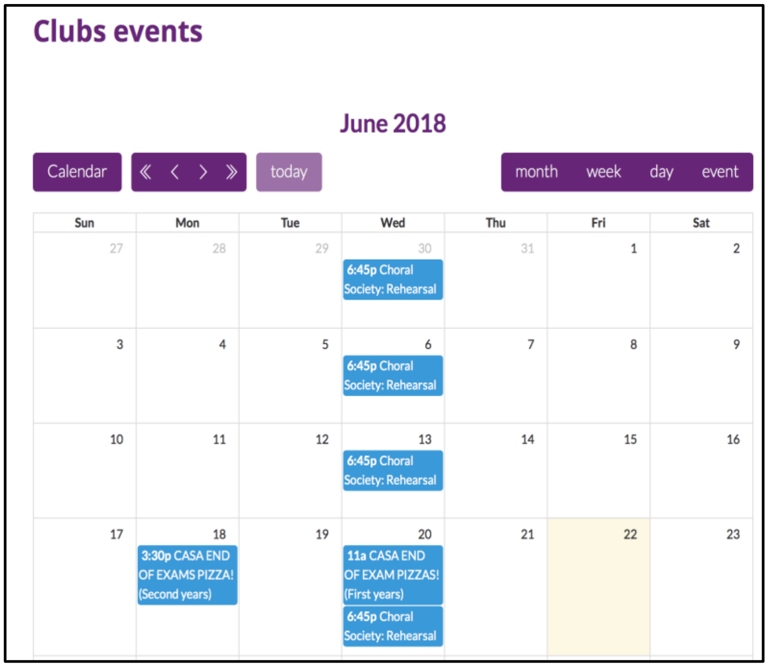
Culture & Language

Food & Beverage

- Cheese Club
- Chinese Food Hunter Club
- Chocolate Lovers' Society
- Cider Appreciation Society
- Coffee Appreciation For Enthusiasts (C.A.F.E)
- Consistently Amazing Kitchen Endeavors (CAKE)
- Food Interest Group (FIG)
- Juice Society
- Spirit, Liqueur and Cocktail Appreciation Society
- Vegan Club
- Wine Society

Figure 2.7: Societies List of University of Melbourne

(ii) List of past and upcoming events including all details of each event



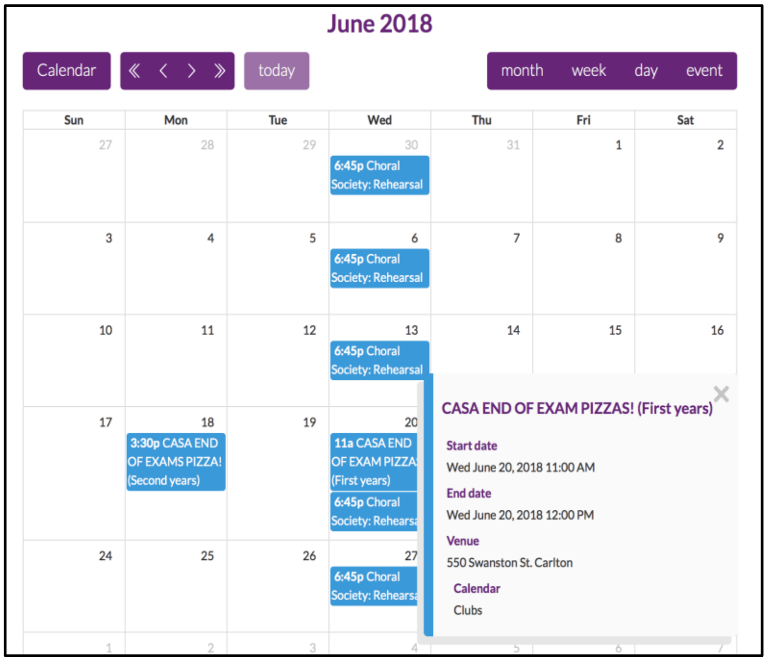
Clubs events

June 2018

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30 6:45p Choral Society: Rehearsal	31	1	2
3	4	5	6 6:45p Choral Society: Rehearsal	7	8	9
10	11	12	13 6:45p Choral Society: Rehearsal	14	15	16
17	18 3:30p CASA END OF EXAMS PIZZA! (Second years)	19	20 11a CASA END OF EXAM PIZZAS! (First years) 6:45p Choral Society: Rehearsal	21	22	23

Figure 2.8: Events List of University of Melbourne

(iii) Information and details for each specific events



June 2018

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30 6:45p Choral Society: Rehearsal	31	1	2
3	4	5	6 6:45p Choral Society: Rehearsal	7	8	9
10	11	12	13 6:45p Choral Society: Rehearsal	14	15	16
17	18 3:30p CASA END OF EXAMS PIZZA! (Second years)	19	20 11a CASA END OF EXAM PIZZA (First years) 6:45p Choral Society: Rehearsa			
24	25	26	27 6:45p Choral Society: Rehearsa			
1	2	3	4	5	6	7

CASA END OF EXAM PIZZAS! (First years)

Start date

Wed June 20, 2018 11:00 AM

End date

Wed June 20, 2018 12:00 PM

Venue

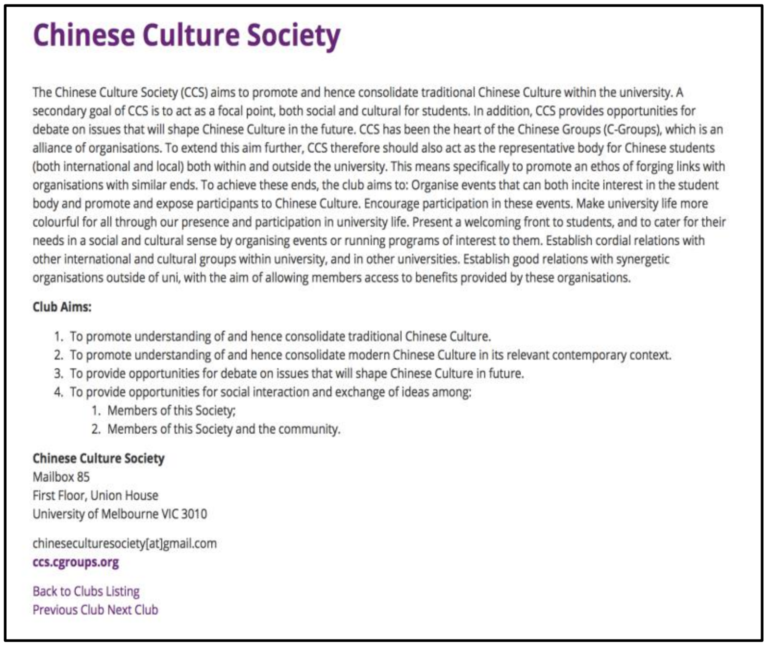
550 Swanston St. Carlton

Calendar

Clubs

Figure 2.9: Events Description of University of Melbourne

(iv) Description of each unique society



Chinese Culture Society

The Chinese Culture Society (CCS) aims to promote and hence consolidate traditional Chinese Culture within the university. A secondary goal of CCS is to act as a focal point, both social and cultural for students. In addition, CCS provides opportunities for debate on issues that will shape Chinese Culture in the future. CCS has been the heart of the Chinese Groups (C-Groups), which is an alliance of organisations. To extend this aim further, CCS therefore should also act as the representative body for Chinese students (both international and local) both within and outside the university. This means specifically to promote an ethos of forging links with organisations with similar ends. To achieve these ends, the club aims to: Organise events that can both incite interest in the student body and promote and expose participants to Chinese Culture. Encourage participation in these events. Make university life more colourful for all through our presence and participation in university life. Present a welcoming front to students, and to cater for their needs in a social and cultural sense by organising events or running programs of interest to them. Establish cordial relations with other international and cultural groups within university, and in other universities. Establish good relations with synergetic organisations outside of uni, with the aim of allowing members access to benefits provided by these organisations.

Club Aims:

1. To promote understanding of and hence consolidate traditional Chinese Culture.
2. To promote understanding of and hence consolidate modern Chinese Culture in its relevant contemporary context.
3. To provide opportunities for debate on issues that will shape Chinese Culture in future.
4. To provide opportunities for social interaction and exchange of ideas among:
    1. Members of this Society;
    2. Members of this Society and the community.

Chinese Culture Society
Mailbox 85
First Floor, Union House
University of Melbourne VIC 3010

chineseculturesociety[at]gmail.com
ccs.cgroups.org

Back to Clubs Listing
Previous Club Next Club

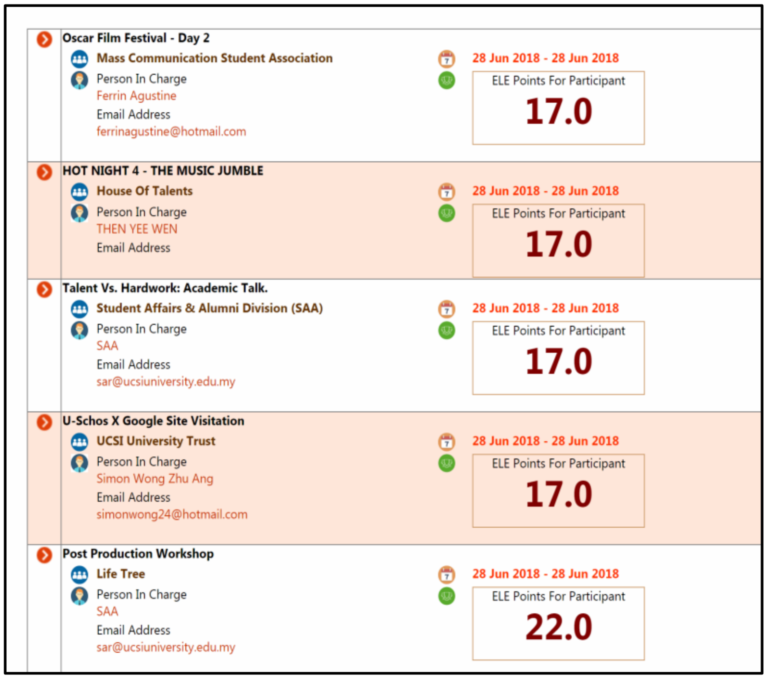
Figure 2.10: Society Description of University of Melbourne

Source taken from:

University of Melbourne, n.d. *Clubs Listing.* [online] Available at: <https://umsu.unimelb.edu.au/getinvolved/clubs/listing/> [Accessed 30 June 2018].

2.1.3.1.3 UCSI University

(i) List of events with detailed information and details



Oscar Film Festival - Day 2
Mass Communication Student Association
Person In Charge
Ferrin Agustine
Email Address
ferrinagustine@hotmail.com
28 Jun 2018 - 28 Jun 2018
ELE Points For Participant
17.0

HOT NIGHT 4 - THE MUSIC JUMBLE
House Of Talents
Person In Charge
THEN YEE WEN
Email Address
28 Jun 2018 - 28 Jun 2018
ELE Points For Participant
17.0

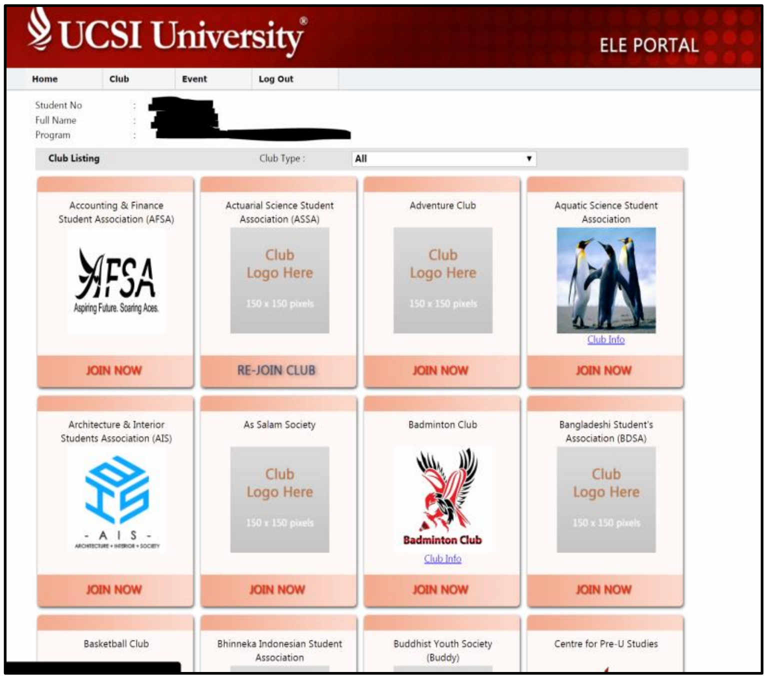
Talent Vs. Hardwork: Academic Talk.
Student Affairs & Alumni Division (SAA)
Person In Charge
SAA
Email Address
sar@ucsiuniversity.edu.my
28 Jun 2018 - 28 Jun 2018
ELE Points For Participant
17.0

U-Schos X Google Site Visitation
UCSI University Trust
Person In Charge
Simon Wong Zhu Ang
Email Address
simonwong24@hotmail.com
28 Jun 2018 - 28 Jun 2018
ELE Points For Participant
17.0

Post Production Workshop
Life Tree
Person In Charge
SAA
Email Address
sar@ucsiuniversity.edu.my
28 Jun 2018 - 28 Jun 2018
ELE Points For Participant
22.0

Figure 2.11: Events List of UCSI University

(ii) List of societies (without categorization)



UCSI University®

ELE PORTAL

Home | Club | Event | Log Out

Student No :
Full Name :
Program :
[redacted]

Club Listing | Club Type : All

- Accounting & Finance Student Association (AFSA) — JOIN NOW
- Actuarial Science Student Association (ASSA) — RE-JOIN CLUB
- Adventure Club — JOIN NOW
- Aquatic Science Student Association — Club Info — JOIN NOW
- Architecture & Interior Students Association (AIS) — JOIN NOW
- As Salam Society — JOIN NOW
- Badminton Club — Club Info — JOIN NOW
- Bangladeshi Student's Association (BDSA) — JOIN NOW
- Basketball Club
- Bhinneka Indonesian Student Association
- Buddhist Youth Society (Buddy)
- Centre for Pre-U Studies

AFSA
Aspiring Future. Soaring Aces.
Club
Logo Here
150 x 150 pixels
- A I S -
ARCHITECTURE • INTERIOR • SOCIETY
Badminton Club

Figure 2.12: Societies List of UCSI University

2.1.3.2 Submission of proposal and proposal review

Every event that organized by any organization has to be approved through a formal proposal which is submitted to the related department. Hence, the submission of proposal is essential as the first step for the success of an event. The co-curriculum information system of UCSI University has this feature such as society proposal and event proposal submission. After the submission, there are additional features provided which are club proposal review and event proposal review. UCSI University fully utilizes the online management system to make all tasks work more efficiently and easily.

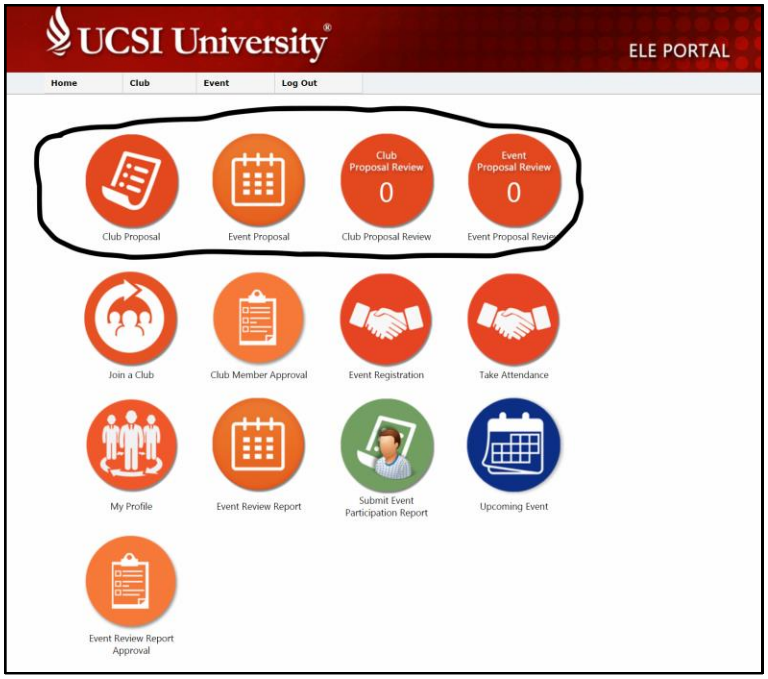


Figure 2.13: Home page of ELE Portal of UCSI University (Proposal submission feature)

2.1.3.3 Taking Attendance

UCSI University is using electronic attendance system in their co-curriculum portal where this system has become a trend in many institution nowadays due to the efficiency and effectiveness. Attendance will be taken before the event starts and it will be managed and evaluated by the respective society after the event ends.

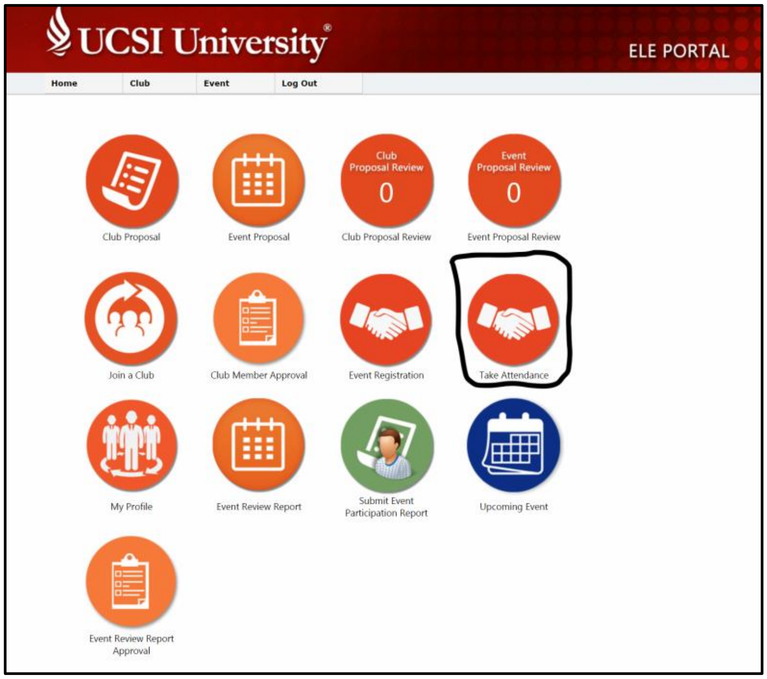
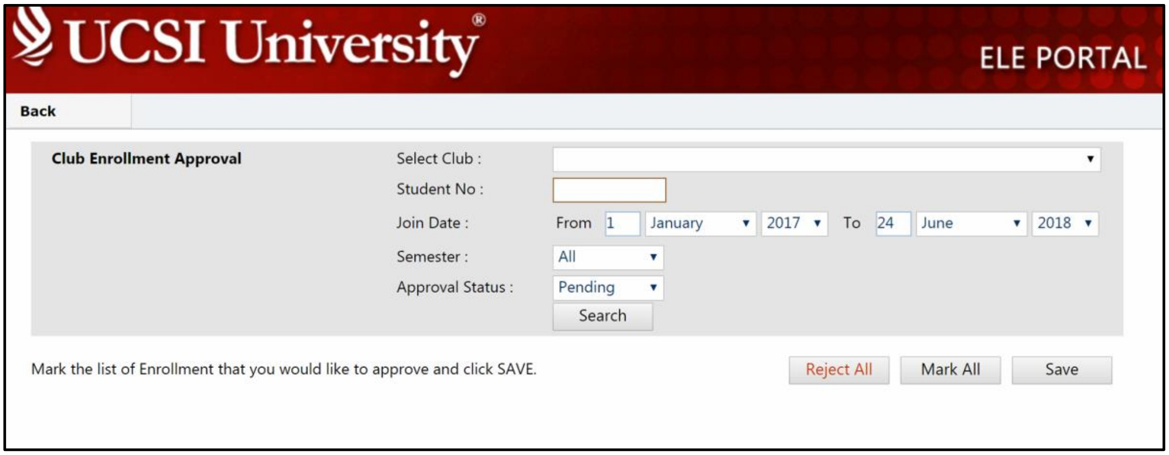


Figure 2.14: Home page of ELE Portal of UCSI University (Attendance taking feature)

2.1.3.4 Event and Member Registration

The co-curriculum portal of UCSI University includes this feature which is more efficient for societies and students compared to member registration using paperwork. This is because staff can view the updates directly through the portal rather than face-to-face meeting the respective people.



Club Enrollment Approval

Select Club :

Student No :

Join Date : From 1 January 2017 To 24 June 2018

Semester : All

Approval Status : Pending

Mark the list of Enrollment that you would like to approve and click SAVE.

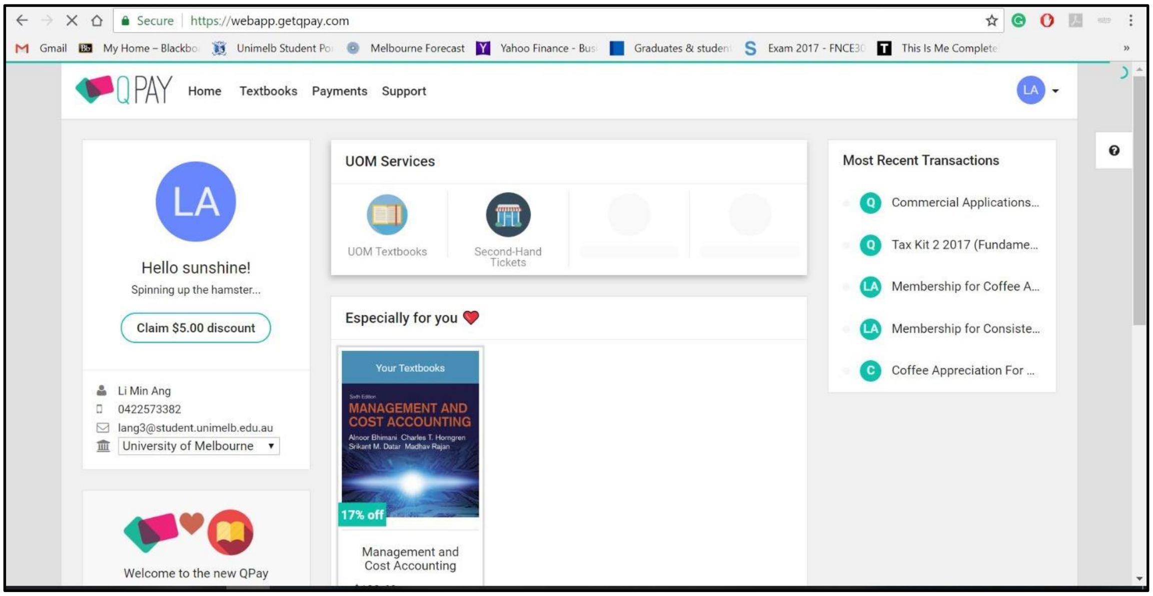
Figure 2.15: Club Enrolment section of ELE Portal of UCSI University

Source taken from:

UCSI University, n.d. *UCSI University.* [online] Available at: <https://www.ucsiuniversity.edu.my/current-students> [Accessed 30 June 2018].

2.1.3.5 Payment System

University of Melbourne is using additional web and mobile application which is known as QPAY application in order to manage the payment of their co-curricular events and other academic transaction. In this application, students are able to view a list of society events and register and make payment for the interested events. The payment system is not included in the online co-curriculum website.



Li Min Ang
0422573382
lang3@student.unimelb.edu.au
University of Melbourne

Figure 2.16: QPAY web application of University of Melbourne

Source taken from:

QPAY, n.d. QPAY. [online] Available at: <https://webapp.getqpay.com> [Accessed 30 June 2018].

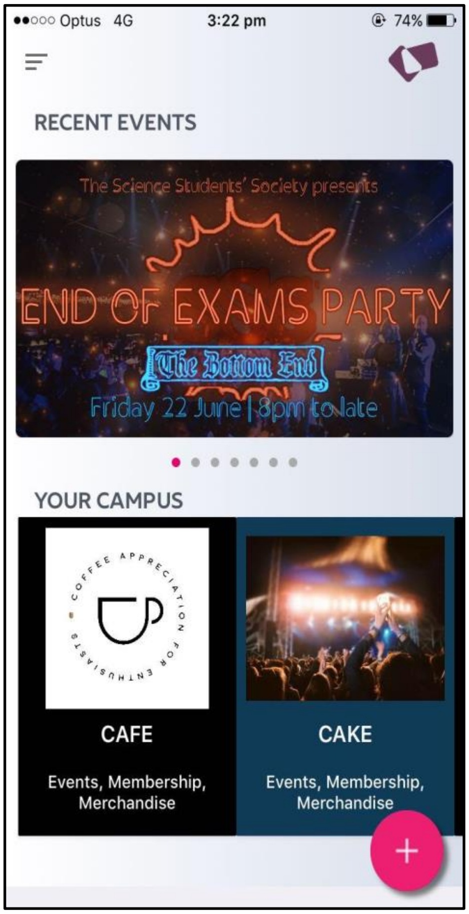

●●○○○ Optus 4G
3:22 pm
74%
RECENT EVENTS
The Science Students' Society presents
END OF EXAMS PARTY
The Bottom End
Friday 22 June | 8pm to late
YOUR CAMPUS
COFFEE APPRECIATION FOR ENTHUSIASTS
CAFE
Events, Membership, Merchandise
CAKE
Events, Membership, Merchandise
+

Figure 2.17: QPAY mobile application of University of Melbourne

2.1.3.6 Conclusion

In short, all of the features that are provided by the existing co-curriculum information system in the universities and colleges (which are stated above) make the system become more useful and convenient to students. However, due to time limitation, only some of these features are able to be included in this system which are section 2.1.2.1 and section 2.1.2.4.

2.1.4 Comparison among existing co-curriculum information system

Table 2.1 Comparison between online co-curriculum information systems in different universities and colleges

UNIVERSITI TEKNOLOGI PETRONAS UCSI University	THE UNIVERSITY OF MELBOURNE UNIVERSITY OF MALAYA USM UNIVERSITI SAINS MALAYSIA MMU MULTIMEDIA UNIVERSITY	NATIONAL YANG-MING UNIVERSITY UTAR HELP University College TARC AIMST UNIVERSITY
Co-curriculum **portal** is exist	Co-curriculum **website** is exist	Co-curriculum **website** is exist
List of **societies and events** are displayed	List of **societies and events** are displayed	List of **societies** are displayed
Login is **required** for the portal	Login is **not required**	Login is **not required**
Staff and society committees are **able** to manage their own societies	Staff and society committees are **not able** to manage their own societies	Staff and society committees are **not able** to manage their own societies

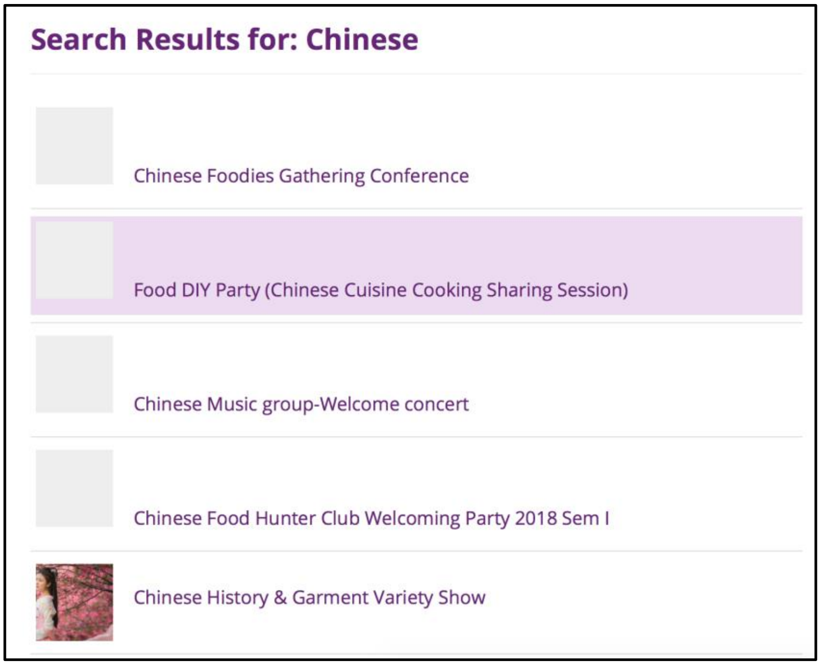
Note: The difference between a portal and a website is that a portal included management of the co-curriculum information while the website is only for viewing information.

2.1.5 Additional included features in the system

This system will include all of the general common features mentioned above in section 2.1.1 which are implemented in the information system of other universities and colleges. The system will also include some features stated in section 2.1.2 such as society or event registration. Other than these features, the system will also include additional features that will make the system becomes more convenient and useful to UTAR staffs and students. The additional features are stated in the Project Scope, which is in section 1.7.2 such as rating feature, recommender system, bidding system and notification. There are another two additional features which are specific search feature and responsive layout as stated below.

2.1.5.1 Specific search function

Students will be able to search through the list of societies and events. The current existing portal in other universities and colleges only consist of overall search function which is search through entire website instead of focusing only societies or events. For example, if student is browsing through a list of societies and wish to search for a particular society with keyword "Chinese", the displayed result should be only consist of related societies and not consist of other information such as related events which are also contain keyword "Chinese". The example is further clarified through the co-curriculum portal in University of Melbourne as shown below:



Search Results for: Chinese

Chinese Foodies Gathering Conference

Food DIY Party (Chinese Cuisine Cooking Sharing Session)

Chinese Music group-Welcome concert

Chinese Food Hunter Club Welcoming Party 2018 Sem I

Chinese History & Garment Variety Show

Figure 2.18: Search result of Societies List of University of Melbourne

Source taken from:

University of Melbourne, n.d. *Clubs Listing.* [online] Available at: <https://umsu.unimelb.edu.au/getinvolved/clubs/listing/> [Accessed 30 June 2018].

2.1.5.2 Responsive layout

By using a progressive web application, students do not need to install an additional mobile application because the layout is responsive in every platform such as tablet, desktop and mobile phone. Installation of mobile application is not preferred by most of the users nowadays due to space limitation in mobile phone. Hence, progressive web application has become a popular trend nowadays.

2.2 Software Development Methodologies Review

2.2.1 Waterfall methodology

The first introduced process model in software development was Waterfall Model which is also known as linear-sequential life cycle model. There are total of six phases in this model, which are Requirements, System Design, Implementation, Testing, Deployment of System and Maintenance. In this model, before proceeding to the next phase, the previous phase must be completed. Hence, there is no overlapping in the six phases in this model (Sharma, 2016). Waterfall Model is often used in small and short projects which consist of clear, well-defined and fixed requirements. Finally, this model will have less interaction with customers (ISTQB Exam Certification, n.d.). The advantages and disadvantages of this methodology are stated in the table below.

Table 2.2 Advantages and disadvantages of Waterfall methodology

Advantages	Disadvantages
It is simple and easy to be used because the phases are not overlapping	Changes are difficult to be made on the application once it comes to the final stage because it works in one-way direction
It is easy to be managed because the model is rigid	A working software cannot be produced and tested until the application reaches the final testing phase.
It works well and efficient in small and	It cannot be used in a complex and

short project because of the simple implementation	object-oriented projects because it is not able to adapt to high uncertainty and risk.

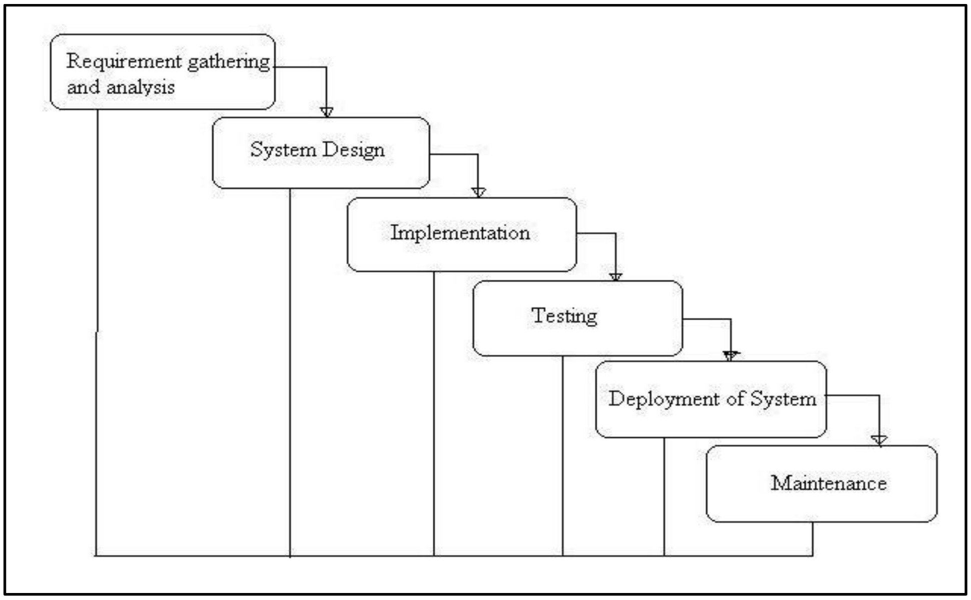


Figure 2.19: Overview of Waterfall Model

Source taken from:
ISTQB Exam Certification, n.d. *What is Waterfall model - advantages, disadvantages and when to use it?* [online] Available at: <http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/> [Accessed 01 July 2018].

2.2.2 Agile software development methodology

Agile methodology is working in an incremental way. Every small incremental process will produce a small released model (also known as increment) which is built on the functionality in previous increment. Each incremental process consist of 5 phases which are Plan, Design, Build, Test and Review. This methodology usually used in time-critical projects where new changes are frequently made and the changes are necessary to be implemented immediately. Low cost is needed for the changes because small increments are often produced time to time (ISTQB Exam Certification, n.d.). The advantages and disadvantages of this methodology are stated in the table below.

Table 2.3 Advantages and disadvantages of Agile methodology

Advantages	Disadvantages
Delivery of a working software is fast and continuous	Less emphasis on a complete and proper documentation and planning
User requirements are always able to be updated due to close interaction between developers, testers and users	Chances of getting out of track are high if the customers are not clear about the final desired outcome
Well and flexible adaptation towards late changes and circumstances	It is not suitable to work on large software deliverables due to the overloaded efforts in the first incremental process

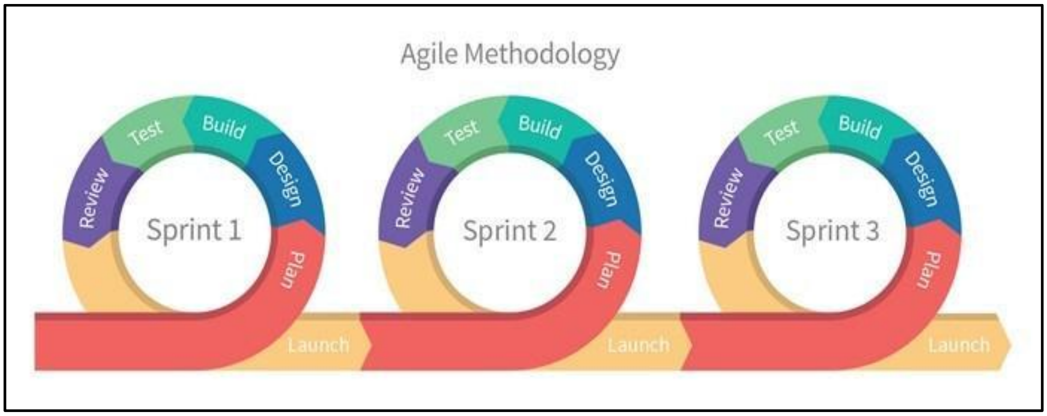


Figure 2.20: Overview of Agile Methodology

Source taken from:

123FormBuilder, n.d. *Our 123FormBuilder team is getting more agile.* [online] Available at: <https://www.123formbuilder.com/blog/2015/02/our-123formbuilder-team-is-getting-agile/> [Accessed 01 July 2018].

2.2.3 Evolutionary Prototyping methodology

In this methodology, a prototype which represents an approximation of the characteristics of final system will be produced, tested and modified continuously until a desired final product is produced. The first prototype is developed with minimal functions and it will evolve into final system by adding additional functions from time to time. First of all, the system requirements will be determined and a preliminary prototype will be developed based on the requirements. Users are involved in this methodology to analyse and evaluate the first prototype. All of the user comments and reviews will be collected and recorded by the developers for next modification. After the first prototype undergoes modification, a second prototype with additional features will be created and sent for evaluation again. The steps above are repeated iteratively until the prototype is satisfied by users. Next, that last prototype will become the reference to develop the final system (Rouse, 2005). After the development phase is completed, the product will be sent for a series of testing such as Unit Testing, Integration Testing and Acceptance Testing. Lastly, the product will be maintained periodically. This methodology is usually be used in the project where some of project requirements are not well-defined in a detailed manner. This methodology is working well in this kind of project due to it is iterative and interacting frequently with users. The advantages and disadvantages of this methodology are stated in the table below (Jamsheer, n.d.).

Table 2.4 Advantages and disadvantages of Evolutionary Prototyping methodology

Advantages	Disadvantages
Clear overview of the overall working prototype is provided for users at the beginning stage	The cost of developing first prototype in the beginning stage is high

Chances of failure in software functionality can be reduced	Development process can be affected and delayed due to excessive involvement of customers
User requirements are always updated due to close interaction with users	Prototype design might constraints designers' ideas

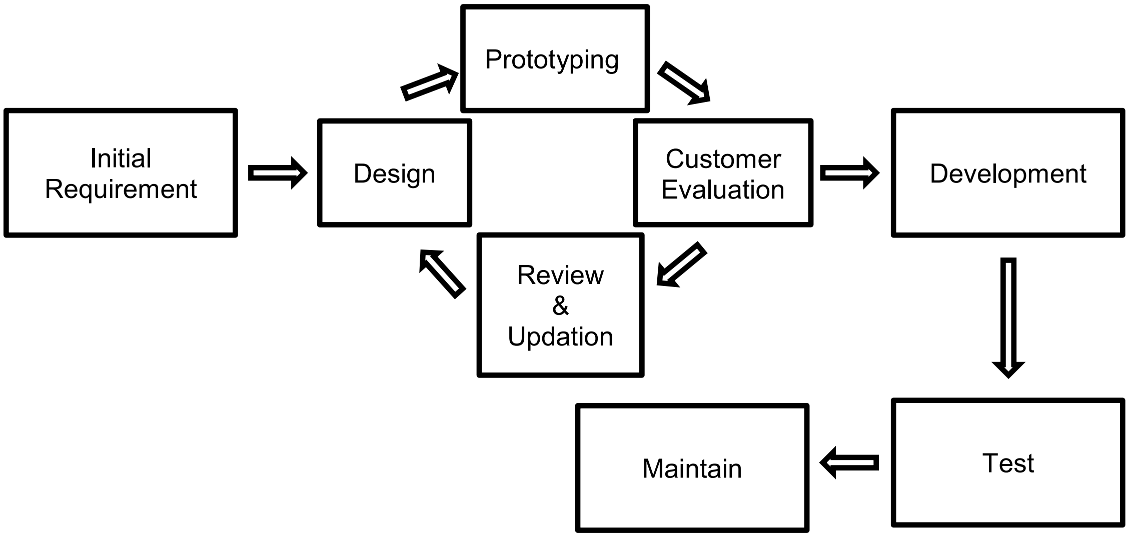


Figure 2.21: Overview of Evolutionary Prototyping methodology

2.2.4 Rapid Application Development methodology

This methodology does not focus on the first common phase in other methodologies which is Planning phase. It focuses on requirement gathering, early testing by users, continuous integration and rapid delivery. It is quite similar to prototype methodology but it is a rapid process. The prototypes produced in every iteration are the functional modules that are modularized from the complete system. Every functional modules are finally integrated to produce the final complete system. This methodology is often used in the system that can be modularized in order to produce prototypes incrementally. The advantages and disadvantages of this methodology are stated in the table below (Tutorials Point, n.d.).

Table 2.5 Advantages and disadvantages of Rapid Application Development methodology

Advantages	Disadvantages
Well and flexible adaptation to changing requirements	Only can be implemented on the system that can be modularized
Reusability of prototypes is high	High demand on highly skilled developers and designers
Development progress can be measured easily	Team members are required to be strong in technical skill to identify business requirements efficiently

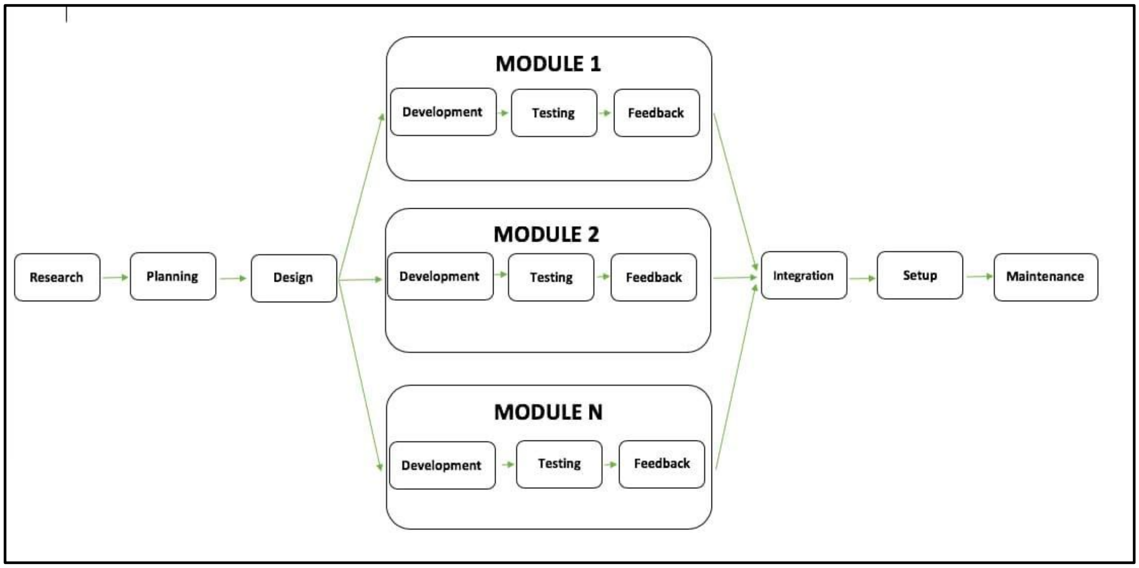


Figure 2.22: Overview of Rapid Application Development model

2.2.5 Comparison among Software Development Methodologies

According to the research from Ratnmala and Haresh (2013), a summary of comparison on several parameters of Waterfall model, Agile model, Prototyping model and RAD model are shown in Table 2.6 below.

Table 2.6 Comparison among software development methodologies

Parameter / Methodology Model	**Waterfall Model**	**Prototyping Model**	**Agile Model**	**RAD Model**
Clear Requirement Specifications	Initial level	Medium level	Change incrementally	Initial level
User Feedback	No	Yes	Yes	No
Speed to Change	Low	Medium	High	Yes
Risk Identification	Initial level	No	Yes	No
Usability	Basic	High	Very High	Medium
Customer Priority	Nil	Intermediate	High	Nil
Elasticity	No	Yes	Very High	Yes
Understand ability	Simple	Intermediate	Complex	Intermediate

2.2.6 Evaluation of the Selected Software Development Methodology

After reviewing 4 types of software development methodologies as stated from section 2.2.1 to section 2.2.4, the respective advantages and disadvantages of each methodology are clearly described. Different types of methodologies are suitable to different characteristics of different projects.

Evolutionary Prototyping methodology is selected as the software development methodology for this project because of the frequent delivery of prototypes and the high involvement of users throughout the development process. User involvement is quite essential for this solution because a lot of different roles of students are included such as students which can be categorized into society committee and society members and also staffs. Besides, since this solution is an information system, it will be used by a huge amount of users. Hence, the usability and learnability of this system are important aspects for every user. Frequent usability testing is needed for further improvement in every prototype according to the collected user feedback. Furthermore, the scope of this solution is considered wide, if the system can be developed slowly starting from small amount of scopes in every iteration and finally integrate them into a final product, then a lot of bugs can be solved in each iteration instead of the final testing phase.

2.3 Databases Review

There are huge differences between SQL and NoSQL databases and they are needed in different kind of system based on its features and characteristics. For instance, is observed by Ashwini (2017) that NoSQL database is needed for the system where there is huge and unstructured data. Besides, speed and scalability are better performed using NoSQL database. For SQL database, it is used for the system where the data integrity is highly emphasized. Furthermore, SQL database stores data in a well-structured and organized manner. Hence, it is more suitable to a system which has complex and inter-related data.

In this system, SQL database is chosen for storing all data because it is a structured, rigid and organized database. SQL database is more considered to be used because the relationship among the data in this system is quite complicated and are interrelated to each other. For example, in this system, a society consists of many members and each member can join many societies. A society can organize many events and each event can be organized by more than one society. Hence, a well-structured and organized database is needed in order to be able to define those relationship more effectively.

2.4 Front-end Frameworks Review

There are two front-end frameworks that have been taken into consideration in developing this system which are React and Angular. Duncan (2017) reported that although JavaScript scene is changing rapidly from time to time, but new technology keep on updating the world nowadays. At this moment, the big players in the aspect of front-end framework seem to be React and Angular. Both of these frameworks have the same effect in the aspects of performance, application size and execution speed. In addition, strong deployment support is provided for both of this frameworks. However, there are also differences between both of this frameworks as stated below:

Table 2.7 Comparison between React and Angular frameworks

React	**Angular**
Rely much on third party libraries such as redux and jQuery.	Does not reply much on third party libraries.
Widely used in start-up or medium sized companies	Widely used in enterprise companies
Focus on React only (or known as ReactJS)	Split into 2 communities (AngularJS or Angular)
Stronger community support such as GitHub (Star and Fork) and Stack Overflow	Moderate community support
Backed by Facebook	Backed by Google
The language used is JSX	The language used is Typescript

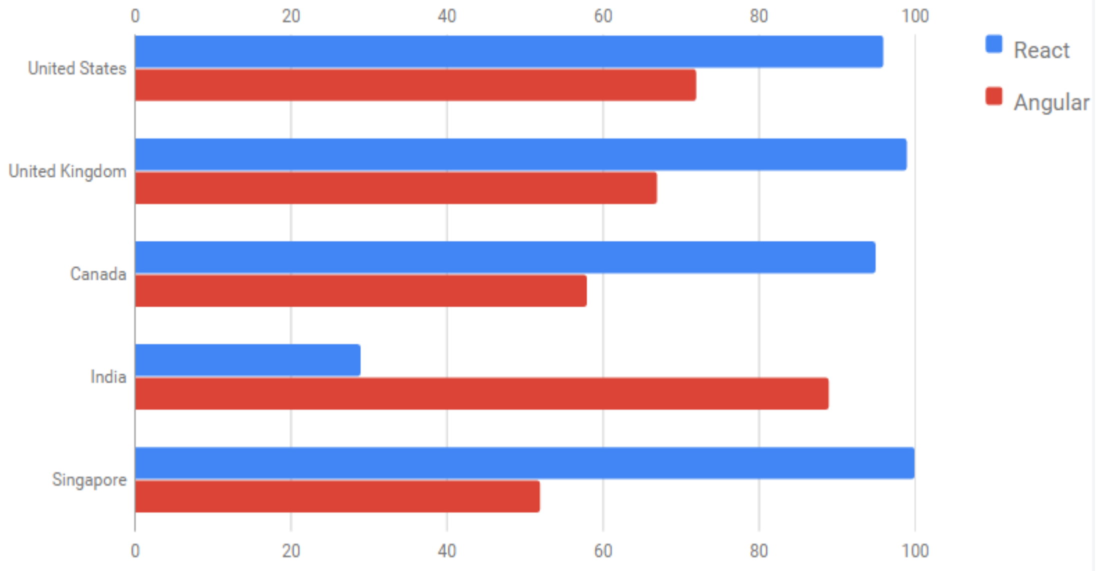


Figure 2.23: Bar graph of International interest based on React and Angular framework

Source taken from:

Korotya, E., 2016. *React and Angular Comparison: Which is Better?* [online] Available at: <https://hackernoon.com/reactjs-vs-angular-comparison-which-is-better-805c0b8091b1> [Accessed 02 July 2018].

In this system, React framework is selected for frontend development is because of the high intention to try a new concept for developing application using component-based architecture instead of following previous experience which is MVC architecture. Besides, this framework is selected is due to the higher familiarity on JavaScript compared with Typescript. Lastly, React framework is also highly supported by beginners and it has higher popularity.

2.5 Cloud Services Review

There are a lot of famous cloud services nowadays such as Google Cloud Service, Amazon Web Service, Azure Cloud Service and IBM Cloud. These cloud service providers are offering high-quality services, high availability, high performance, high scalability, high security and good customer support. However, the cloud services provided that are dominating the marking nowadays are Amazon Web Service, Microsoft Azure and Google Cloud. In this system, the chosen cloud service is Amazon Web Service due to the availability of free credit and new experience in using it. Cloud services are chosen to be used because the high scalability and high availability are important aspects for every system. The services that are related to the features of this system and are provided by both of these cloud providers are stated as shown below (Thorpe, 2018).

Table 2.8 Comparison among services provided by Google Cloud Platform and Amazon Web Service

Services	Google Cloud	Amazon Web Service
Platform-as-a-Service (PaaS)	App Engine (Standard/Flexible Environment)	Elastic Beanstalk
Object storage service	Google Cloud Storage	Simple Storage Services (S3)
Management of relational database	Cloud SQL Cloud Spanner	Relational Database Service (RDS)
Management of data warehouse	Big Query	Redshift
Programmatic access	Cloud SDK Cloud Tools for PowerShell	Command Line Interface

2.6 Native and web applications Review

There are many options of building mobile application nowadays such as Progressive Web App, hybrid app and native iOS and Android apps. But the popularity of native app is getting worse nowadays. Gazdecki (2018) found that most of the users download zero application in a month. The apps that are most frequently used by users are only those social media such as Facebook and Instagram or email apps in their daily routine. Many of them are hesitant in downloading new application from Google Play Store or App Store due to the limited space. A lot of steps need to be done by users in order to use the native apps such as signing up store, checking memory space, downloading and finally installing. Furthermore, Native apps require more resources and time to build, maintain and its cross-platform issue. Hence, native apps are difficult to attract users nowadays.

Ajay (2018) reported that Progressive web app (PWA) becomes the trend in today's technology because it is more reliable, fast and engaging. It combines the best features in web and native apps. It does not require additional installation when user wants to access the application. In addition, it is able to provide interfaces for users even though the internet connection is poor and it is instant-loading. Besides, it provides the same user experience as what native app provides such as home screen shortcut and push notifications. Since PWA is a web apps, it is able to be used in all platforms and this helps in increasing the reachability.

The key success factors of PWA are due to Service Worker and Web App Manifest which are not provided in native app development. Service worker is running in the background which is separated from the main browser. This helps the application to provide a lot of features even when the application is closed or offline and so increase the native app experience for users. An app manifest file is used to describe our app resources which included app's displayed name and icons.

In short, PWA becomes the trend in today's world of technology due to the best combination of features from both web and native apps. Hence, PWA is selected to develop this system instead of native apps.

2.7 Recommender System Review

Isinkaye, Folajimi and Ojokoh (2015) stated that the growth of digital information and the amount of visitors' access to the Internet causes a huge challenge which is information overload on the Internet. Although some of the information retrieval systems such as Google, Altavista and DevilFinder have solved this problem partially, but one critical issue is still existing which is lacking of prioritization and personalization. Personalization means the information retrieval systems provide available content based on user’s interest or preferences. Hence, the demand of recommender system is increasing nowadays. Recommender system is a system which is able to filter information out of large amount of dynamically generated information based on user’s interests, preferences or observed behaviour about items and this helps to solve the information overload issue.

There are two more popular and easier techniques in developing a recommender system which are Collaborative Filtering (CF) and Content-Based Filtering (CBF). CBF is an algorithm which analyses attributes of items to generate predictions. User profile will be used in this technique where the features will be extracted from the content of items in user profile. According to Michael and Daniel (n.d.), a content-based recommender system is not able to provide desired recommendations if there is insufficient information to differentiate between the items that user likes and user doesn’t like in the content. CBF will become more effective if the availability of descriptive data is high.

Due to the limitations of CBF, CF is introduced. It is introduced as a domain-independent prediction technique which is suitable to be used for content that cannot be described by metadata easily such as movies and music. CF is a technique which works by building a database in user-item matrix based on users’ preferences on items. In addition, Xiaoyuan and Taghi (2009) concluded that CF is able to produce recommendations and predictions with high quality if several challenges such as data sparsity, scalability, cold start and synonym found in CF can be addressed successfully.

In short, different techniques will be chosen to be used based on the characteristics of the recommendations intended to be provided. If there is sufficient amount of descriptive data, then CBF is the best technique to be used for predicting and recommending. Otherwise, CF will be a better choice. In this system, CF will be chosen to develop the recommender system because of the data involved in recommendation is not described in metadata. There is no detailed description and information in the data involved.

CHAPTER 3

METHODOLOGY AND PROJECT MANAGEMENT

Section 3.1 will introduce about the detailed phases involved in the software development methodology which is used in developing this project. Section 3.2 introduces about the research method used to collect user requirements in this project. The development tools which are used to develop this project are stated in last section which is Section 3.3.

3.1 Chosen Software Development Methodology

After reviewing 4 types of software development methodologies as stated from section 2.2.1 to section 2.2.4, the respective advantages and disadvantages of each methodology are clearly described. Different types of methodologies are suitable to different characteristics of different projects. Evolutionary Prototyping methodology is finally selected to be used to develop this system due to several reasons as stated below.

This information system involves many different backgrounds of users such as students, society committees and staffs. Each of them has different roles in using this system. Hence, a lot of discussion and interaction with users are needed in the process of developing this system.

Besides, this methodology was selected because of the frequent delivery of prototypes. Prototypes are necessary to be delivered to users for reviewing and evaluating in order to make next improvements. This also helps in enabling the additional requirements provided by customers can be added into the prototypes at any time. In addition, bugs can be defined and fixed earlier before causing any crucial consequences through this incremental process. The overall phases which involve in the Evolutionary Prototyping model is shown in Figure 3.1 below.

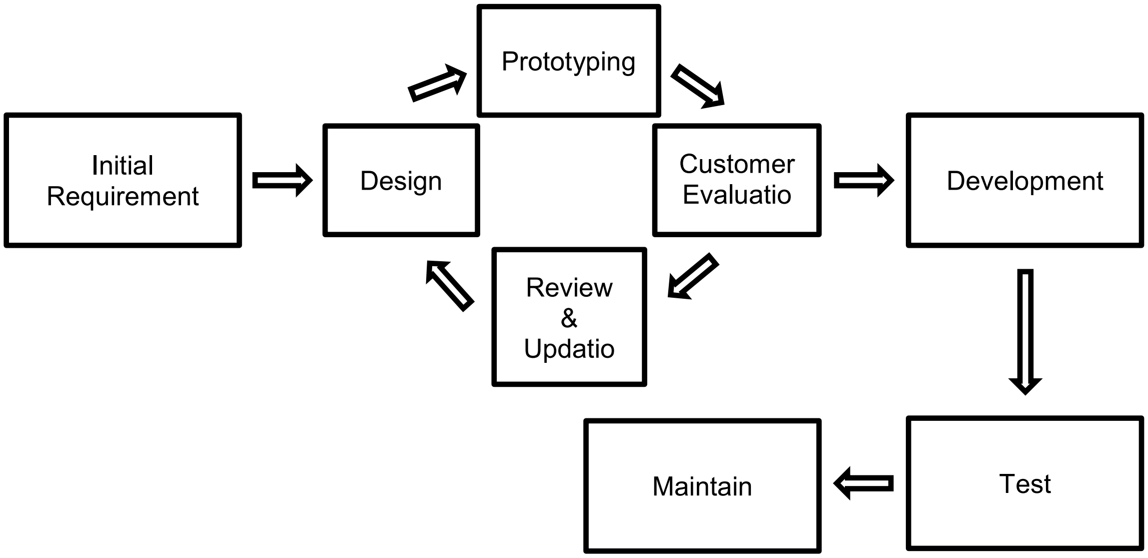


Figure 3.1: Overview of Evolutionary Prototyping Model

3.1.1 Requirements Gathering

During this phase, requirements of this project are collected through a series of research. The research is mainly focus on the features provided by some similar existing co-curriculum information system, the features which are required but are not provided by the similar existing co-curriculum information system, the workflow of co-curriculum management and the overall project scopes of this system. The research was carried out by using two methods which are literature review and survey using questionnaire.

Literature review was done on several similar existing co-curriculum information system in order to further understand the features provided those systems. Throughout this research method, all general common features which are provided by all existing systems are able to be identified. In addition, some additional features which are provided by certain existing systems also can be identified and reviewed in order to determine the suitability to include them in this system.

Next, the purpose of carrying out survey using questionnaire is to seek for opinions students from UTAR and other universities and colleges regarding the situation and problems faced by them in the aspect of co-curriculum. Besides, the survey was carried out to obtain more suggestions from those students about the additional features of this system. The questionnaire was designed using various types of questions such as multiple-choice questions, dichotomous questions and open-ended questions. A large amount of respondents can be obtained in a short period of time by using this method and this situation helps in improving the accuracy and reliability of the final statistic result after the survey is completed. Lastly, the student feedback obtained from this method helps in validating the importance of the features which will be implemented in this system.

Lastly, functional and non-functional requirements of this system were determined. Use case modelling was developed in this phase through Use Case Diagram and Use Case Description. Use Case Diagram have been designed to represent the system behaviour and Use Case Description have been determined to further elaborate every use case.

3.1.2 Prototype Development

There are four continuously repeating phases in developing prototypes which are Design, Prototyping, Customer Evaluation and Review. For this project, there are three iterations of prototype development phase.

3.1.2.1 First Iteration

In the first iteration of prototype development phase, only front-end development was involved while back-end development is excluded. Front-end development mainly focused on designing user interface and the overall screen flow of the system. During design phase, a system architecture diagram was designed at the beginning. Next, a draft of system screen flow and user interface were designed in order to develop the first prototype. For prototyping phase, a prototype was built to show initial user interface and screen flow of the system based on the system architecture diagram and the draft of screen flow. These steps were achieved by using React framework. During customer evaluation phase, the first completed prototype was sent to users for first usability testing. All feedbacks and comments from users were recorded. Lastly, during review and updating phase, the collected feedbacks and comments from users were reviewed. The first prototype was further revised and improved based on the collected feedbacks from users until it meets all of the updated user requirements. After the refinement, second iteration started.

3.1.2.2 Second and Third Iteration

In the second and third iteration of prototype development phase, back-end development began such as developing various features and functions of the system. During the design phase, UML diagrams such as activity diagrams, sequence diagrams, class diagrams and ERD diagrams had been drawn to show the detailed workflow of the whole system. During prototyping phase, user interface and screen flow which involved in the first iteration were further developed in order to be compatible with the back-end development. The back-end of the system was developed by using framework such as Flask framework and cloud services such as Amazon Web Services. The major back-end development involved implementing database, developing bidding system, recommender system, RESTful API and user authentication. During the design phase of third iteration, cleaning of code was involved in order to make the code looks neat and can be read easily.

During customer evaluation phase, the second and third prototypes were sent to user again for usability testing and evaluation. Lastly, during review and updating phase, the prototypes were refined and improved again according to the recorded user feedbacks and comments. The prototypes were keep on refining until it meet all latest requirements from users. When the third prototype is completed, it was developed into the final system and this final system was sent to testing phase.

3.1.3 Testing

During this phase, a series of testing had been carried out to ensure that the final completed system was able to produce desired outcome. Unit testing and integration testing were the two main testing approach for this final completed system. After all testing were passed, then the tested system was ready for implementation.

3.2 Research Method

The research method which was used to collect user requirements in this project is survey using questionnaire. McLeod (2018) justified that questionnaire method is economical because we are able to obtain large amount of research data by using low cost. Besides, the information we collected from targeted respondents are able to be converted into quantitative data which can be used for statistical analysis. In addition, reliability and consistency of the final result can be checked easily because the questions we set for the survey are standardized and the questions are exactly same to all respondents. For this project, there are 13 questions set for the questionnaire and is sent to total of 60 UTAR students who are often involving in UTAR co-curricular activities actively and 5 questions set for 6 students from other universities and colleges.

3.3 Development Tools

This system is a web application system, hence web development tools are required to develop this system.

3.3.1 Programming Languages

Programming languages are the core components of developing any software application. There are several factors which need to be considered when choosing suitable programming languages to be used when developing a project. For examples, the important factors are familiarity, efficiency and the complexity. The programming languages which are selected to be used in developing this project are stated below.

3.3.1.1 SQL

SQL stands for Structured Query Language which is used for management of database such as retrieving, storing and manipulating data in databases (W3Schools, n.d.). SQL is mainly used to manage the transaction of various kind of data involved in this system.

3.3.1.2 JavaScript XML (JSX)

JSX is an object-oriented and statistically-typed programming language. It is a combination of HTML and JavaScript. It is used together with the React framework which is involved in the front-end development of this project. It is faster due to the optimization while compiling code to JavaScript. Besides, it is type-safe because the errors are usually be caught during compilation and not run-time (Tutorials Point, n.d.).

3.3.1.3 Python

Python is an object-oriented and interpreted programming language. It is popular due to the clear syntax, low complexity and high readability.

3.3.2 Frameworks

3.3.2.1 React

React is a frontend framework which is mainly built upon the concept of reusable components, states and props. Many small components are built and combined into several big components and these big components are able to be used across different projects (Buna, 2017).

3.3.2.2 Flask

Flask is a Python web framework which is small and powerful. It is able to be learned and used easily when developing a web application in a short period of time (Polepeddi, 2013).

3.3.2.3 Bootstrap

Bootstrap is a HTML, CSS and JavaScript framework which is used to develop responsive and mobile-first websites (W3Schools, n.d.).

3.3.3 Cloud Services

3.3.3.1 Amazon Web Service

AWS is also one for the most popular and highly demanded Cloud Computing service provider. AWS provides services for various kinds of aspects such as analytics, compute, storage, database, machine learning and IoT.

3.3.4 Database

3.3.4.1 MySQL Database

MySQL is a reliable, performant and simple open source database. It has become the leading database in developing web-based applications and is used by Facebook, YouTube and Twitter.

3.3.4.2 Firebase

Firebase is a Backend-as-a-Service provided by Google and it offers a lot of useful features. Firebase Storage is used in this system for bidding system. It is a NoSQL database which is able to be updated in real time.

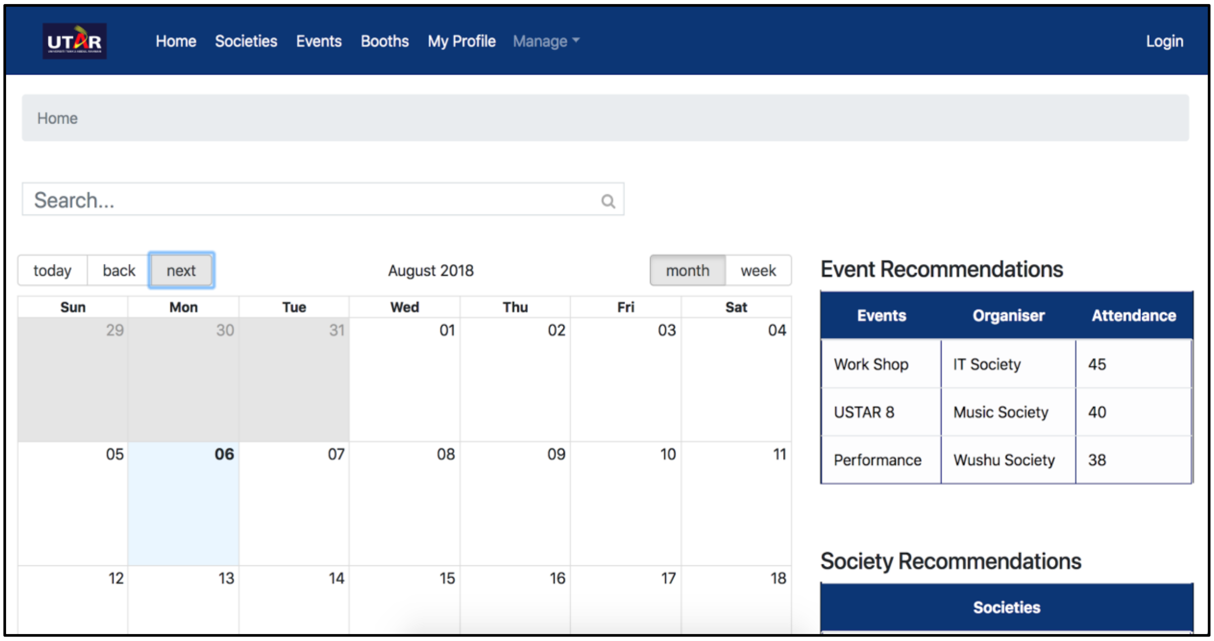
3.3.5 Version Control

3.3.5.1 Git

Git is a modern version control system nowadays. It is used to record all changes that have made in every file of the project so that the project version can be changed at any time due to modification made. Version control is necessary to be used in developing this system is because there are a lot of modification made throughout the development.

3.4 Preliminary User Interface Design

A preliminary user interface design is completed in the first iteration of project development. The user interface is completed using React framework. The screen flow of user interface is further described in the flow chart as attached in Appendix E: Flow Chart.



Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31	01	02	03	04
05	06	07	08	09	10	11
12	13	14	15	16	17	18

Event Recommendations

Events	Organiser	Attendance
Work Shop	IT Society	45
USTAR 8	Music Society	40
Performance	Wushu Society	38

Society Recommendations

Societies

Figure 4.1: UI - Show Home page (calendar view shows all past and upcoming events)

Note: The following pictures of the user interfaces will exclude the top navigation bar.

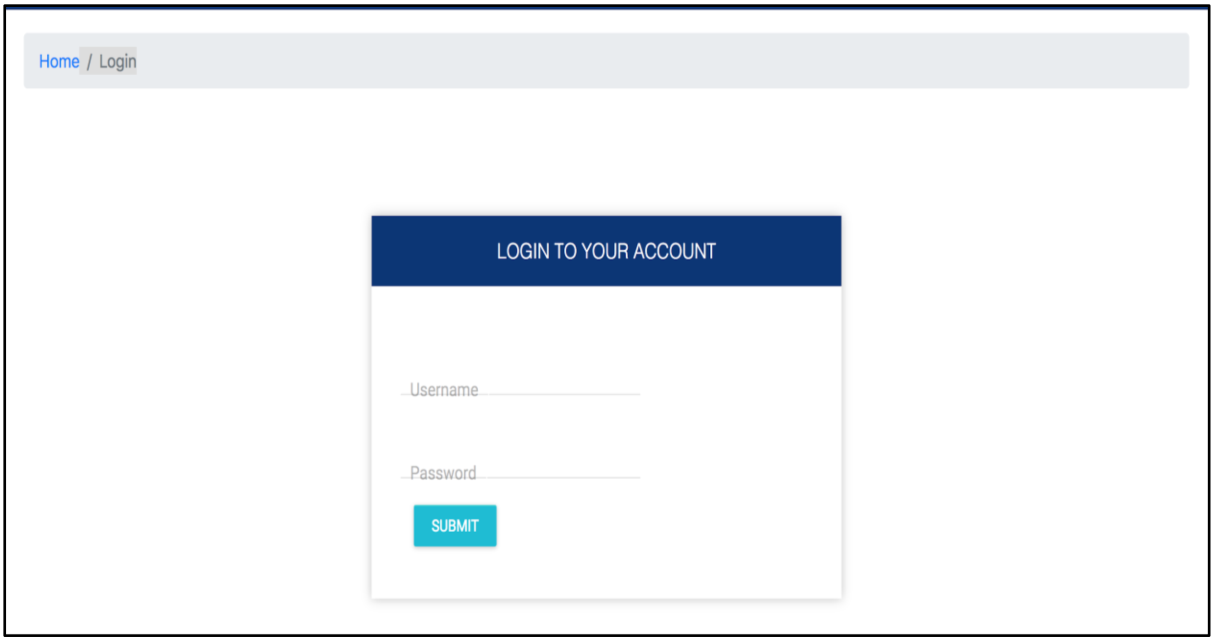


Figure 4.2: UI – Login

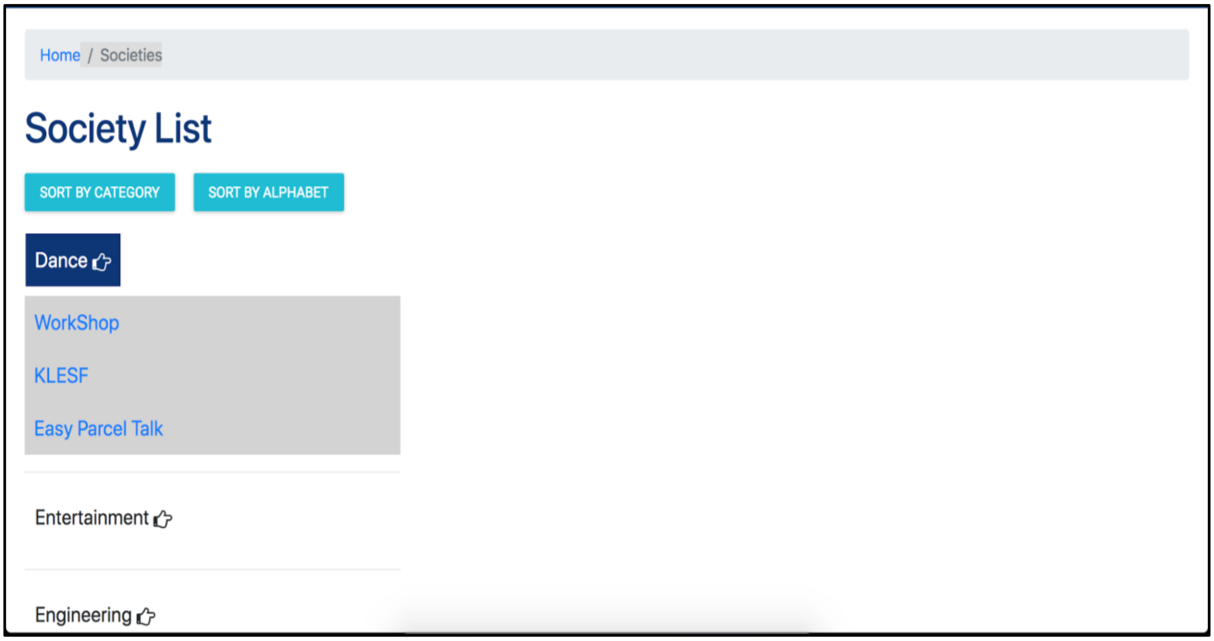


Figure 4.3: UI - Show list of available societies (categorized according to society categories)

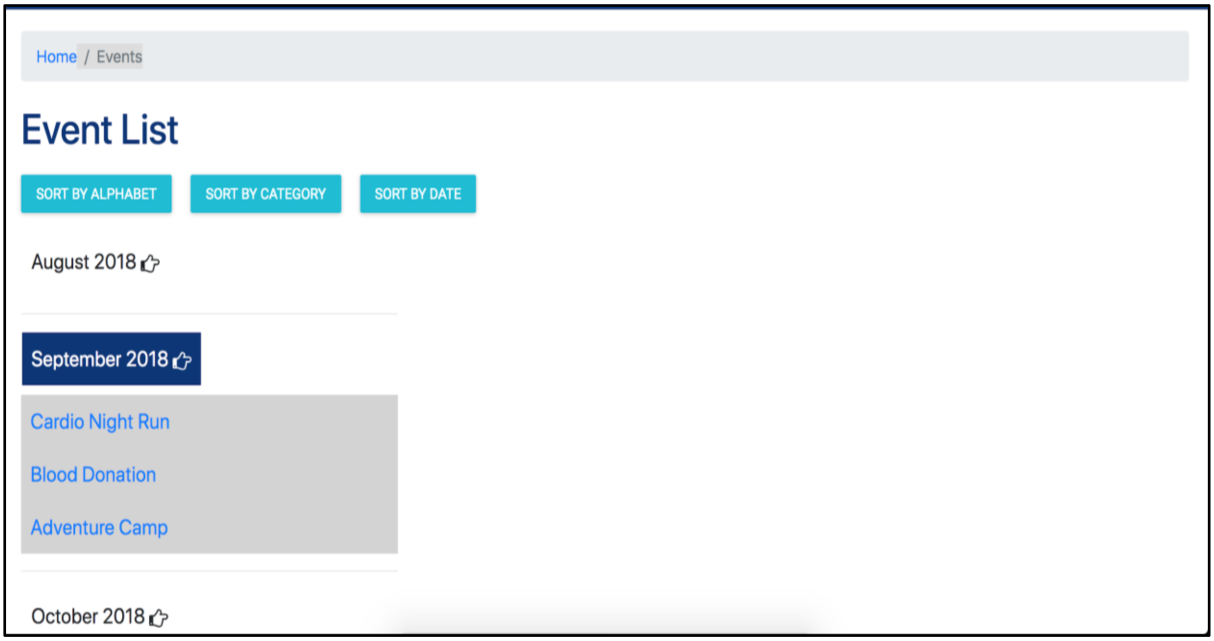
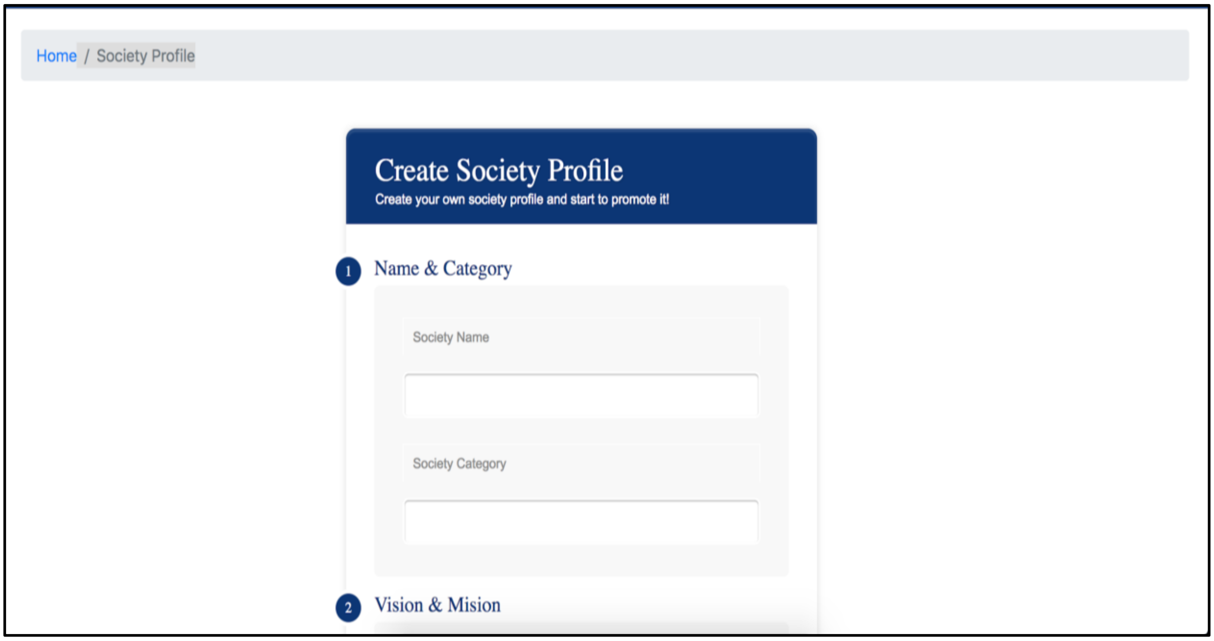


Figure 4.4: UI - Show list of available events (categorized according to month and year)



Home / Society Profile

Create Society Profile

Create your own society profile and start to promote it!

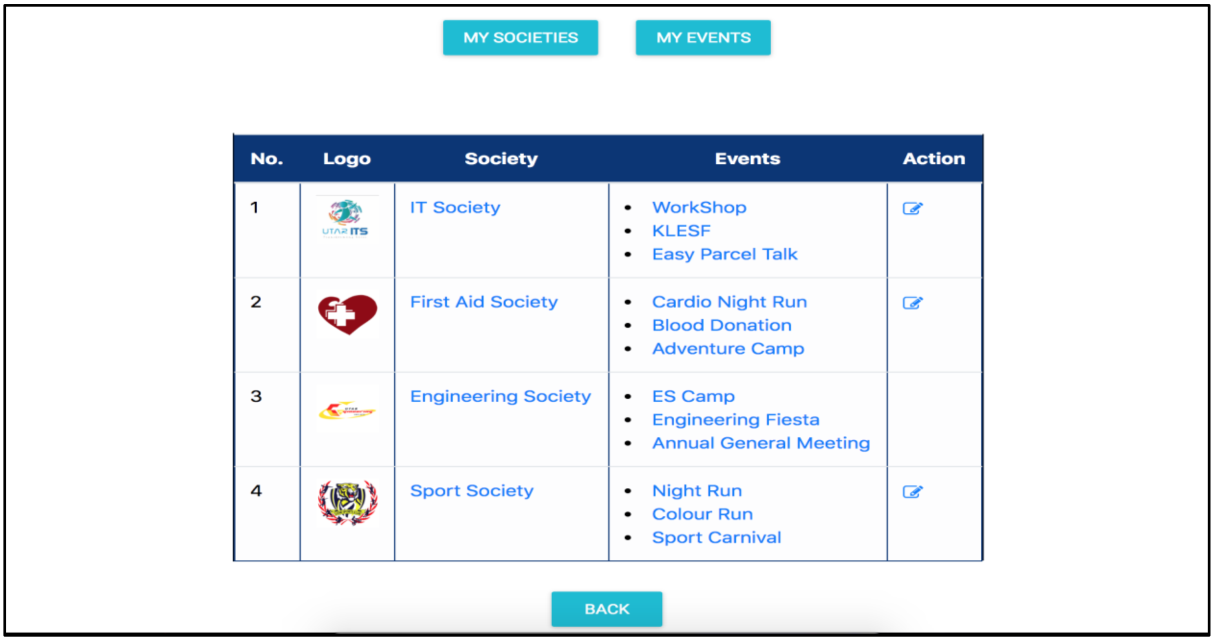
1 Name & Category

Society Name

Society Category

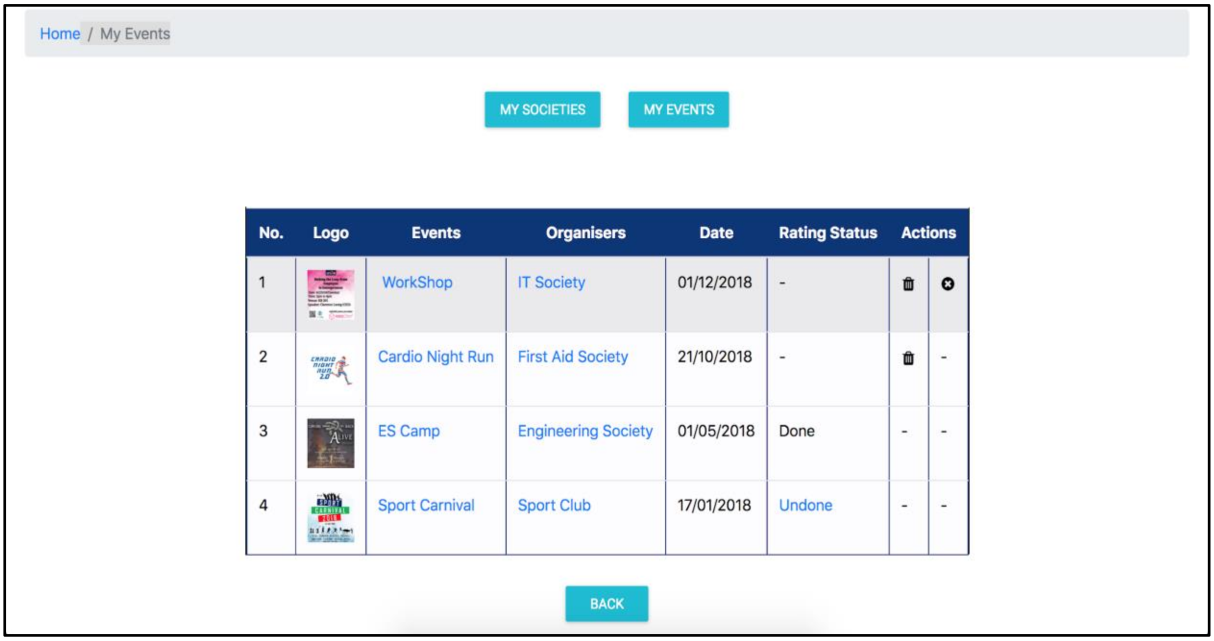
2 Vision & Mision

Figure 4.5: UI - Create Society Profile



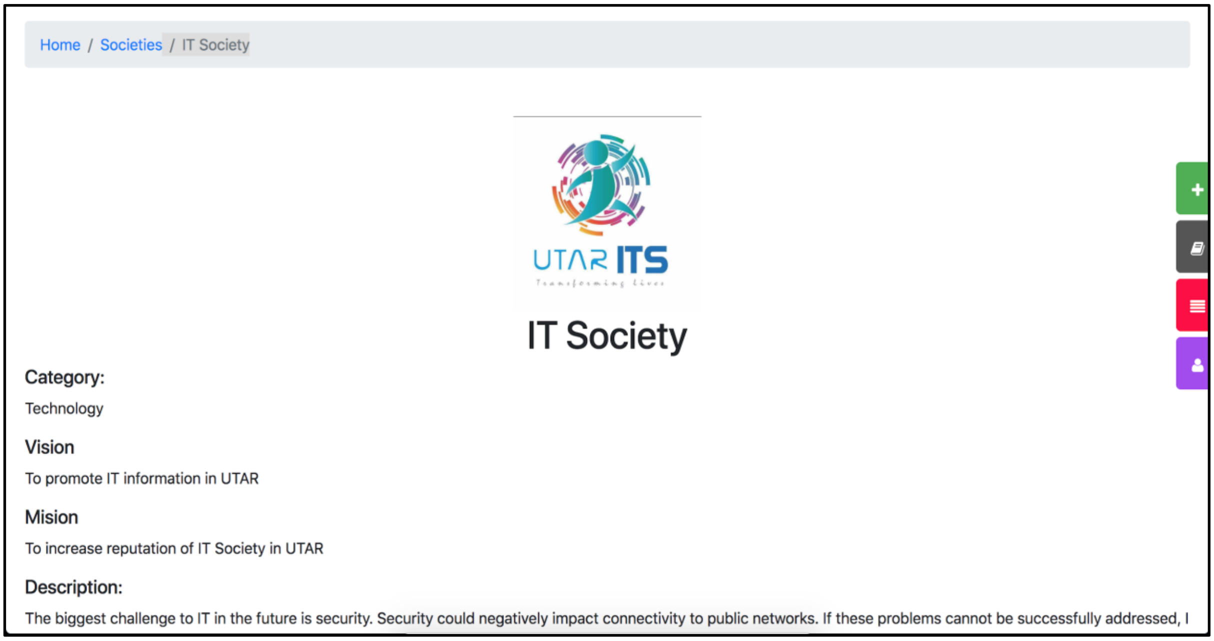
No.	Logo	Society	Events	Action
1		IT Society	WorkShop KLESF Easy Parcel Talk	✎
2		First Aid Society	Cardio Night Run Blood Donation Adventure Camp	✎
3		Engineering Society	ES Camp Engineering Fiesta Annual General Meeting	
4		Sport Society	Night Run Colour Run Sport Carnival	✎

Figure 4.6: UI - Show all registered societies



No.	Logo	Events	Organisers	Date	Rating Status	Actions	
1		WorkShop	IT Society	01/12/2018	-	🗑	⊗
2		Cardio Night Run	First Aid Society	21/10/2018	-	🗑	-
3		ES Camp	Engineering Society	01/05/2018	Done	-	-
4		Sport Carnival	Sport Club	17/01/2018	Undone	-	-

Figure 4.7: UI - Show all registered events



IT Society

Category:

Technology

Vision

To promote IT information in UTAR

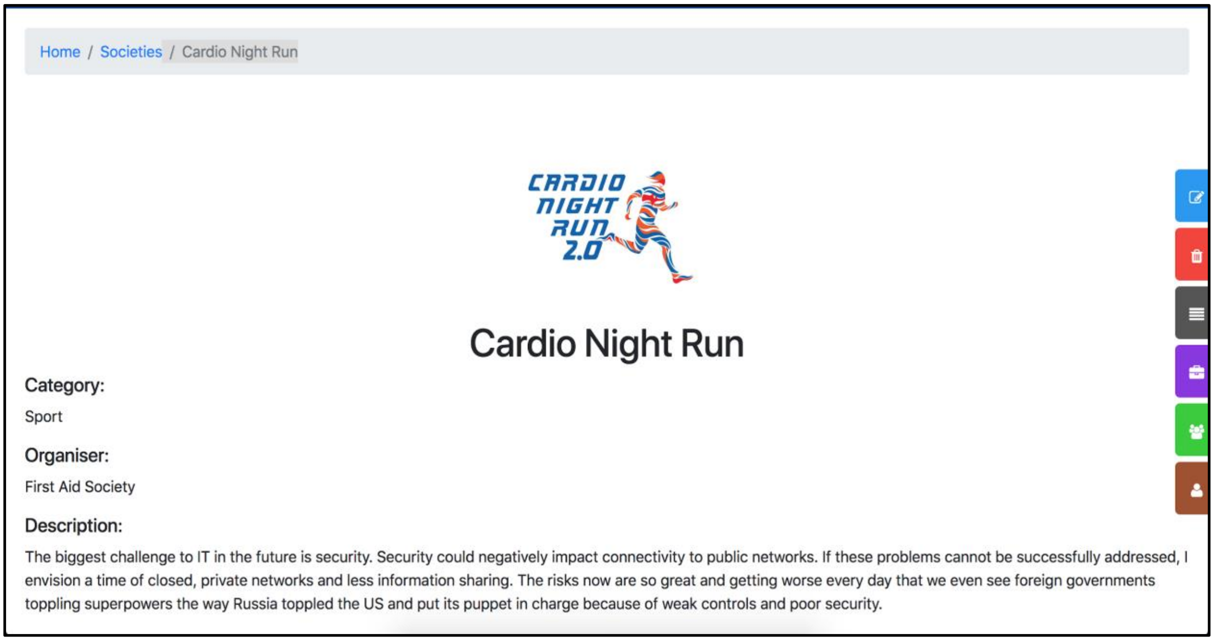
Mision

To increase reputation of IT Society in UTAR

Description:

The biggest challenge to IT in the future is security. Security could negatively impact connectivity to public networks. If these problems cannot be successfully addressed, I

Figure 4.8: UI - Show detailed information of each society



Home / Societies / Cardio Night Run

Cardio Night Run

Category:

Sport

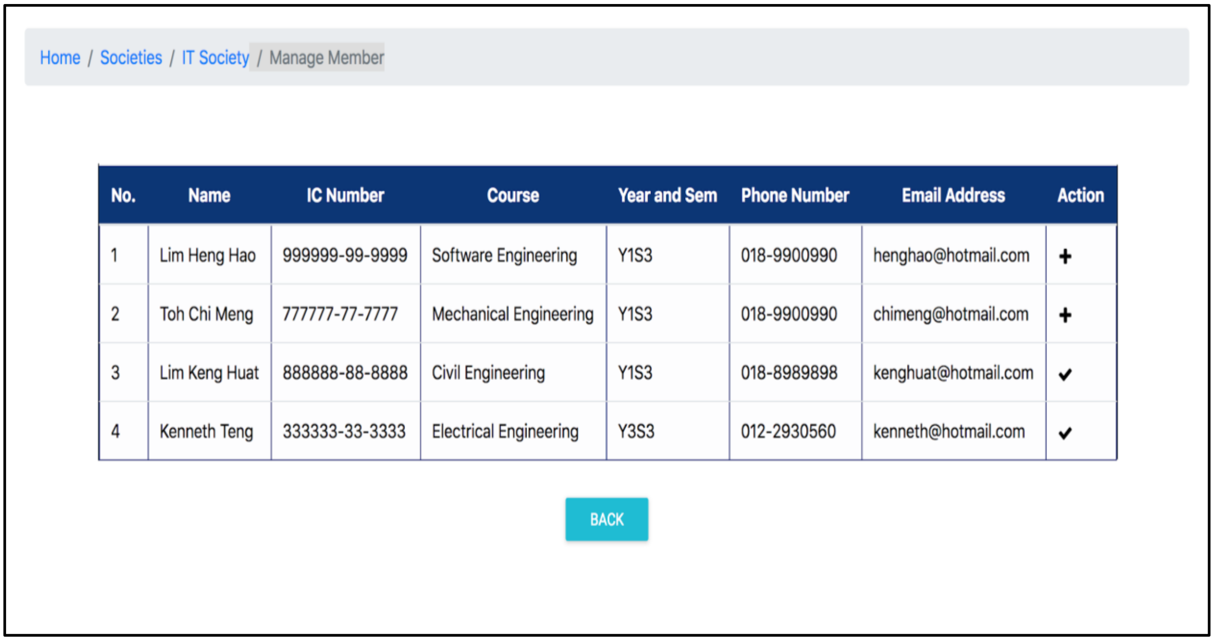
Organiser:

First Aid Society

Description:

The biggest challenge to IT in the future is security. Security could negatively impact connectivity to public networks. If these problems cannot be successfully addressed, I envision a time of closed, private networks and less information sharing. The risks now are so great and getting worse every day that we even see foreign governments toppling superpowers the way Russia toppled the US and put its puppet in charge because of weak controls and poor security.

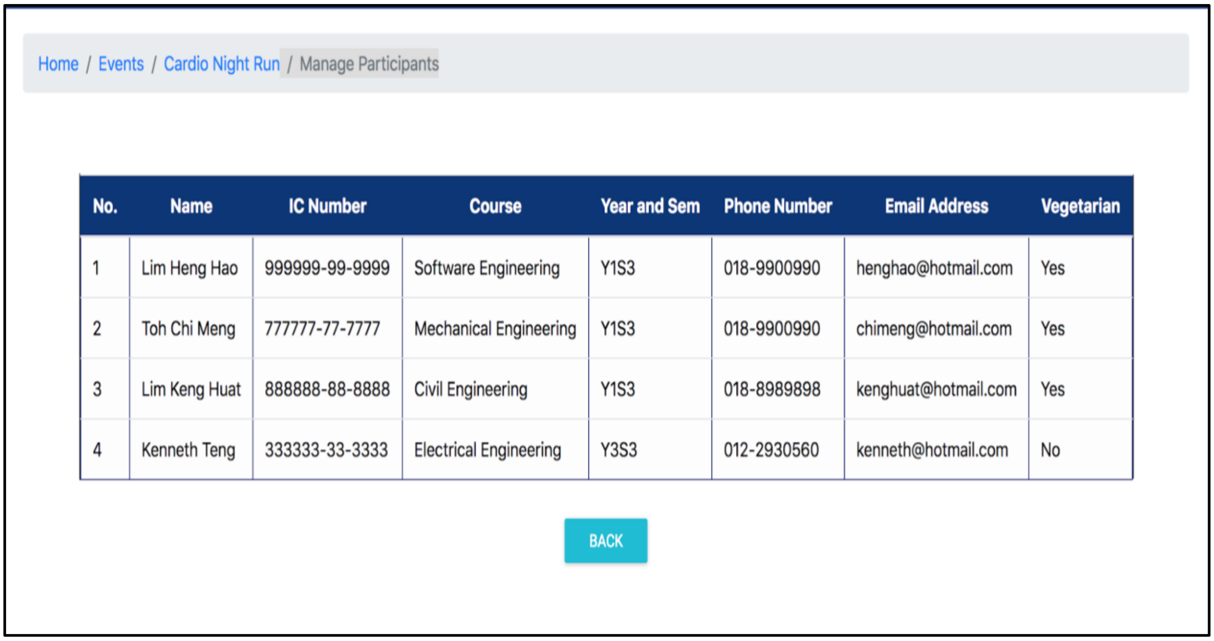
Figure 4.9: UI - Show detailed information of each event



Home / Societies / IT Society / Manage Member

No.	Name	IC Number	Course	Year and Sem	Phone Number	Email Address	Action
1	Lim Heng Hao	999999-99-9999	Software Engineering	Y1S3	018-9900990	henghao@hotmail.com	+
2	Toh Chi Meng	777777-77-7777	Mechanical Engineering	Y1S3	018-9900990	chimeng@hotmail.com	+
3	Lim Keng Huat	888888-88-8888	Civil Engineering	Y1S3	018-8989898	kenghuat@hotmail.com	✓
4	Kenneth Teng	333333-33-3333	Electrical Engineering	Y3S3	012-2930560	kenneth@hotmail.com	✓

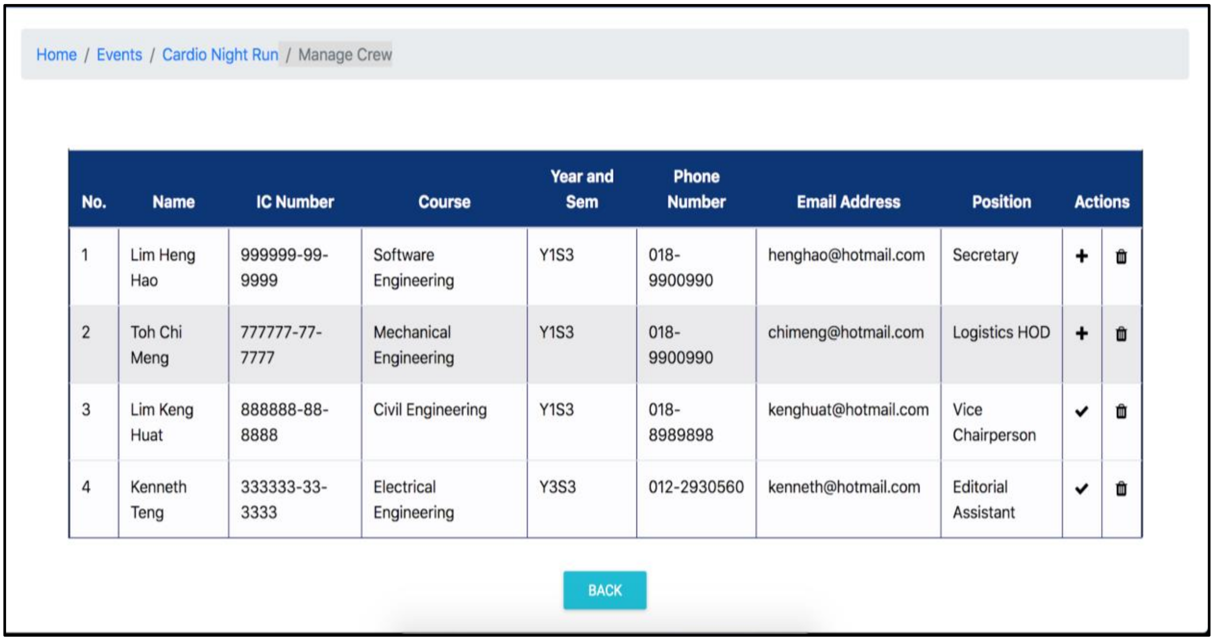
Figure 4.10: UI - Manage member registration of society



Home / Events / Cardio Night Run / Manage Participants

No.	Name	IC Number	Course	Year and Sem	Phone Number	Email Address	Vegetarian
1	Lim Heng Hao	999999-99-9999	Software Engineering	Y1S3	018-9900990	henghao@hotmail.com	Yes
2	Toh Chi Meng	777777-77-7777	Mechanical Engineering	Y1S3	018-9900990	chimeng@hotmail.com	Yes
3	Lim Keng Huat	888888-88-8888	Civil Engineering	Y1S3	018-8989898	kenghuat@hotmail.com	Yes
4	Kenneth Teng	333333-33-3333	Electrical Engineering	Y3S3	012-2930560	kenneth@hotmail.com	No

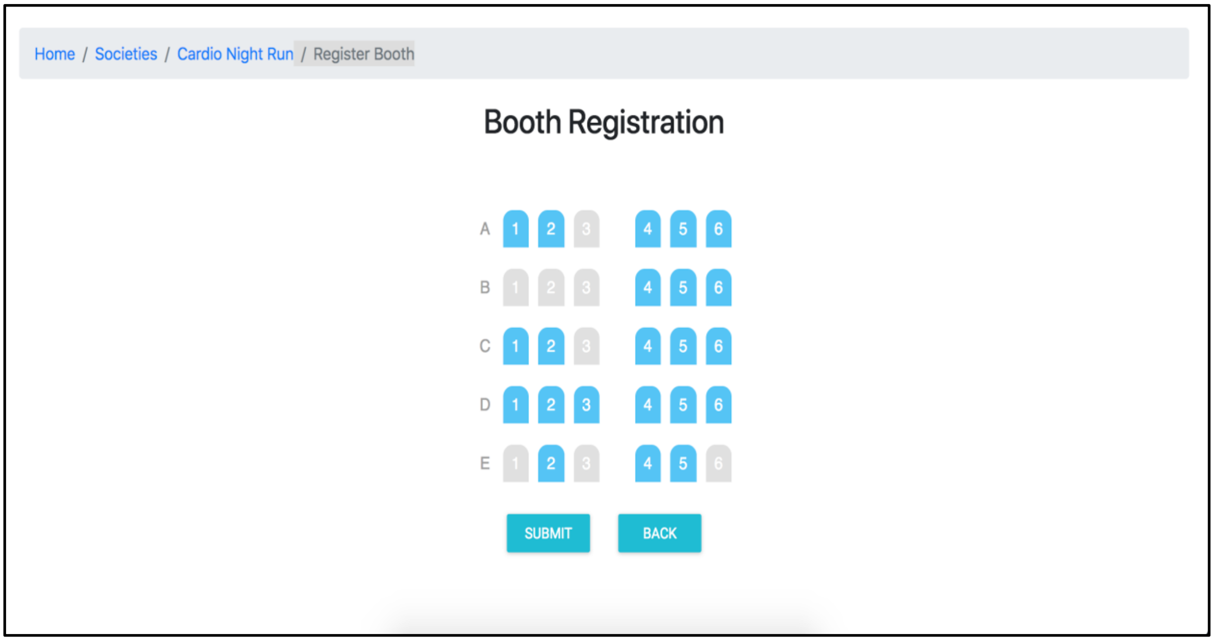
Figure 4.11: UI - Show list of registered participants for each event



Home / Events / Cardio Night Run / Manage Crew

No.	Name	IC Number	Course	Year and Sem	Phone Number	Email Address	Position	Actions	
1	Lim Heng Hao	999999-99-9999	Software Engineering	Y1S3	018-9900990	henghao@hotmail.com	Secretary	+	🗑
2	Toh Chi Meng	777777-77-7777	Mechanical Engineering	Y1S3	018-9900990	chimeng@hotmail.com	Logistics HOD	+	🗑
3	Lim Keng Huat	888888-88-8888	Civil Engineering	Y1S3	018-8989898	kenghuat@hotmail.com	Vice Chairperson	✓	🗑
4	Kenneth Teng	333333-33-3333	Electrical Engineering	Y3S3	012-2930560	kenneth@hotmail.com	Editorial Assistant	✓	🗑

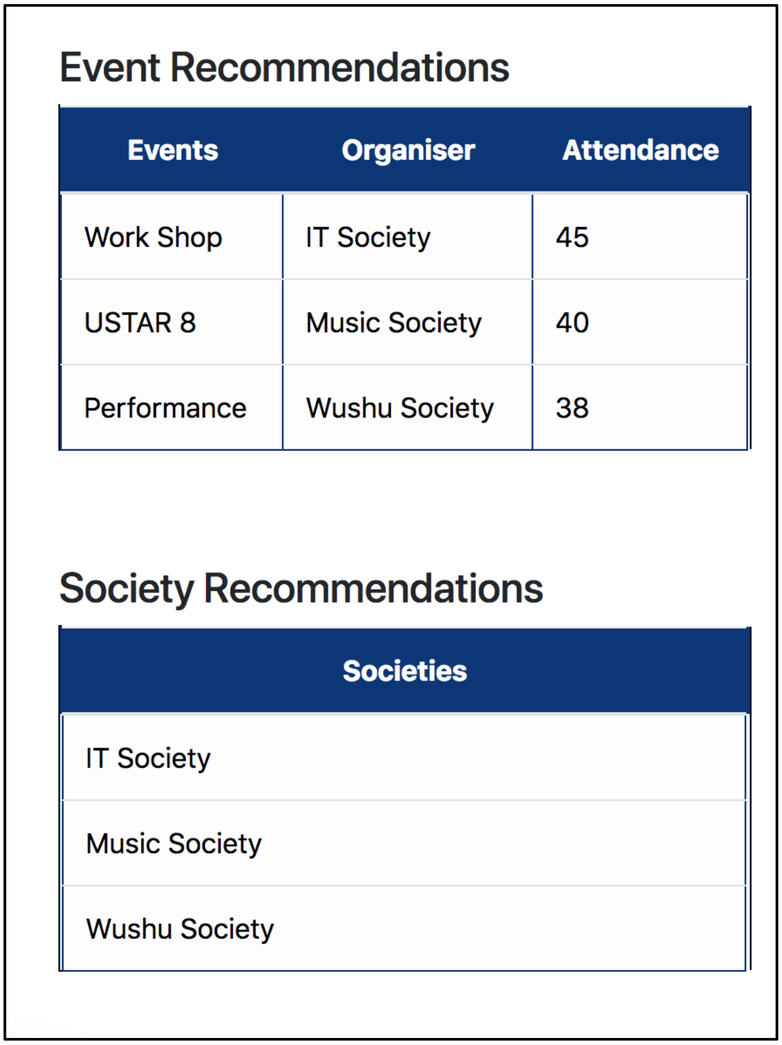
Figure 4.12: UI - Manage registered crew for each event



Home / Societies / Cardio Night Run / Register Booth

Booth Registration

Figure 4.13: UI - Register booth for society’s event



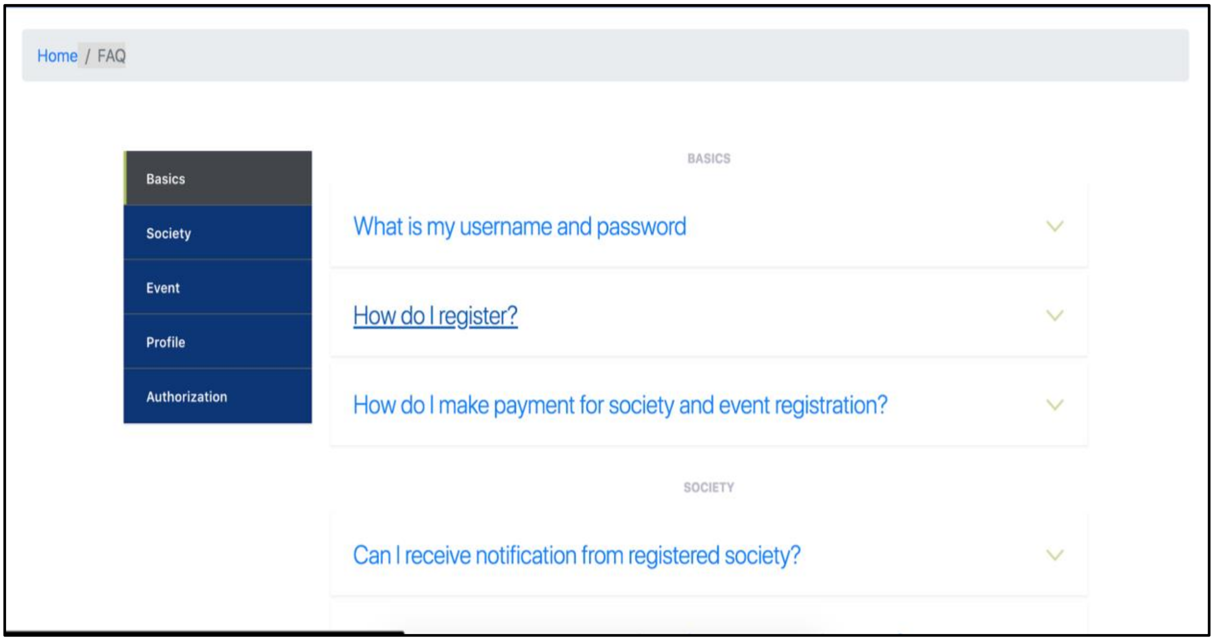
Event Recommendations

Events	Organiser	Attendance
Work Shop	IT Society	45
USTAR 8	Music Society	40
Performance	Wushu Society	38

Society Recommendations

Societies
IT Society
Music Society
Wushu Society

Figure 4.14: UI - Show recommendations of suitable societies and events



Home / FAQ

Basics

Society

Event

Profile

Authorization

BASICS

What is my username and password

How do I register?

How do I make payment for society and event registration?

SOCIETY

Can I receive notification from registered society?

Figure 4.15: UI - Show list of FAQ (categorized)

3.5 Project Plan

The project is planned based on the work breakdown structure and Gantt chart. Please refer to “Appendix A: Work breakdown structure and Gantt chart”.

CHAPTER 4

PROJECT SPECIFICATION

This chapter will introduce about the functional and non-functional requirements of the project in section 4.1. Section 4.2 will introduce about use case diagram while section 4.3 is about the use case description of this project. The last section which is section 4.4 will introduce about the preliminary user interface design.

4.1 Requirement Specification

4.1.1 Functional Requirements

The functional requirements are categorized according to the different roles of users who will be using this system. Note that also a society committee is also a student.

As a student

1. The user shall be able to login to their account.
2. The user shall be able to view list of past and upcoming co-curriculum events.
3. The user shall be able to view list of existing societies.
4. The user shall be able to view details of every event.
5. The user shall be able to view details of every society.
6. The user shall be able to search through the list of events.
7. The user shall be able to register for societies.
8. The user shall be able to register events.
9. The user shall be able to cancel the society registration.
10. The user shall be able to cancel the event registration.
11. The user shall be able to receive notifications about the registered events.
12. The user shall be able to obtain recommendations about the suitable societies and events to be participated.
13. The user shall be able to provide rating for every participated event.

As a society committee

1. The authorized user shall be able to manage society.

   1.1 The authorized user shall be able to modify society profile.

   1.2 The authorized user shall be able to manage society registration for crew and members.

   1.3 The authorized user shall be able to select authorized crew position for every event of the society.

2. The authorized user shall be able to manage events of respective societies.

   2.1 The authorized user shall be able to create new event.

   2.2 The authorized user shall be able to modify event details.

   2.3 The authorized user shall be able to delete the created event.

3. The authorized user shall be able to bid for their event's booth location.
4. The authorized user shall be able to export data of society members and event participants.

As a staff in-charged of respective societies

1. The authorized user shall be able to manage society.
2. The authorized user shall be able to manage events of respective societies.
3. The authorized user shall be able to export data of society members and event participants.

As a DSA officer

1. The authorized user shall be able to manage all societies.
2. The authorized user shall be able to manage events of respective societies.
3. The authorized user shall be able to manage the bidding system of event booth location.

   3.1 The authorized user shall be able to upload floor plan for booth location and determine amount of available booths.

4. The authorized user shall be able to export data of society members and event participants.

4.1.2 Non-functional requirements

Availability

The system shall be available for 24 hours per day and 7 days per week without any crashing when multiple users are accessing the system concurrently.

Security

1. The system data shall be backup from time to time every day in a secure database to prevent any data loss.
2. The system shall be able to display information according to the different roles of users logging in.

Usability

1. The system shall be user-friendly by providing a consistent user interface which is able to be used easily and efficiently.
2. The system shall be able to provide a user interface with high navigability. For example, user is able to complete one task through one or two steps instead of a lot of complicated steps.
3. The system shall be able to display error message when the user input is wrong.
4. The system shall be able to validate user input for required input to prevent empty data.

Performance

1. The system shall be able to be accessed through optimum browsing speed. For example, a page does not take a long period of time for loading.
2. The system shall be able to send notification to user on time.

Adaptability

1. The system shall be able to be modified easily when there is any new feature or new technology that will be implemented.

4.2 Use Case Modelling

In order to further describe the functionalities of this system, a use case modelling is done which describe detailed information of every functionality.

4.2.1 Use Case Diagram

Please refer to “Appendix B: Use Case Diagram”.

4.2.2 Use Case Description

4.2.2.1 Login

Use Case Name: Login	ID: 001	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to login to the pre-registered account. Staff - wants to login to the pre-registered account.		
Brief Description: This use case describes how student / staff login to the registered account.		
Trigger: Student / staff clicks on “Login” button on navigation bar.		
Relationships: Association: Student, Staff Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. User clicks on “Login” button on navigation bar. 2. System shows Login page. 3. User enters student ID and password and clicks on “Submit” button.		

4. System validates the user input and logs user into the registered account and redirects user to Home page.
Sub Flows: Not applicable
Alternate / Exceptional Flows: 3.1.1 User enters invalid student ID or password and clicks on “Submit” button. 3.1.2 System displays error message. 3.2.1 User enters student ID and password, then clicks on “Remember Me” checkbox and “Login” button. 3.2.2 System stores the remember token into “User” table in database. 3.3.1 User leaves the input field empty. 3.3.2 System displays error message.

4.2.2.2 View Events

Use Case Name: View Events	ID: 002	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to view list of UTAR events organized by various societies. Staff- wants to view list of UTAR events organized by various societies.		
Brief Description: This use case describes how student / staff view the list of events and the detailed information of each event.		
Trigger: Student / staff clicks on “Home” button on navigation bar.		
Relationships:		

Association: Student, Staff Include: - Extend: - Generalization: -
Normal Flow of Events: 1. User clicks on “Home” button on navigation bar. 2. System shows the page with a calendar view and all the events are recorded in the calendar. 3. User clicks on any single event. 4. System shows the page with the detailed information of that particular event.
Sub Flows: Not applicable
Alternate / Exceptional Flows: -

4.2.2.3 View Societies

Use Case Name: View Societies	ID: 003	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to view list of societies which are existing and active. Staff- wants to view list of societies which are existing and active.		
Brief Description: This use case describes how student / staff view the list of societies and the detailed information of each society.		
Trigger: Student / staff clicks on “Societies” button on navigation bar.		

Relationships: Association: Student, Staff Include: - Extend: - Generalization: -
Normal Flow of Events: 1. User clicks on “Societies” button on navigation bar. 2. System shows the page with a list of societies. 3. User clicks on any single society. 4. System shows the page with the detailed information of that particular society.
Sub Flows: Not applicable
Alternate / Exceptional Flows: -

4.2.2.4 Register Event

Use Case Name: Register Event	ID: 004	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to register for the interested events.		
Brief Description: This use case describes how student registers for the interested events.		
Trigger: Student clicks on “Join Participant” button.		
Relationships:		

Association: Student

Include: -

Extend: -

Generalization: -

Normal Flow of Events:

1. User clicks on “Join Participant” button.
2. System redirects user to Event Registration page based on the selected event.
3. User enters personal details and clicks on “Submit” button.
4. System checks and validates all input data.
5. System stores the user input into database.
6. System shows a success message which indicates that the event has been registered successfully.
7. System redirects user back to “My Profile” page.

Sub Flows:
Not applicable

Alternate / Exceptional Flows:

1.1.1 User clicks on “Join Crew” button.

3.1.1 User enters invalid personal details and clicks on “Submit” button.
    3.1.1.1 User enters invalid email address.
    3.1.1.2 User enters invalid phone number.
    3.1.1.3 User enters invalid student ID.
    3.1.1.4 User leaves the input field empty.
3.1.2 System displays error message.

3.2.1 User clicks on “Back” button.
3.2.2 System redirects user back to the previous page that user stayed.

4.2.2.5 Register Society

<table>
<tr><td>Use Case Name: Register Society</td><td>ID: 005</td><td>Importance Level: High</td></tr>
<tr><td>Primary Actor: Student</td><td colspan="2">Use Case Type: Detailed, Essential</td></tr>
<tr><td colspan="3">Stakeholders and Interests:
Student - wants to register for the interested societies.</td></tr>
<tr><td colspan="3">Brief Description:
This use case describes how student registers for the interested societies.</td></tr>
<tr><td colspan="3">Trigger: Student clicks on “Join” button.</td></tr>
<tr><td colspan="3">Relationships:
Association: Student
Include: -
Extend: -
Generalization: -</td></tr>
<tr><td colspan="3">Normal Flow of Events:
1. User clicks on “Join” button.
2. System redirects user to Society Registration page based on the selected society.
3. User enters personal details and clicks on “Submit” button.
4. System checks and validates all input data.
5. System stores the user input into database.
6. System shows a success message which indicates that the society has been registered successfully.
7. System redirects user back to “My Profile” page.</td></tr>
<tr><td colspan="3">Sub Flows:
Not applicable</td></tr>
<tr><td colspan="3">Alternate / Exceptional Flows:
3.1.1 User enters invalid personal details and clicks on “Submit” button.
3.1.1.1 User enters invalid email address.
3.1.1.2 User enters invalid phone number.</td></tr>
</table>

3.1.1.3 User enters invalid student ID. 3.1.1.4 User leaves the input field empty. 3.1.2 System displays error message. 3.2.1 User clicks on “Back” button. 3.2.2 System redirects user back to the previous page that user stayed.

4.2.2.6 Cancel Event Registration

Use Case Name: Cancel Event Registration	ID: 006	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to cancel the event registration which had been done previously.		
Brief Description: This use case describes how student cancels the event registration.		
Trigger: Student clicks on a cross (“X”) symbol.		
Relationships: Association: Student Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. User clicks on a cross symbol in the column of “Action” in the table of their personal profile. 2. System displays a dialogue box to confirm the cancellation of event registration. 3. User clicks on “Yes” button of the dialogue box. 4. System deletes the information of user from the related tables in database. 5. System closes the dialogue box. 6. System shows a success message which indicates that the event registration has been deleted successfully. 7. System retains user in the same page.		

Sub Flows: Not applicable.
Alternate / Exceptional Flows: 3.1.1 User clicks on “No” button of the dialogues box. 3.1.2 System closes the dialogue box and retains user in the same page.

4.2.2.7 Provide Rating

Use Case Name: Provide Rating	ID: 009	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to provide rating for the participated events.		
Brief Description: This use case describes how student provides rating for the participated events.		
Trigger: Student clicks on “Undone” button in a table.		
Relationships: Association: Student Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. User clicks on “Undone” button in the column of “Rating” of a table in their personal profile. 2. System redirects user to Event Rating page based on the selected event. 3. User enters personal details and provides rating. 4. User clicks on “Submit” button. 5. System checks and validates all input data. 6. System stores the user input into database. 7. System shows a success message which indicates that the event rating has been recorded successfully. 8. System redirects user back to “My Profile” page.		

Sub Flows: Not applicable.
Alternate / Exceptional Flows: 3.1.1 User enters invalid personal details and clicks on “Submit” button. 3.1.1.1 User enters invalid email address. 3.1.1.2 User enters invalid phone number. 3.1.1.3 User enters invalid student ID. 3.1.1.4 User leaves the input field empty. 3.1.2 System displays error message. 4.1.1 User clicks on “Back” button. 4.1.2 System redirects user back to the previous page that user stayed.

4.2.2.8 Receive reminder notification

Use Case Name: Receive Reminder Notification	ID: 010	Importance Level: High
Primary Actor: Student	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Student - wants to receive reminder notification from the system.		
Brief Description: This use case describes how student receives reminder notification form the system.		
Trigger: Student toggles the “Web Notification” button.		
Relationships: Association: Student Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. Student toggles the “Web Notification” button in Event Registration page.		

2. System stores the response of toggle button (either true or false) into database. 3. System sends reminder notification regarding user’s registered events based on the required time.
Sub Flows: Not applicable.
Alternate / Exceptional Flows: -

4.2.2.9 Manage Society Profile

<table>
<tr><td>Use Case Name: Manage Society Profile</td><td>ID: 012</td><td>Importance Level: High</td></tr>
<tr><td>Primary Actor: Society Committee</td><td colspan="2">Use Case Type: Detailed, Essential</td></tr>
<tr><td colspan="3">Stakeholders and Interests:
Society committee - wants to manage society profile.</td></tr>
<tr><td colspan="3">Brief Description:
This use case describes how society committee manages society profile.</td></tr>
<tr><td colspan="3">Trigger: Society committee clicks on “Create Society Profile” button.</td></tr>
<tr><td colspan="3">Relationships:
Association: Society Committee
Include: -
Extend: -
Generalization: -</td></tr>
<tr><td colspan="3">Normal Flow of Events:
1. User clicks on “Create Society Profile” button.</td></tr>
</table>

2. System redirects user to Create Society Profile page.
3. User enters information and details of the respective society.
4. User clicks on “Submit” button.
5. System checks and validates all input data.
6. System stores the user input into database.
7. System shows a success message which indicates that the society profile has been created successfully.
8. System redirects user back to the created profile page.

Sub Flows:
Not applicable.

Alternate / Exceptional Flows:

1.1.1 User clicks on “Edit” button.
1.1.2 System redirects user to “Edit Society Profile” page of the selected society.
1.1.3 User edits the information and clicks “Submit” button.
1.1.4 System redirects user to “Society Profile” page and displays success message.

1.2.1 User clicks on “Delete” button.
1.2.2 System displays a dialogue box to confirm the deletion of society profile.
1.2.3 User clicks on “Yes” button of the dialogue box.
1.2.4 System deletes the information of society from the related tables in database.
1.2.5 System closes the dialogue box.
1.2.6 System shows a success message which indicates that the society profile has been deleted successfully.
1.2.7 System retains user in the same page.

1.2.3.1 User clicks on “No” button of the dialogue box.
1.2.3.2 System closes the dialogue box.
1.2.3.3 System retains user in the same page.

3.1.1 User enters invalid society details and clicks on “Submit” button.
   3.1.1.1 User leaves the input field empty.
3.1.2 System displays error message.

4.1.1 User clicks on “Back” button.
4.1.2 System redirects user back to the previous page that user stayed.

4.2.2.10 Manage Event

Use Case Name: Manage Events	ID: 013	Importance Level: High
Primary Actor: Society Committee	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Society committee - wants to manage society’s events.		
Brief Description: This use case describes how society committee manages society’s events.		
Trigger: Society committee clicks on “Create Event” button.		
Relationships: Association: Society Committee Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. User clicks on “Create Event” button. 2. System redirects user to Create Event page. 3. User enters information and details of the new event. 4. User clicks on “Submit” button. 5. System checks and validates all input data. 6. System stores the user input into database. 7. System shows a success message which indicates that the new event has been created successfully. 8. System redirects user back to the created event page.		
Sub Flows: Not applicable.		
Alternate / Exceptional Flows: 1.1.1 User clicks on “Edit” button. 1.1.2 System redirects user to “Edit Event” page of the selected event. 1.1.3 User edits the information and clicks “Submit” button.		

1.1.4 System redirects user to “Event” page and displays success message. 1.2.1 User clicks on “Delete” button. 1.2.2 System displays a dialogue box to confirm the deletion of event. 1.2.3 User clicks on “Yes” button of the dialogue box. 1.2.4 System deletes the information of event from the related tables in database. 1.2.5 System closes the dialogue box. 1.2.6 System shows a success message which indicates that the event has been deleted successfully. 1.2.7 System retains user in the same page. 1.3.1 User clicks on “View participants” button. 1.3.2 System displays list of participants of the selected event. 1.2.3.1 User clicks on “No” button of the dialogue box. 1.2.3.2 System closes the dialogue box. 1.2.3.3 System retains user in the same page. 3.1.1 User enters invalid event details and clicks on “Submit” button. 3.1.1.1 User leaves the input field empty. 3.1.2 System displays error message. 4.1.1 User clicks on “Back” button. 4.1.2 System redirects user back to the previous page that user stayed.

4.2.2.11 Manage Event Crew Registration

Use Case Name: Manage Event Crew Registration	ID: 014	Importance Level: High
Primary Actor: Society Committee	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Society Committee - wants to manage event crew registration.		
Brief Description: This use case describes how society committee manages event crew registration.		
Trigger: Society committee clicks on “Manage Crew” button.		
Relationships: Association: Society Committee Include: -		

Extend: -

Generalization: -

Normal Flow of Events:

1. User clicks on “Manage Crew” button.
2. System redirects user to “Event Crews” page.
3. User clicks on the “Approve” button of selected event crew which has not be approved.
4. System displays a dialogue box to confirm the approval of the event crew.
5. User clicks on “Yes” button of the dialogue box.
6. System stores the information of the crew to the related tables in database.
7. System closes the dialogue box.
8. System shows a success message which indicates that the event crew has been approved successfully.
9. System retains user in the same page.

Sub Flows:
Not applicable.

Alternate / Exceptional Flows:

3.1.1 User clicks on the “Delete” button of selected event crew.
3.1.2 System displays a dialogue box to confirm the deletion of the event crew.
3.1.3 User clicks on “Yes” button of the dialogue box.
3.1.4 System delete the information of the crew from the related tables in database.
3.1.5 System closes the dialogue box.
3.1.6 System shows a success message which indicates that the event crew has been removed successfully.
3.1.7 System retains user in the same page.

3.1.3.1 User clicks on “No” button of the dialogue box.
3.1.3.2 System closes the dialogue box.
3.1.3.3 System retains user in the same page.

4.2.2.12 Bid Event Booth

Use Case Name: Bid Event Booth	ID: 015	Importance Level: High
Primary Actor: Society Committee	Use Case Type: Detailed, Essential	

Stakeholders and Interests:
Society committee - wants to bid for event booth’s location.

Brief Description:
This use case describes how society committee bids for event booth’s location.

Trigger: Society committee clicks on “Register Booth” button.

Relationships:

Association: Society Committee

Include: -

Extend: -

Generalization: -

Normal Flow of Events:

1. User clicks on “Register Booth” button in Society or Event page.
2. System redirects user to “Booth Registration” page.
3. User clicks on the intended location of booth.
4. System displays the selected location in green colour.
5. User clicks on “Submit” button.
6. System stores the user input into database.
7. System shows a success message which indicates that the location of booth has been registered successfully.
8. System redirects user back to the society or event page.

Sub Flows:
Not applicable.

Alternate / Exceptional Flows:

5.1.1 User clicks on “Back” button.
5.1.2 System redirects user back to the previous page that user stayed.

4.2.2.13 Manage Bidding System

<table>
<tr><td>Use Case Name: Manage Bidding System</td><td>ID: 017</td><td>Importance Level: High</td></tr>
<tr><td>Primary Actor: DSA</td><td colspan="2">Use Case Type: Detailed, Essential</td></tr>
<tr><td colspan="3">Stakeholders and Interests:
Student Representative Council / Staff - want to manage bidding system of event booth location.</td></tr>
<tr><td colspan="3">Brief Description:
This use case describes how DSA manage the bidding system of event booth location. .</td></tr>
<tr><td colspan="3">Trigger:</td></tr>
<tr><td colspan="3">Relationships:
Association: Student
Include: -
Extend: -
Generalization: -</td></tr>
<tr><td colspan="3">Normal Flow of Events:
1. User clicks on “Manage Booth” button in “Manage” drop down on navigation bar.
2. User enters information and details of the respective booth management.
3. User clicks on “Submit” button.
4. System checks and validates all input data.
5. System stores the user input into database.
6. System shows a success message which indicates that the booth management has been updated successfully.
7. System redirects user back to the home page.</td></tr>
<tr><td colspan="3">Sub Flows:
Not applicable.</td></tr>
<tr><td colspan="3">Alternate / Exceptional Flows: -</td></tr>
</table>

4.2.2.14 Manage Membership Registration

Use Case Name: Manage Membership Registration	ID: 018	Importance Level: High
Primary Actor: Staff	Use Case Type: Detailed, Essential	
Stakeholders and Interests: Staff - wants to manage society’s membership registration.		
Brief Description: This use case describes how staff manages society’s membership registration.		
Trigger: Staff clicks on “Manage Member” button.		
Relationships: Association: Staff Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. User clicks on “Manage Member” button. 2. System redirects user to “Society Members” page. 3. User clicks on the “Approve” button of selected new member which has not be approved. 4. System displays a dialogue box to confirm the approval of the new member. 5. User clicks on “Yes” button of the dialogue box. 6. System stores the information of the new member to the related tables in database. 7. System closes the dialogue box. 8. System shows a success message which indicates that the new member has been approved successfully. 9. System retains user in the same page.		
Sub Flows: Not applicable.		
Alternate / Exceptional Flows:		

3.1.1 User clicks on the “Delete” button of selected new member.
3.1.2 System displays a dialogue box to confirm the deletion of the new member.
3.1.3 User clicks on “Yes” button of the dialogue box.
3.1.4 System delete the information of the new member from the related tables in database.
3.1.5 System closes the dialogue box.
3.1.6 System shows a success message which indicates that the new member has been removed successfully.
3.1.7 System retains user in the same page.

3.1.3.1 User clicks on “No” button of the dialogue box.
3.1.3.2 System closes the dialogue box.
3.1.3.3 System retains user in the same page.

CHAPTER 5

DESIGN

This chapter describes the design constructed to develop the information system.

5.1 Software Architecture Design

The architecture design used is client-server architecture. ReactJS is used for the client side while Flask is used for the server side. Both of these frameworks are chosen to be used in this system is because of its simplicity and familiarity based on previous experience. The overview of client-server architecture is shown in Figure 5.1 below.

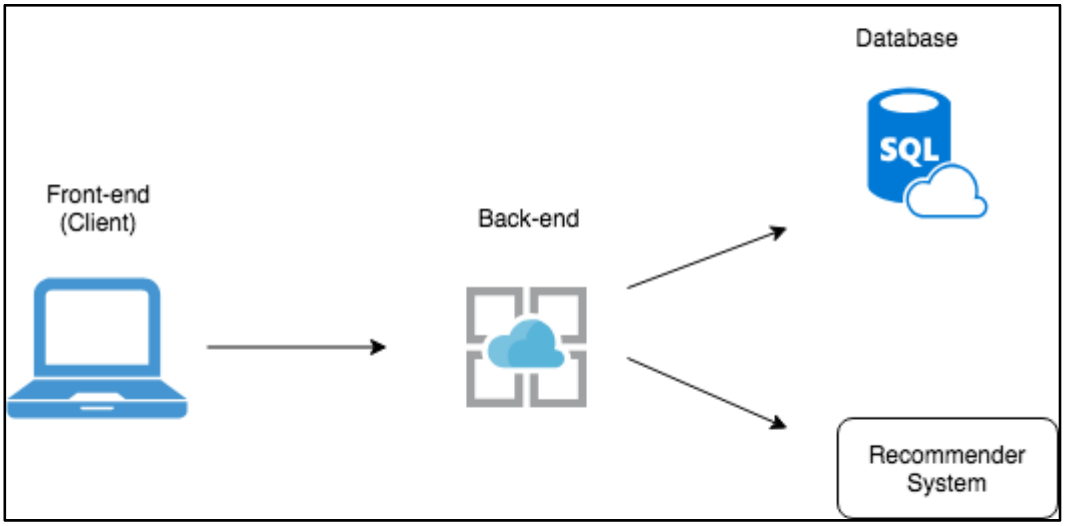


Figure 5.1: Client-Server Architecture of the system

5.2 Software Component Design

There are two parts in the software architecture of this system which are client and server sides. The overall system architecture is constructed to show the components of both client and server sides as shown in Figure 5.2 below.

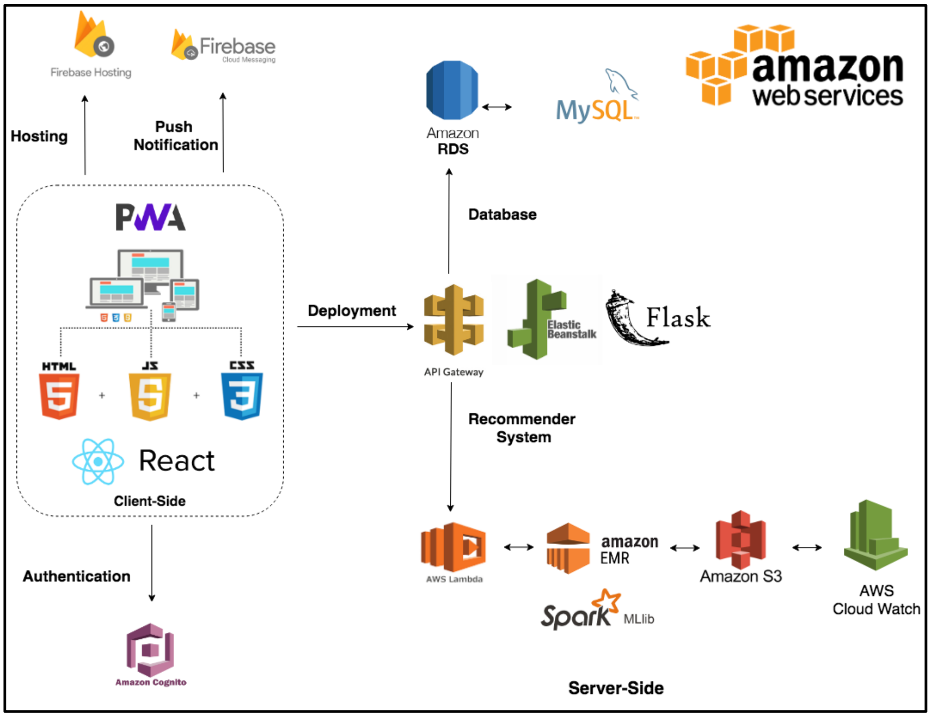


Figure 5.2: System Architecture Design

5.2.1 Client Components

1. Progressive Web Application (PWA)

   PWA are web applications that render websites and web pages but able to provide some user functionalities. It is reliable, fast and engaging. It is reliable as it loads instantly without long loading period even the network is not stable. It is fast because it responds quickly towards user interaction with smooth scrolling and animations. Lastly, it is engaging because it is available in home screen and able to receive push notifications (Google Developers, n.d.).

2. React Framework

   React is an open source JavaScript library which is used to develop front-end user interface for single page applications. Single page applications are web application which can complete data transaction without page reloading. React is a fast, simple and scalable framework and it only works for user interfaces which is the View in MVC template (C# Corner, n.d.).

5.2.2 Server Components

1. AWS Elastic Beanstalk

   AWS Elastic Beanstalk is a cloud service which is used to deploy and scale web applications. Code can be uploaded to cloud and Elastic Beanstalk will help to handle the deployment such as load balancing, capacity provisioning and auto-scaling. The web applications are deployed on familiar servers such as Nginx, Apache and Passenger (Amazon Web Services, n.d.).

2. Amazon Relational Database Service (Amazon RDS)

   Amazon RDS is a cloud service for setting up, operating and scaling a relational database in cloud. There are few options for database engines such as PostgreSQL, Amazon Aurora, MySQL, MariaDB, SQL Server and Oracle Database (Amazon Web Services, n.d.).

3. Amazon Elastic MapReduce (Amazon EMR)

   Amazon EMR processes large amount of data easily, fast, and cost-effective with the existence of a managed Hadoop framework. Other distributed frameworks can be run on it as well such as Apache Spark, Presto, HBase and Flink. It is also able to interact with other data stores such as Amazon DynamoDB and Amazon S3. In addition, EMR is able to handle a lot of big data use cases such as ETL, log analysis, financial analysis and machine learning (Amazon Web Services, n.d.).

4. Amazon Simple Storage Service (Amazon S3)

   Amazon S3 is a storage service which can store and protect various amount of data for mobile applications, websites, IoT devices, backup and restore. It provides management services which is easy to be used in order to self organize the data and customize the access control to meet business or organizational requirements (Amazon Web Services, n.d.).

5. Amazon Cognito

   Amazon Cognito makes user authentication process easier and access control to both web and mobile applications in a more quick and easy manner. Amazon Cognito is able to be scaled to millions of users and it enables authentication with social providers such as Google and Facebook. (Amazon Web Services, n.d.).

6. Amazon Lambda

   Amazon Lambda allows code execution in a serverless environment without server management by the developers. Lambda can be triggered by other AWS services or called by other web and mobile application for execution.

7. Apache Spark MLlib

   MLlib is a machine learning library of Spark. It makes machine learning more easy and scalable (Apache Spark, n.d.).

8. Flask Web Framework

   Flask is a web framework written in Python which is also known as micro web framework. It does not require additional libraries or tools and does not support form validation or database abstraction layer (Real Python, n.d.).

9. Firebase Hosting

   Firebase hosting provides secure and fast hosting for websites and web applications. It also supports hosting for single page web applications (Firebase, n.d.).

10. Firebase Cloud Messaging (FCM)

    FCM is a used to deliver messages without any cost as a messaging solution for cross-platform application. It sends notification message to client app to notify user about important information (Firebase, n.d.).

5.3 Detailed Architecture Design

5.3.1 Recommender System Architecture

The recommender system is being developed using few different types of cloud services in AWS. Lambda is used to run code in (Python) without managing the computer resources such as servers. Lambda is being triggered by other cloud services in AWS to start the execution. In addition, SQL database is being hosted using RDS to retrieve student's rating information and to store prediction output. MLlib is being used for the recommendation and the data analysis is being executed using EMR. Scheduler is used to schedule the whole process to be executed in a weekly manner.

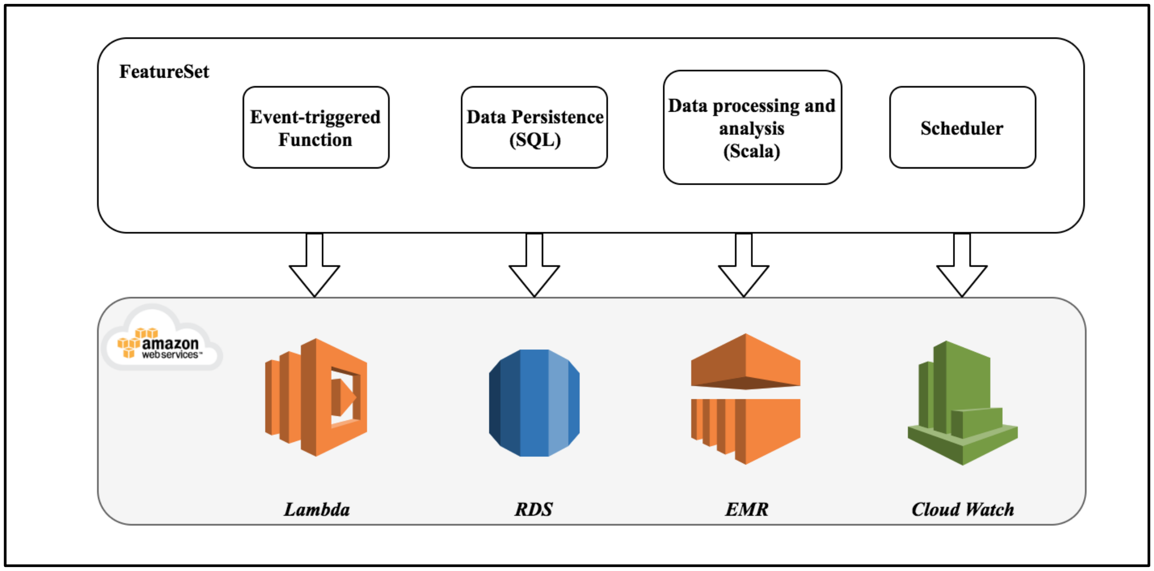


Figure 5.3: Recommender System Architecture

5.3.2 Microservices Architecture

Microservices architecture design is used in developing this system where a monolithic application is being decomposed into smaller separated applications or services. Those separated applications and services are being deployed independently in the same or different cloud services as others. For instance, the micro services which are being developed in this system are authentication, recommender system and the backend API design. The purpose of using this microservice architecture is to increase the high flexibility, high reliability and high scalability.

1. High flexibility - The application can be developed using different kinds of technologies for each of the isolated service.
2. High reliability - The malfunction of one of the service will not affect the entire application or other isolated services.
3. High scalability - Rebuilt of the entire system is not necessary when the application need to be updated from time to time.

5.4 Database Design

5.4.1 Entity Relational Diagram (ERD)

ERD shows the relationship between every entity which are stored in database. ERD is a diagram which has not been normalised yet. In this application, there are a lot of many-to-many relationship. This relationship produces many pivot tables to store they key data of both entities. There are three separated and standalone tables where they are used to store general information for booth location and the recommendation and analysis data produced by the recommender system.

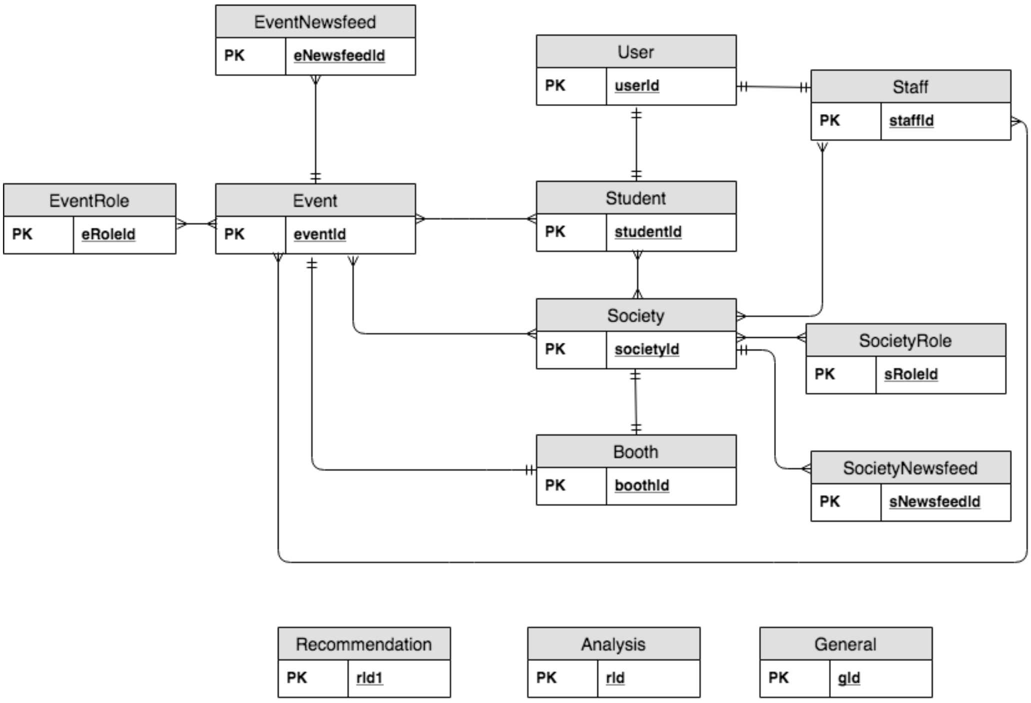


Figure 5.4: Entity Relational Diagram

5.4.2 Logical Entity Relational Diagram (ERD)

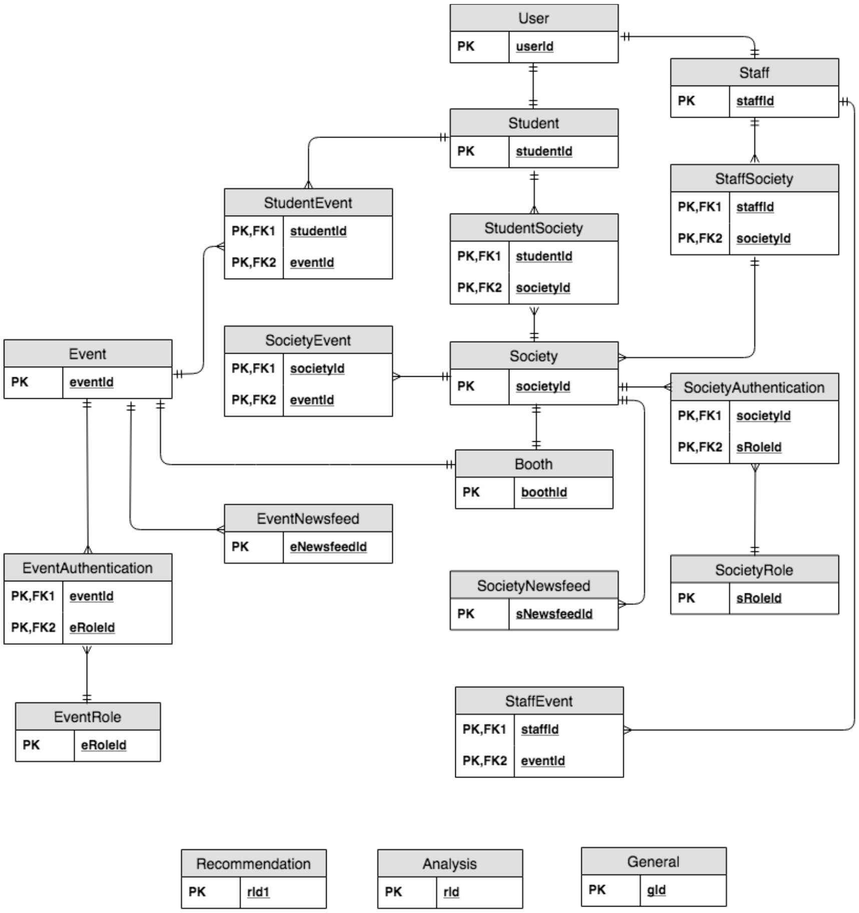


Figure 5.5: Logical Entity Relational Diagram

5.4.3 Data Dictionary

Table name: User

Column	**Description**	**Data Type**	**Primary Key**	**Foreigner Key**	**FK referenced table**
userId	Unique ID for all users	Int	Yes	No	-
username	User's name	Varchar	No	No	-
fcmToken	User's token for Firebase Cloud Messaging	Varchar	No	No	-

Table name: Student

Column	Description	Data Type	Primary Key	Foreigner Key	FK referenced table
studentId	Unique identification for all students	Int	Yes	No	-
studentName	Student's name	Varchar	No	No	-
officialId	Student's ID in University	Varchar	No	No	-
ic	Student's IC	Varchar	No	No	-
email	Student's Email	Varchar	No	No	-
contact	Student's contact number	Varchar	No	No	-
course	Student's course	Varchar	No	No	-
year	Student's academic year	Int	No	No	-
semester	Student's academic semester	Int	No	No	-
userId	Unique identification for all users	Int	No	Yes	User

Table name: Staff

Column	Description	Data Type	Primary Key	Foreigner Key	FK referenced table
staffId	Unique identification for all staffs	Int	Yes	No	-
staffName	Staff's name	Varchar	No	No	-
officialId	Staff's ID in University	Varchar	No	No	-
ic	Staff's IC	Varchar	No	No	-
email	Staff's Email	Varchar	No	No	-

contact	Staff’s contact number	Varchar	No	No	-
userId	Unique identification for all users	Int	No	Yes	User

Table name: Booth

Column	Description	Data Type	Primary Key	Foreigner Key	FK referenced table
boothId	Unique ID for all booths	Int	Yes	No	-
row	Booth row	Int	No	No	-
seat	Booth seat	Int	No	No	-
location	Booth location	Varchar	No	No	-

Table name: Society

Column	Description	Data Type	Primary Key	Foreigner Key	FK referenced table
societyId	Unique ID for all societies	Int	Yes	No	-
name	Society name	Varchar	No	No	-
description	Society description	Varchar	No	No	-
category	Society category	Varchar	No	No	-
vision	Society vision	Varchar	No	No	-
mission	Society mission	Varchar	No	No	-
logoUrl	Society logo URL	Varchar	No	No	-
boothId	Unique identification for all booths	Int	No	Yes	Booth

Table name: Event

Column	Description	Data Type	Primary Key	Foreigner Key

eventId	Unique ID for all events	Int	Yes	No
name	Event name	Varchar	No	No
category	Event category	Varchar	No	No
startDate	Event start date	Date Time	No	No
endDate	Event end date	Date Time	No	No
organiserId	Event organiser ID	Int	No	No
organiserName	Event organiser name	Varchar	No	No
description	Event description	Varchar	No	No
venue	Event venue	Varchar	No	No
fee	Event fee	Int	No	No
ssCategory	Event soft skill category	Int	No	No
ssPoint	Event soft skill point	Int	No	No
chairperson	Event chairperson	Varchar	No	No
contact	Event chairperson contact	Varchar	No	No
logoUrl	Event logo URL	Varchar	No	No
totalParticipant	Event total participants	Int	No	No
totalCrew	Event total crews	Int	No	No
crewPosition	Event available crew positions	Varchar	No	No
boothId	Unique identification for all booths	Int	No	Yes

Table name: SocietyRole

Column	**Description**	**Data Type**	**Primary Key**	**Foreigner Key**	**FK referenced table**
sRoleId	Unique ID for all society roles	Int	Yes	No	-
roleName	Society role name	Varchar	No	No	-

Table name: EventRole

Column	**Description**	**Data Type**	**Primary Key**	**Foreigner Key**	**FK referenced table**
eRoleId	Unique ID for all event roles	Int	Yes	No	-
roleName	Event role name	Varchar	No	No	-

Table name: SocietyNewsfeed

Column	**Description**	**Data Type**	**Primary Key**	**Foreigner Key**	**FK referenced table**
sNewsfeedId	Unique ID for all society news feed	Int	Yes	No	-
societyId	Unique identification for all societies	Int	No	Yes	Society
name	News feed name	Varchar	No	No	-
category	News feed category	Varchar	No	No	-
description	News feed description	Varchar	No	No	-
dateCreate	Creation date of news feed	Date Time	No	No	-
type	News feed type	Varchar	No	No	-

Table name: EventNewsfeed

Column	**Description**	**Data Type**	**Primary Key**	**Foreigner Key**	**FK referenced table**
eNewsfeedId	Unique ID for all event news feed	Int	Yes	No	-
eventId	Unique identification for all events	Int	No	Yes	Event
name	News feed name	Varchar	No	No	-
category	News feed category	Varchar	No	No	-
description	News feed description	Varchar	No	No	-
dateCreate	Creation date of news feed	Date Time	No	No	-
type	News feed type	Varchar	No	No	-

5.5 User Interface Design

5.5.1 Viewing

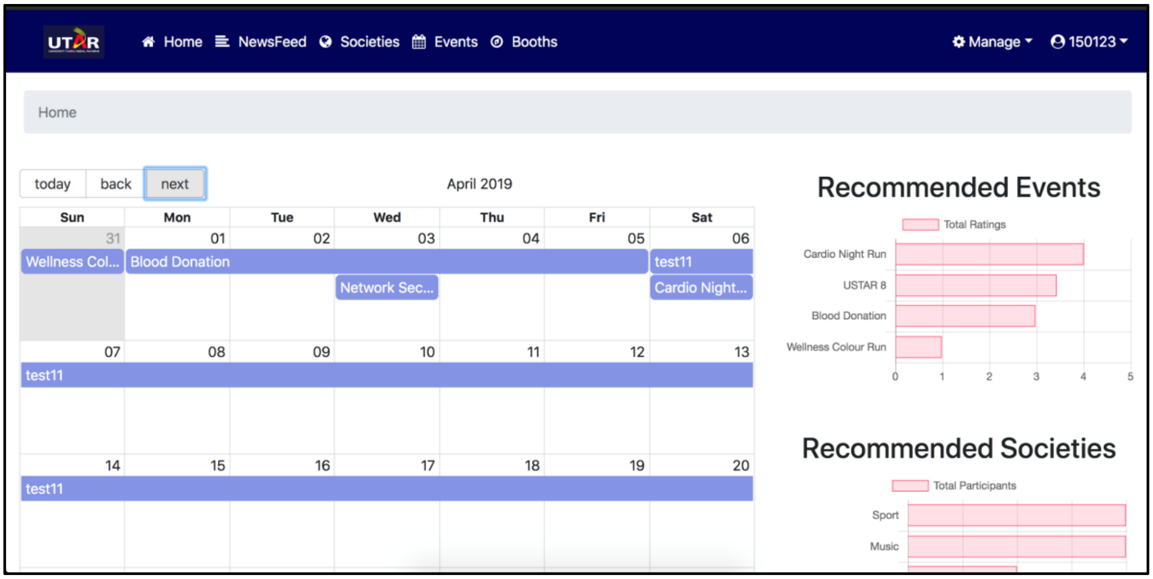


Figure 5.6: Home page with events in calendar view and recommendation in chart view

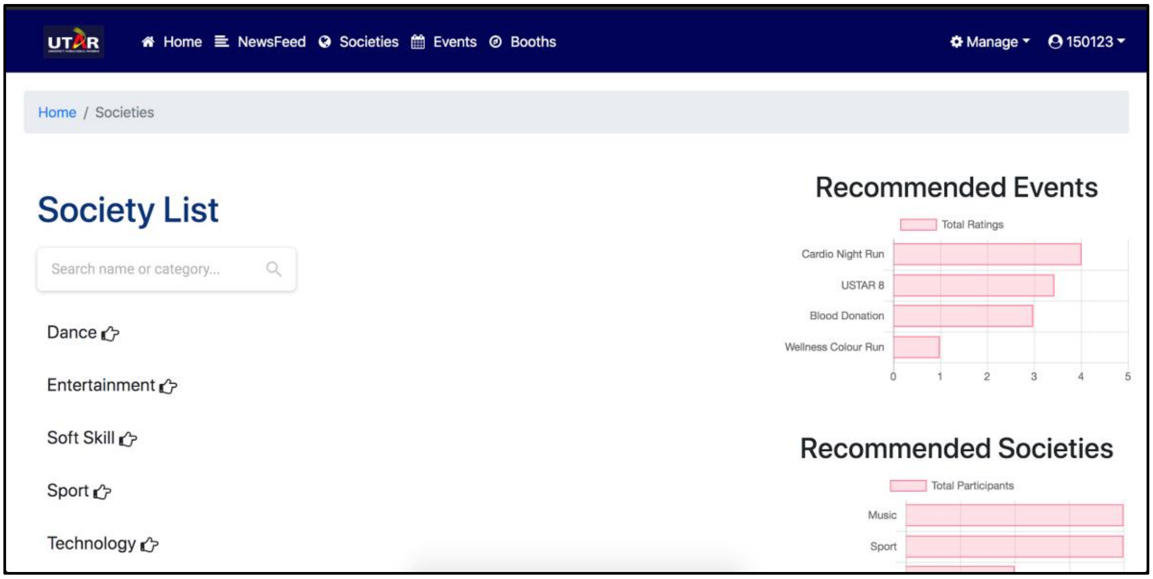
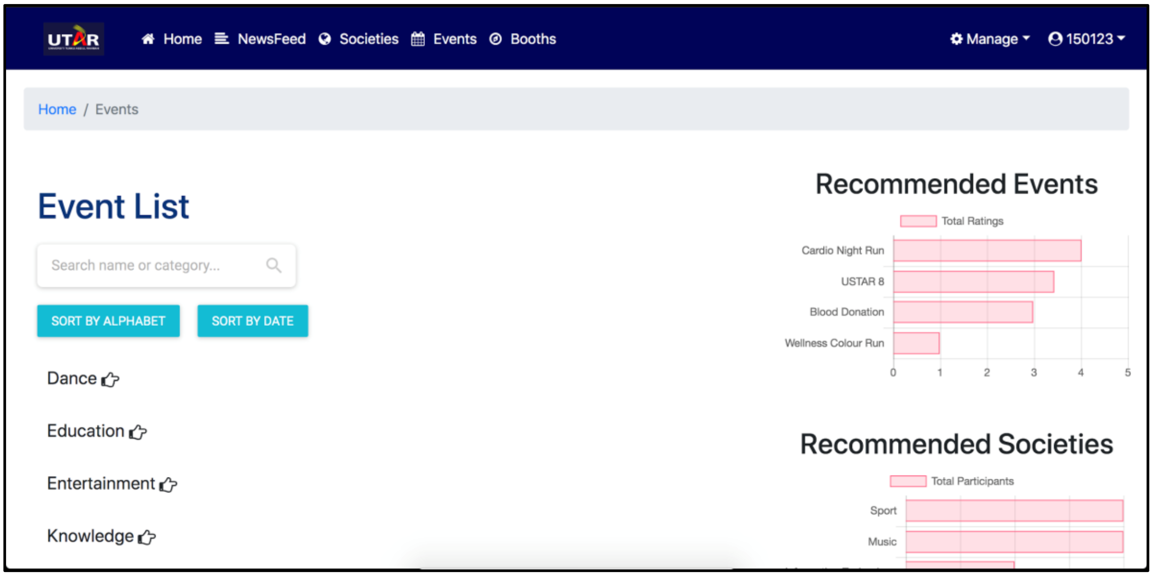


Figure 5.7: List of categorized societies



Home / Events

Event List

Dance

Education

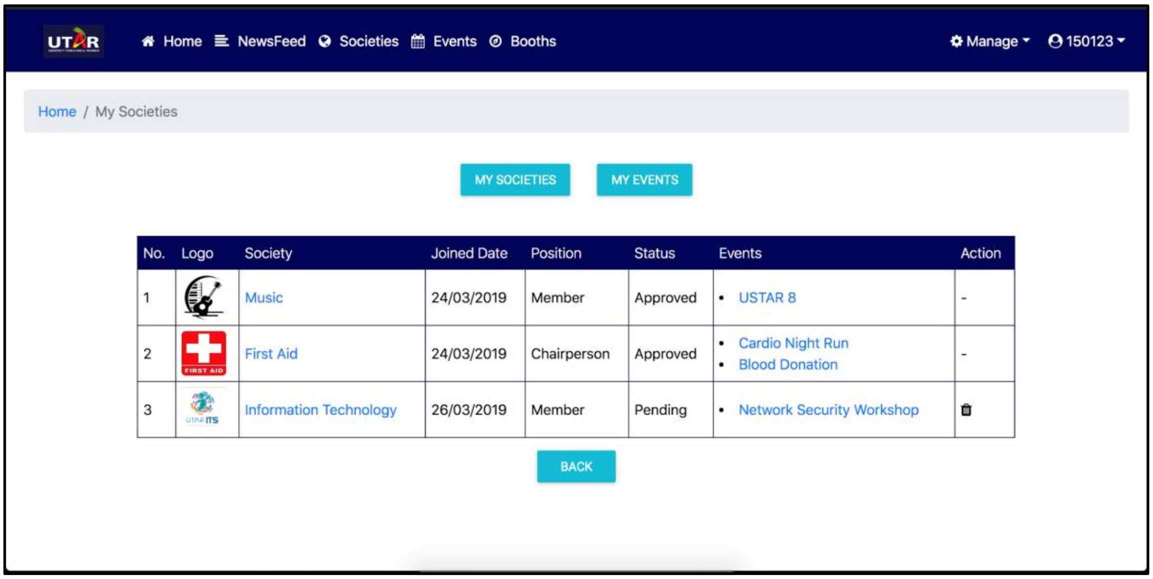
Entertainment

Knowledge

Recommended Events

Recommended Societies

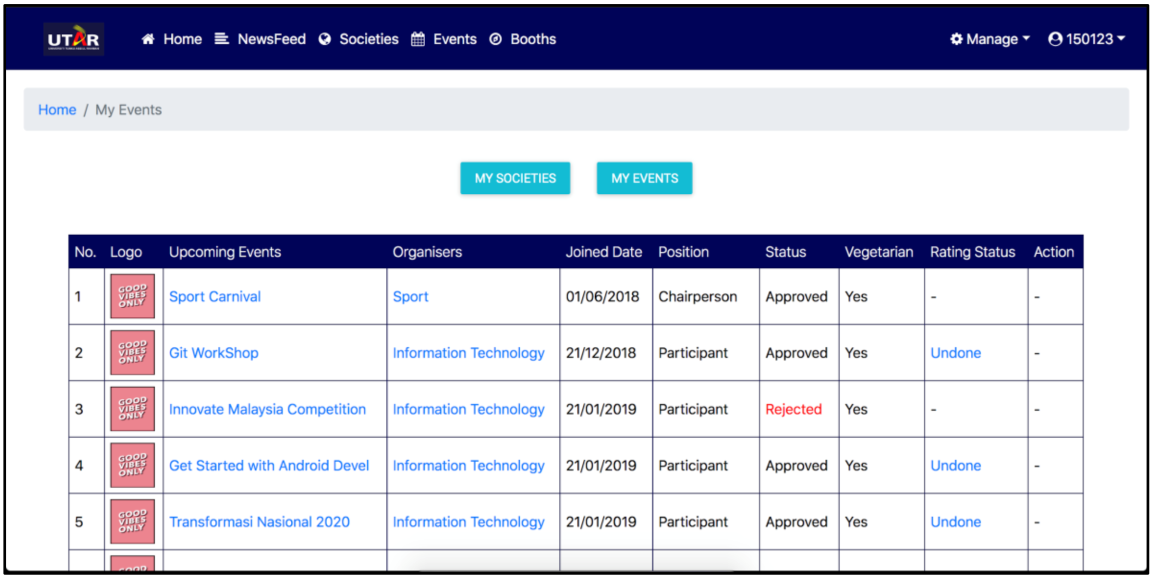
Figure 5.8: List of categorized events which can be sorted in alphabetical or date



Home / My Societies

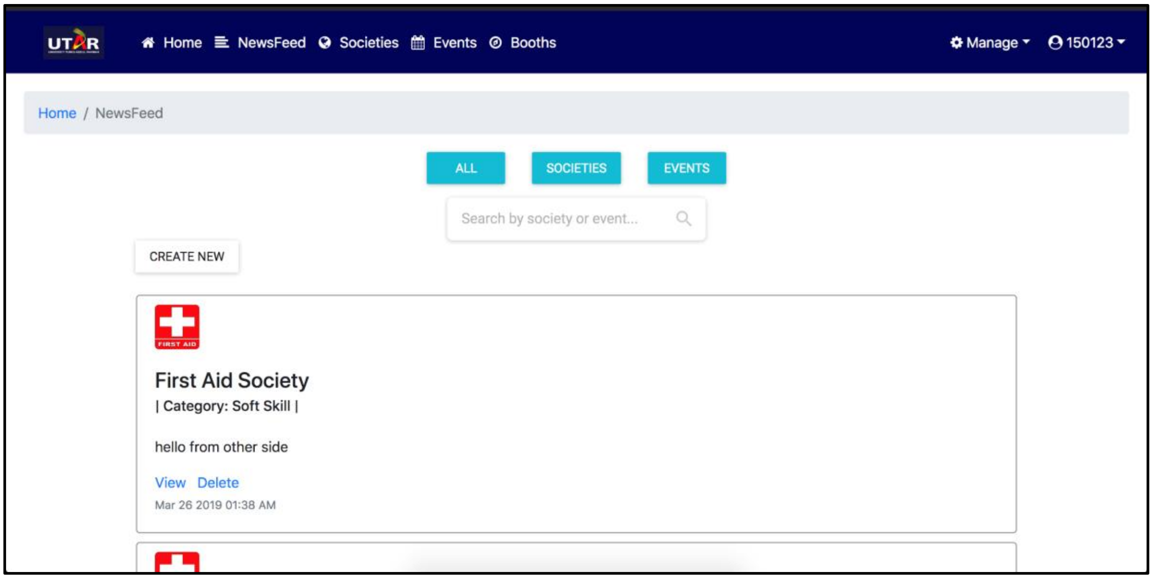
No.	Logo	Society	Joined Date	Position	Status	Events	Action
1		Music	24/03/2019	Member	Approved	• USTAR 8	-
2	FIRST AID	First Aid	24/03/2019	Chairperson	Approved	• Cardio Night Run • Blood Donation	-
3	UTAR ITS	Information Technology	26/03/2019	Member	Pending	• Network Security Workshop	🗑

Figure 5.9: List of participated societies



No.	Logo	Upcoming Events	Organisers	Joined Date	Position	Status	Vegetarian	Rating Status	Action
1	GOOD VIBES ONLY	Sport Carnival	Sport	01/06/2018	Chairperson	Approved	Yes	-	-
2	GOOD VIBES ONLY	Git WorkShop	Information Technology	21/12/2018	Participant	Approved	Yes	Undone	-
3	GOOD VIBES ONLY	Innovate Malaysia Competition	Information Technology	21/01/2019	Participant	Rejected	Yes	-	-
4	GOOD VIBES ONLY	Get Started with Android Devel	Information Technology	21/01/2019	Participant	Approved	Yes	Undone	-
5	GOOD VIBES ONLY	Transformasi Nasional 2020	Information Technology	21/01/2019	Participant	Approved	Yes	Undone	-

Figure 5.10: List of participated events



First Aid Society

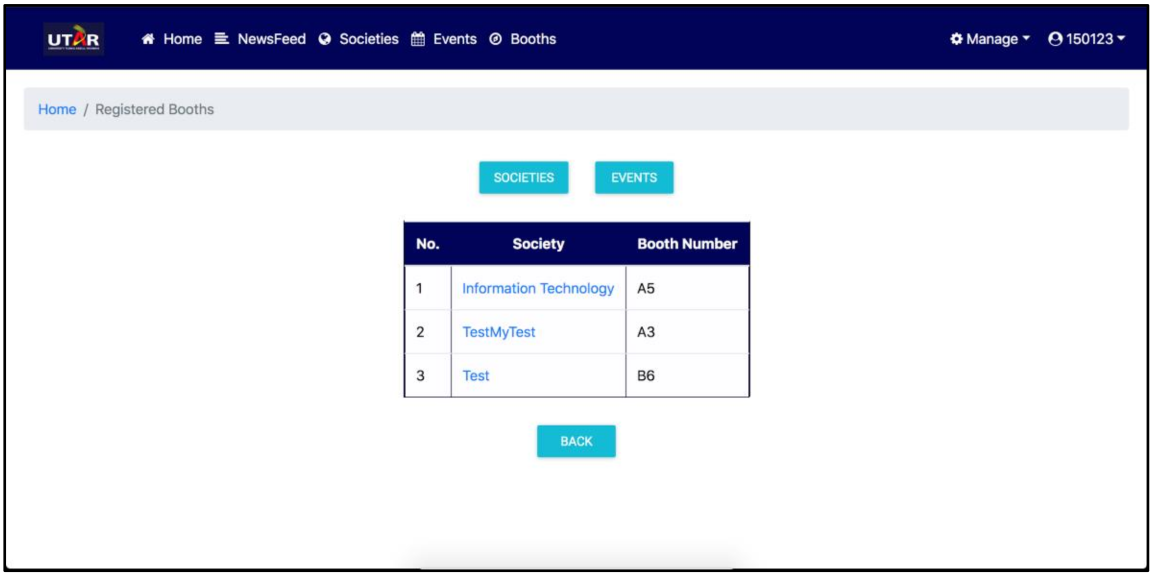
| Category: Soft Skill |

hello from other side

View Delete

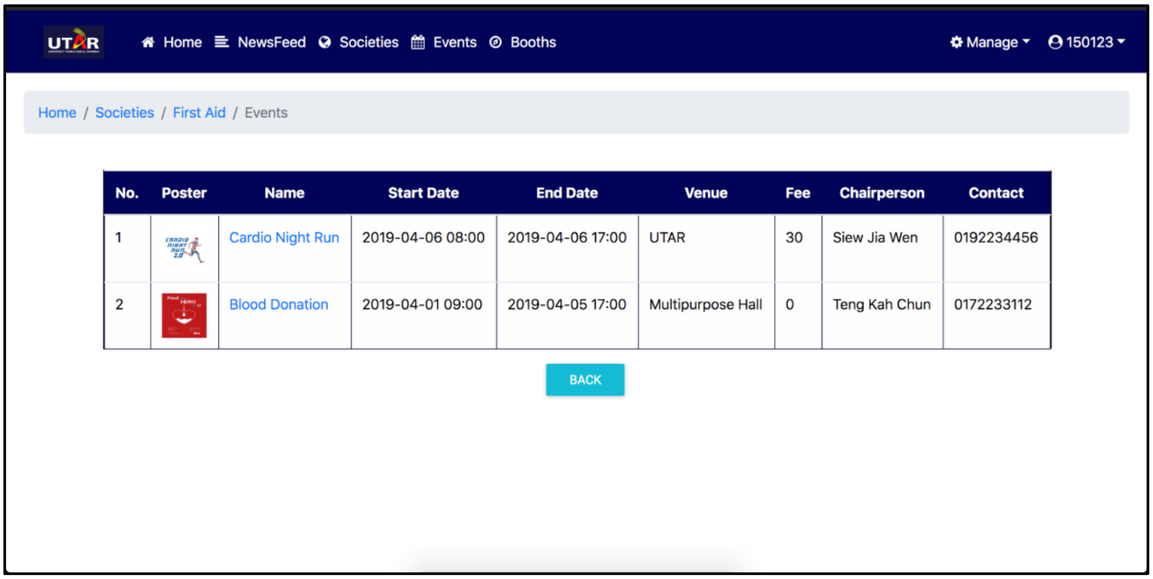
Mar 26 2019 01:38 AM

Figure 5.11: List of news feeds for societies and events



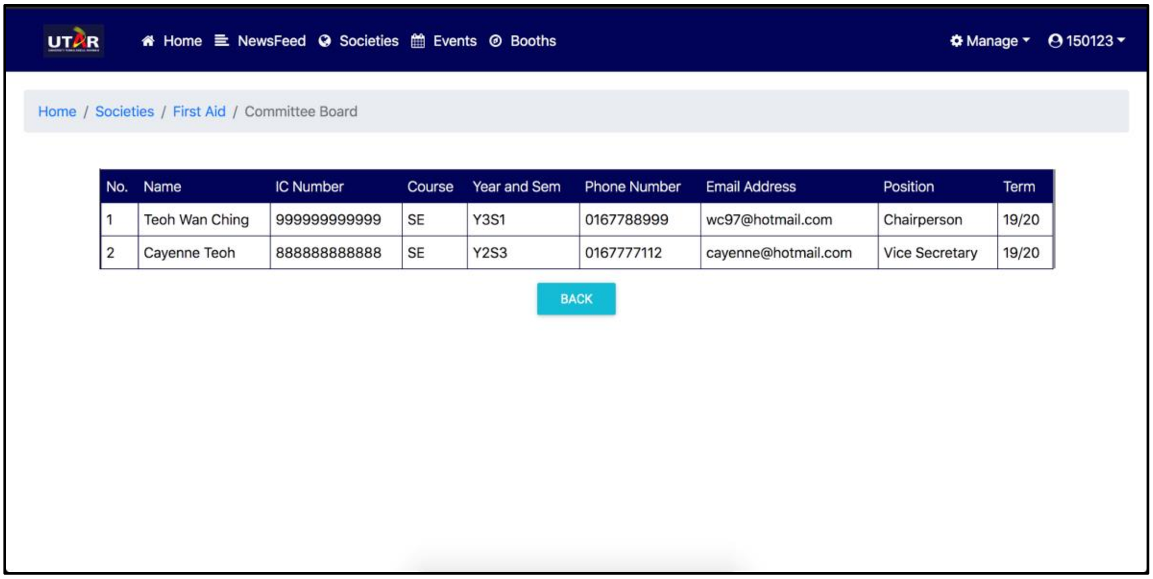
No.	Society	Booth Number
1	Information Technology	A5
2	TestMyTest	A3
3	Test	B6

Figure 5.12: List of registered booths by societies and events



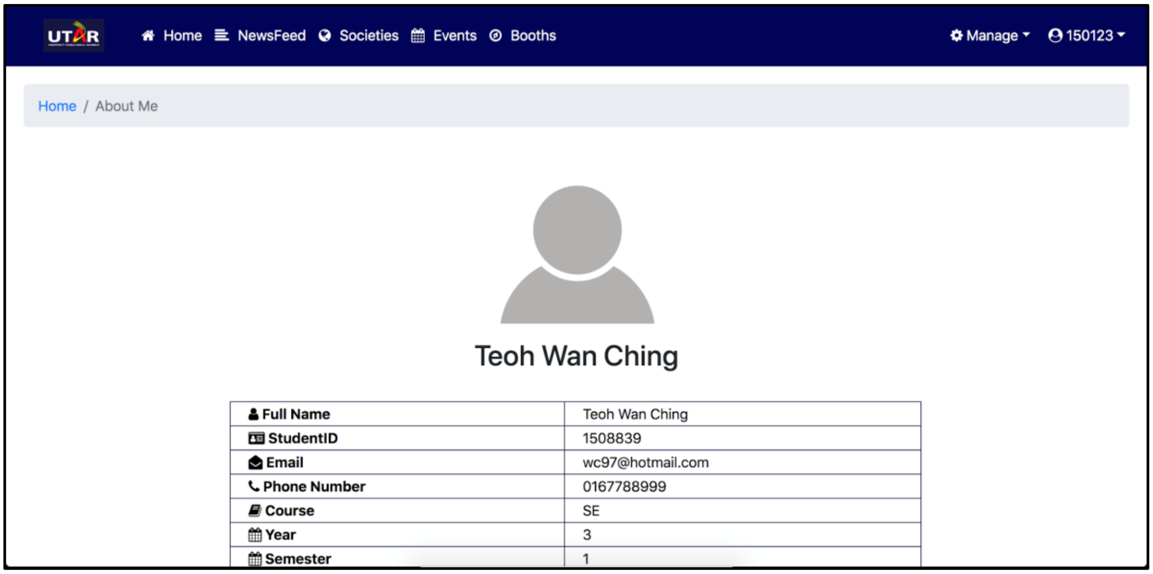
No.	Poster	Name	Start Date	End Date	Venue	Fee	Chairperson	Contact
1		Cardio Night Run	2019-04-06 08:00	2019-04-06 17:00	UTAR	30	Siew Jia Wen	0192234456
2		Blood Donation	2019-04-01 09:00	2019-04-05 17:00	Multipurpose Hall	0	Teng Kah Chun	0172233112

Figure 5.13: List of events organized by respective society



No.	Name	IC Number	Course	Year and Sem	Phone Number	Email Address	Position	Term
1	Teoh Wan Ching	999999999999	SE	Y3S1	0167788999	wc97@hotmail.com	Chairperson	19/20
2	Cayenne Teoh	888888888888	SE	Y2S3	0167777112	cayenne@hotmail.com	Vice Secretary	19/20

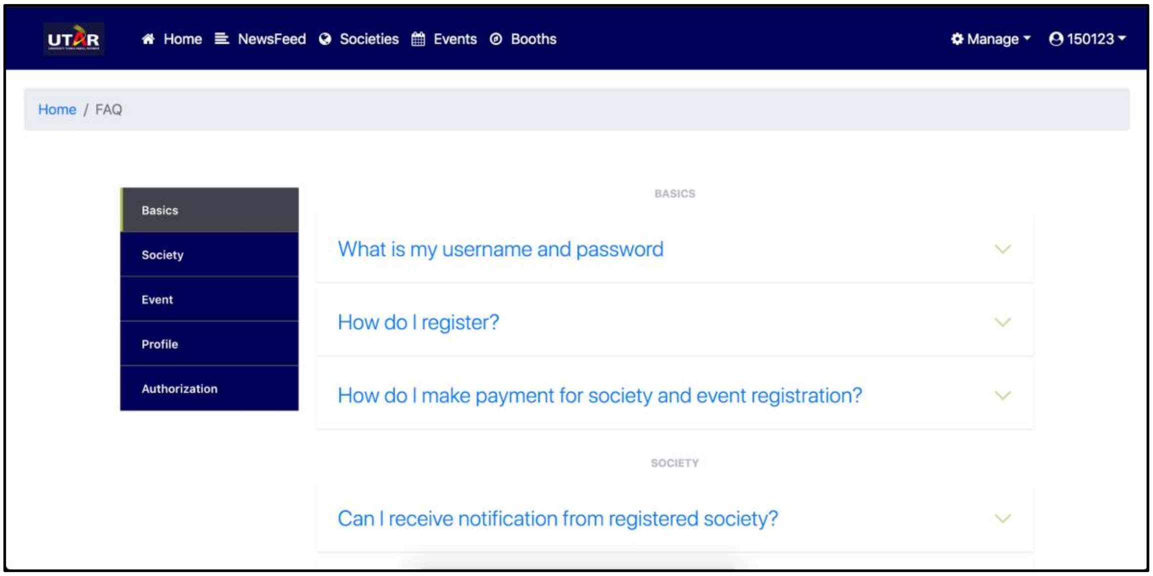
Figure 5.14: List of committees of the respective event



Teoh Wan Ching

Full Name	Teoh Wan Ching
StudentID	1508839
Email	wc97@hotmail.com
Phone Number	0167788999
Course	SE
Year	3
Semester	1

Figure 5.15: About Me page with personal details



Home / FAQ

Basics

Society

Event

Profile

Authorization

BASICS

What is my username and password

How do I register?

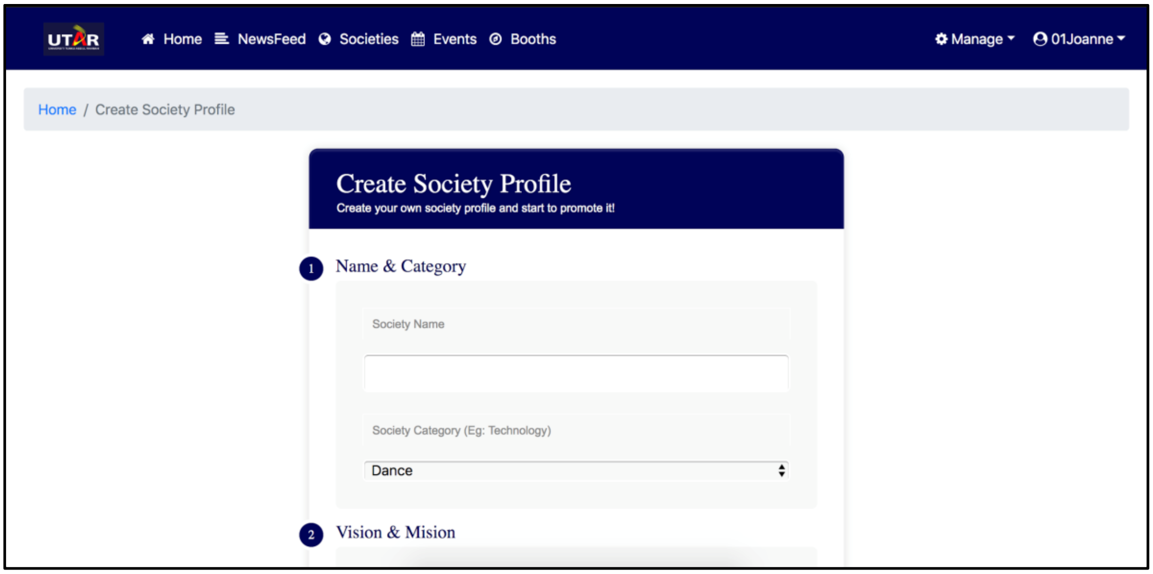
How do I make payment for society and event registration?

SOCIETY

Can I receive notification from registered society?

Figure 5.16: List of categorized FAQ

5.5.2 Management



Home / Create Society Profile

Create Society Profile

Create your own society profile and start to promote it!

1 Name & Category

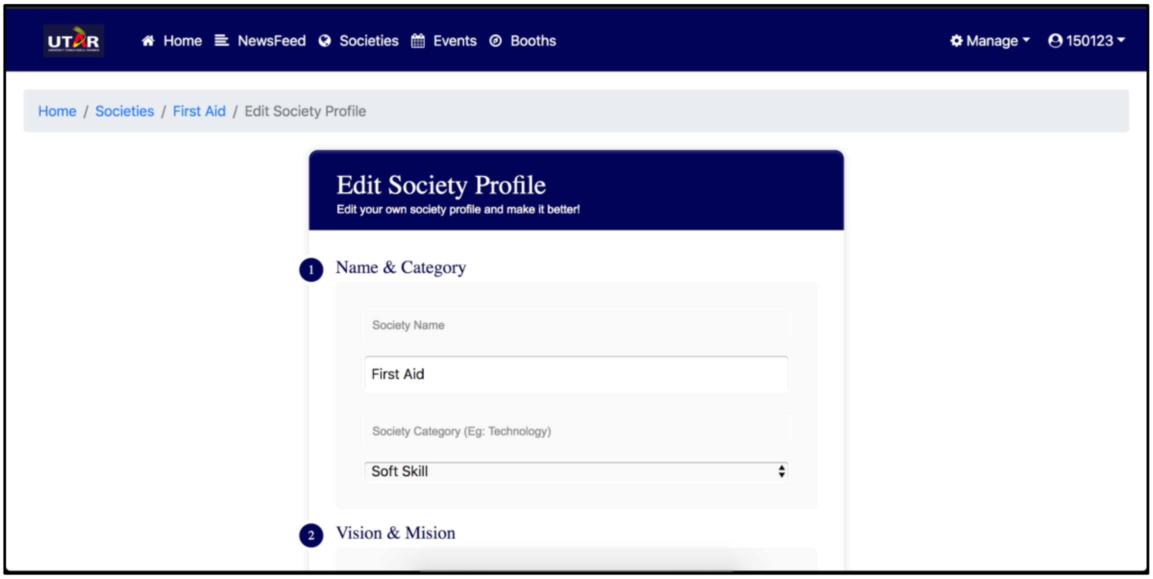
Society Name

Society Category (Eg: Technology)

Dance

2 Vision & Mision

Figure 5.17: Form to create society profile



Home / Societies / First Aid / Edit Society Profile

Edit Society Profile

Edit your own society profile and make it better!

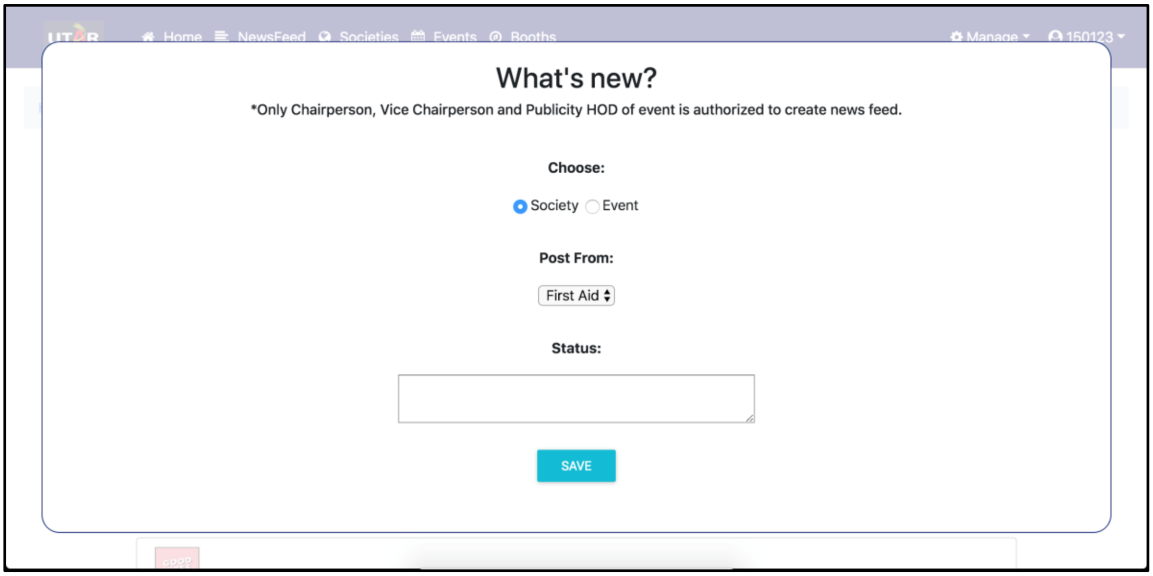
1 Name & Category

Society Name: First Aid

Society Category (Eg: Technology): Soft Skill

2 Vision & Mision

Figure 5.18: Form to edit society profile



What's new?

*Only Chairperson, Vice Chairperson and Publicity HOD of event is authorized to create news feed.

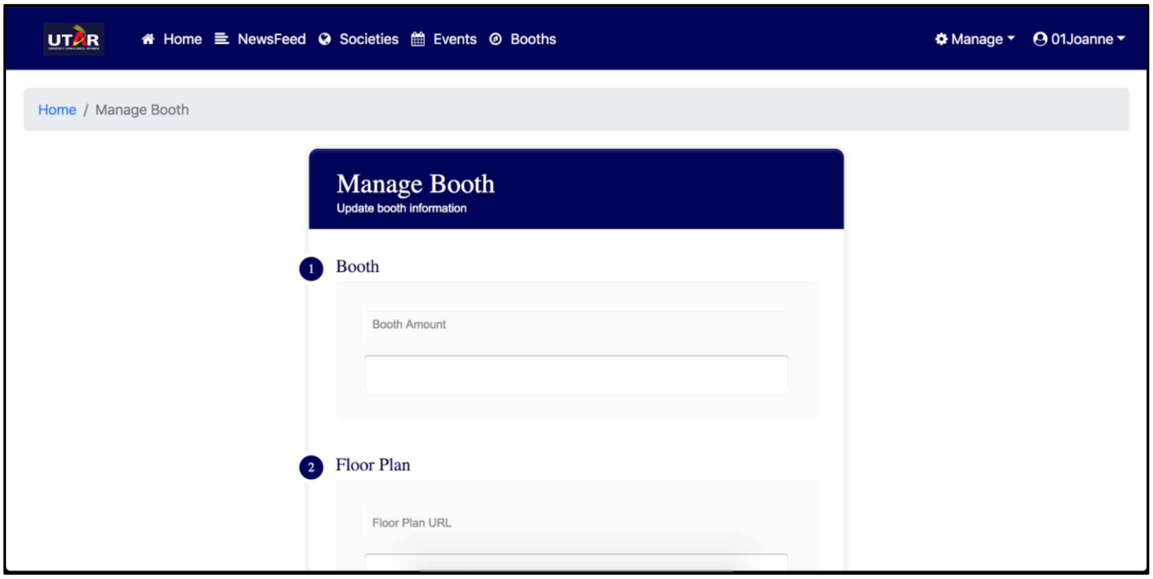
Choose:

● Society ○ Event

Post From: First Aid

Status:

Figure 5.19: Form to create news feeds



Manage Booth

Update booth information

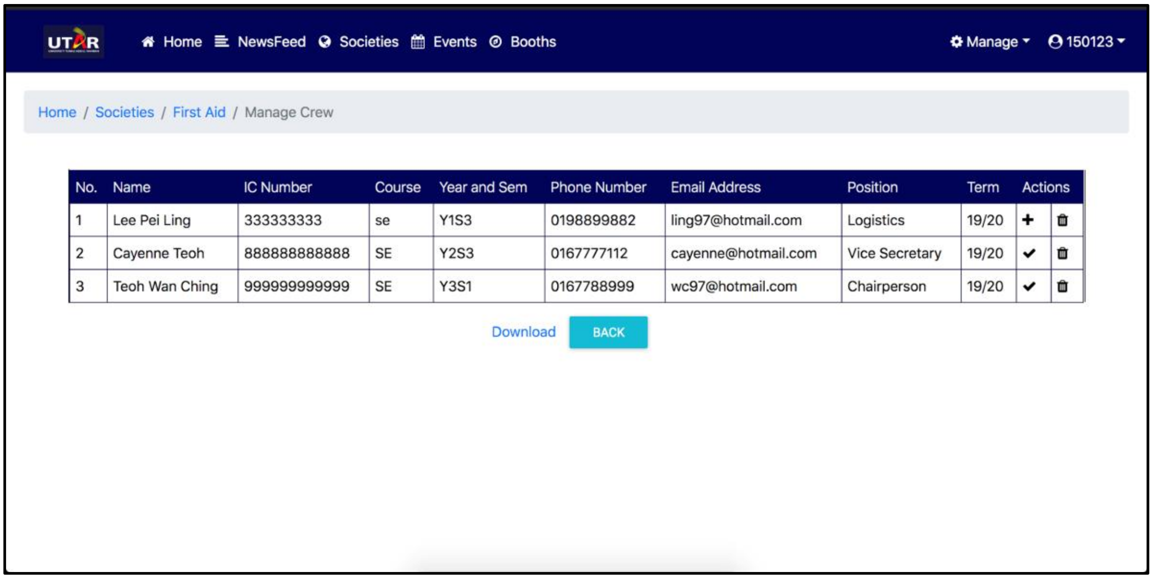
1 Booth

Booth Amount

2 Floor Plan

Floor Plan URL

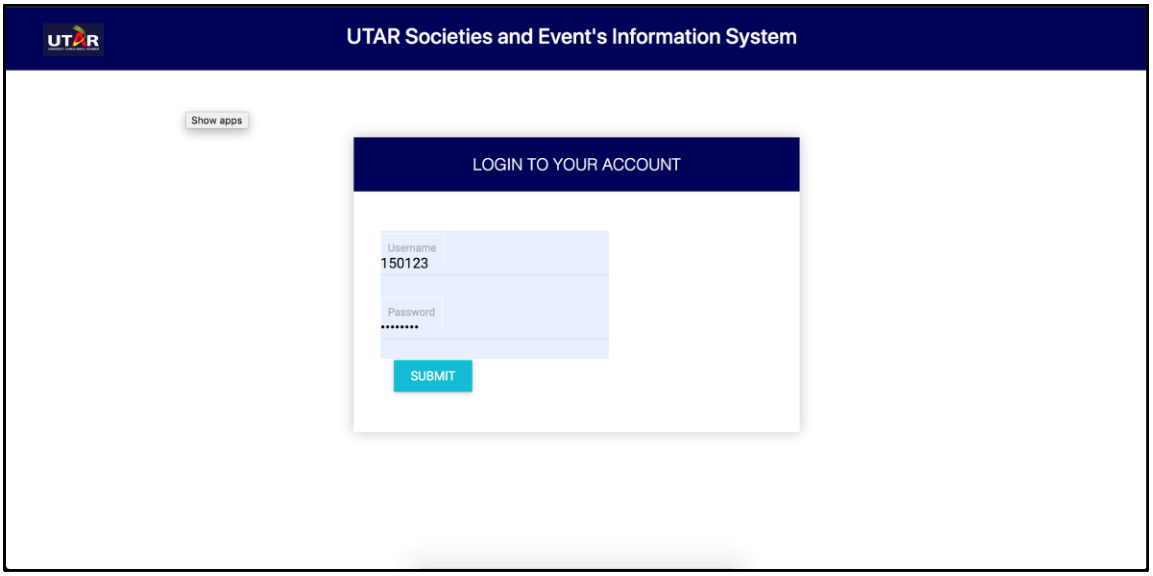
Figure 5.20: Form to manage booth registration



No.	Name	IC Number	Course	Year and Sem	Phone Number	Email Address	Position	Term	Actions	
1	Lee Pei Ling	333333333	se	Y1S3	0198899882	ling97@hotmail.com	Logistics	19/20	+	🗑
2	Cayenne Teoh	888888888888	SE	Y2S3	0167777112	cayenne@hotmail.com	Vice Secretary	19/20	✓	🗑
3	Teoh Wan Ching	999999999999	SE	Y3S1	0167788999	wc97@hotmail.com	Chairperson	19/20	✓	🗑

Figure 5.21: List of event crew to be managed

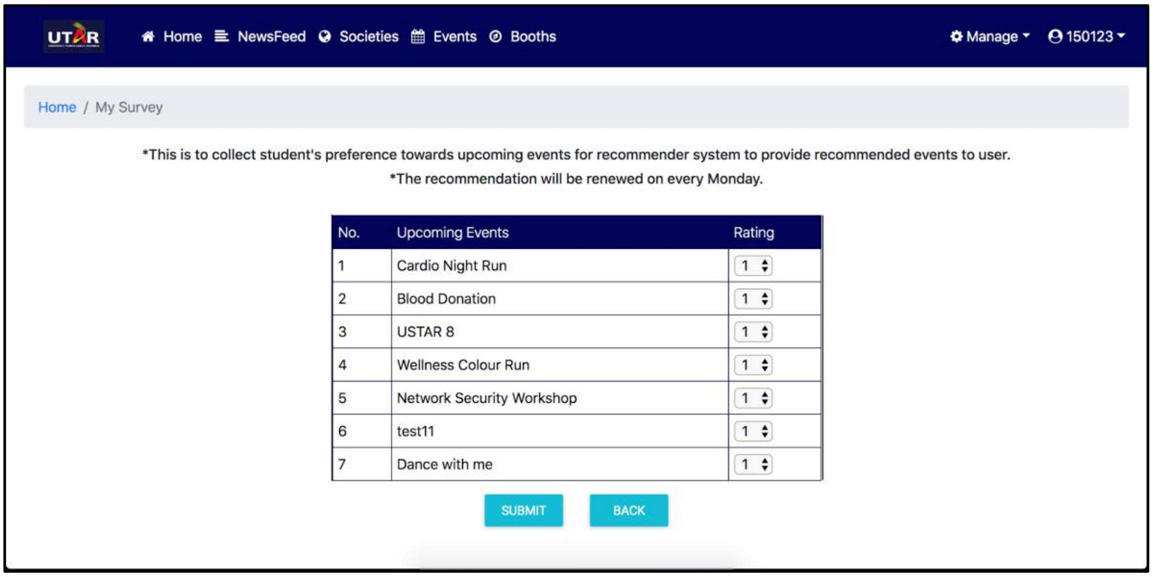
5.5.3 Others



Username: 150123

Password: ••••••••

Figure 5.22: Login page

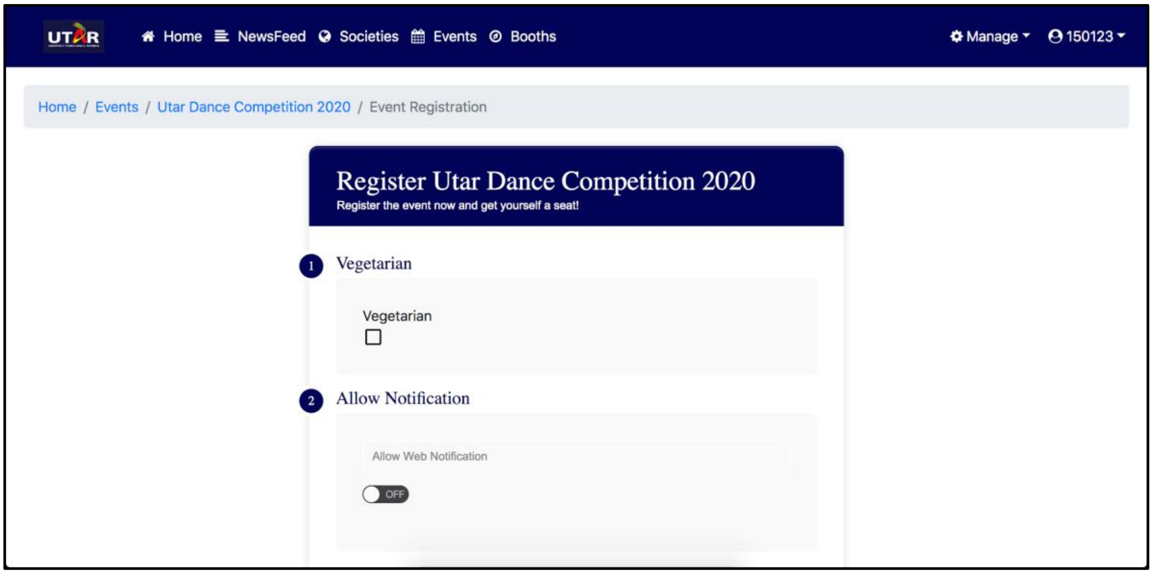


*This is to collect student's preference towards upcoming events for recommender system to provide recommended events to user.

*The recommendation will be renewed on every Monday.

No.	Upcoming Events	Rating
1	Cardio Night Run	1
2	Blood Donation	1
3	USTAR 8	1
4	Wellness Colour Run	1
5	Network Security Workshop	1
6	test11	1
7	Dance with me	1

Figure 5.23: List of events for student to provide rating


UTAR
Home
NewsFeed
Societies
Events
Booths
Manage
150123
Home / Events / Utar Dance Competition 2020 / Event Registration

Register Utar Dance Competition 2020

Register the event now and get yourself a seat!

1 Vegetarian

Vegetarian □

2 Allow Notification

Allow Web Notification

OFF

Figure 5.24: Form to register event

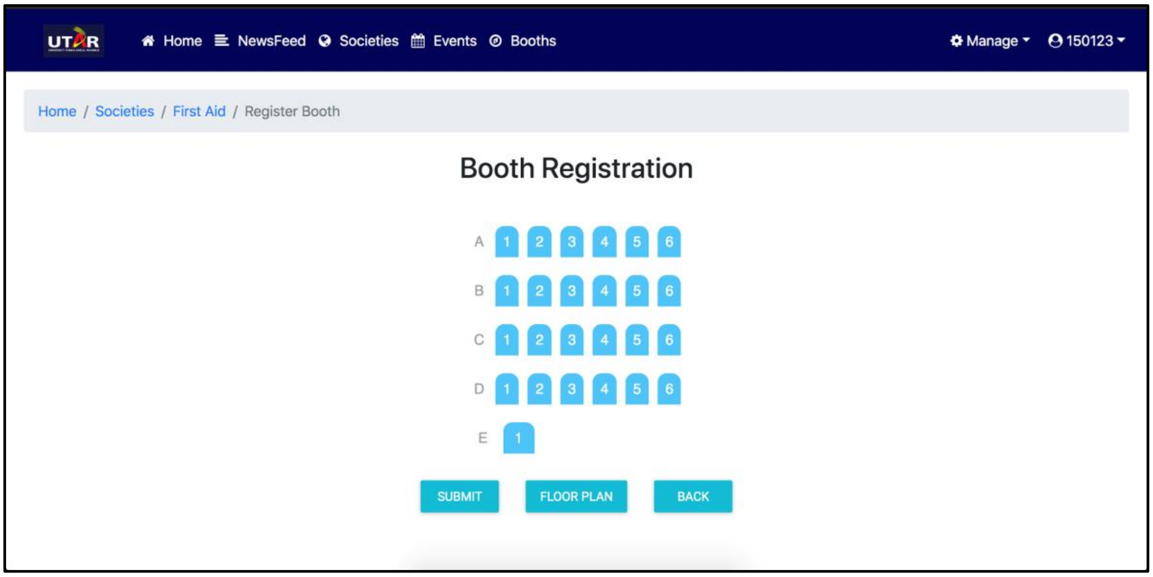


Figure 5.25: Booth registration view

CHAPTER 6

IMPLEMENTATION

This chapter shows the complete implementation of this system based on the constructed design.

6.1 Process Flow Diagram

Process flow diagrams describes the overall flow of the system features.

6.1.1 Recommender System

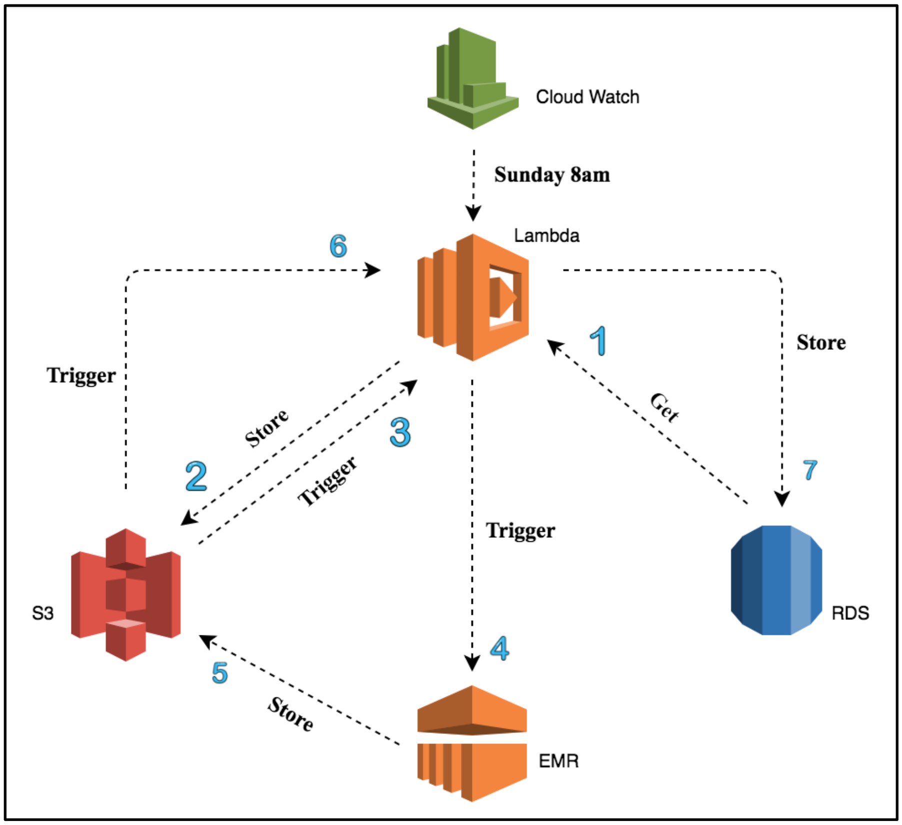


Figure 6.1: Recommender System using Amazon Web Service

Description:

1. Cloud watch triggers Lambda at 8am on every Sunday.
2. Lambda retrieves student rating information from RDS and store into S3.
3. S3 triggers Lambda after data is stored.
4. Lambda triggers EMR to train the prediction model.
5. EMR stores predicted data into S3 after the training is done.
6. S3 triggers Lambda.
7. Lambda stores predicted data to RDS (for front-end to retrieve).

6.1.2 Push Notification

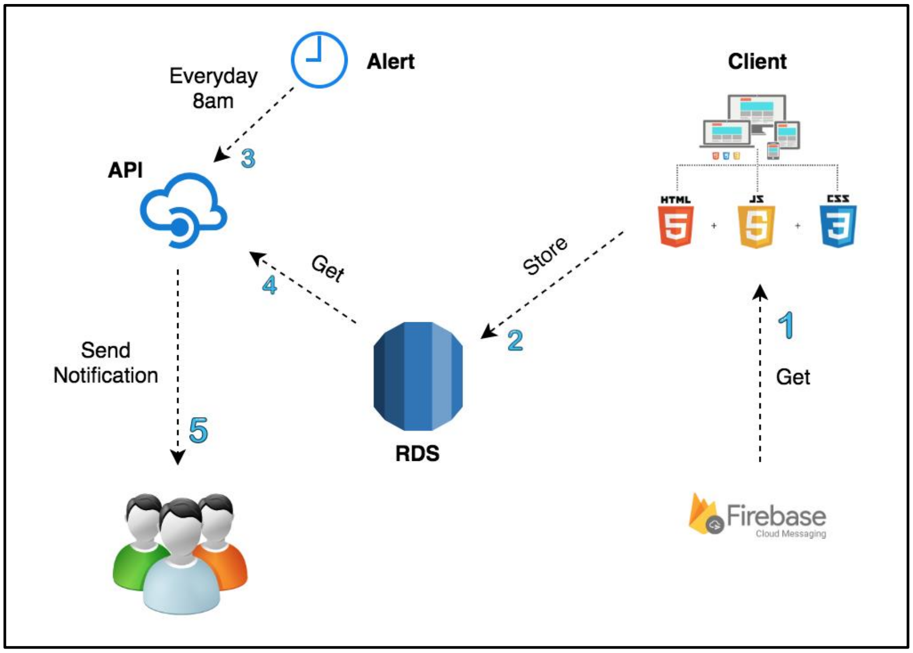


Figure 6.2: Notification using Amazon Web Service

Description:

1. Application retrieves Firebase Cloud Messaging (FCM) token.
2. Application stores FCM token into RDS.
3. Cloud Watch triggers an API everyday at 8am to retrieve student’s information (such as today’s event that student participated, students that allowed notification and FCM token) from RDS.
4. Notification is sent to group of users after retrieving data from RDS.

6.1.3 Real Time Implementation

Real time update is being implemented in this information system using socket.io which allows event-based and bidirectional event communication. Socket.io works in a way that event is being emitted from the server and the client is listening to the specific emitted event. The client will then execution function to update information on front-end to display for users in real-time. The figure below shows the overview of socket implementation between server and client.

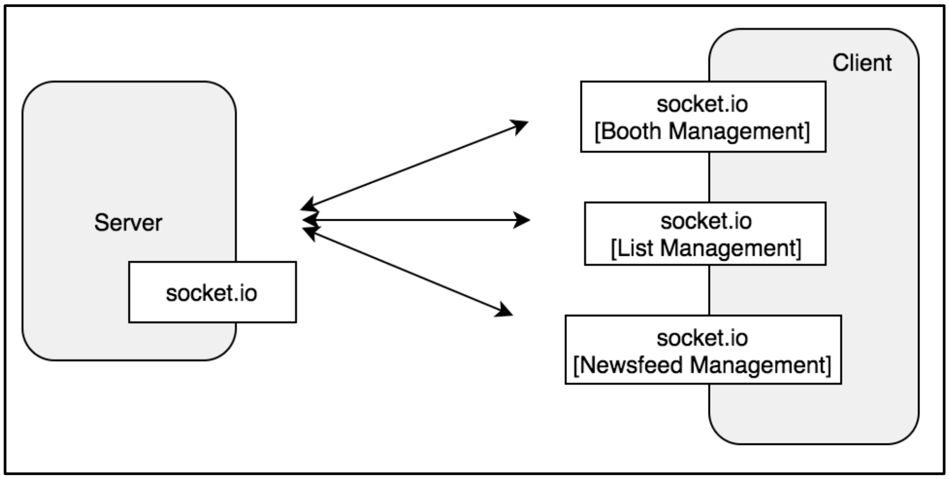


Figure 6.3: Socket Implementation

6.2 Activity Diagram

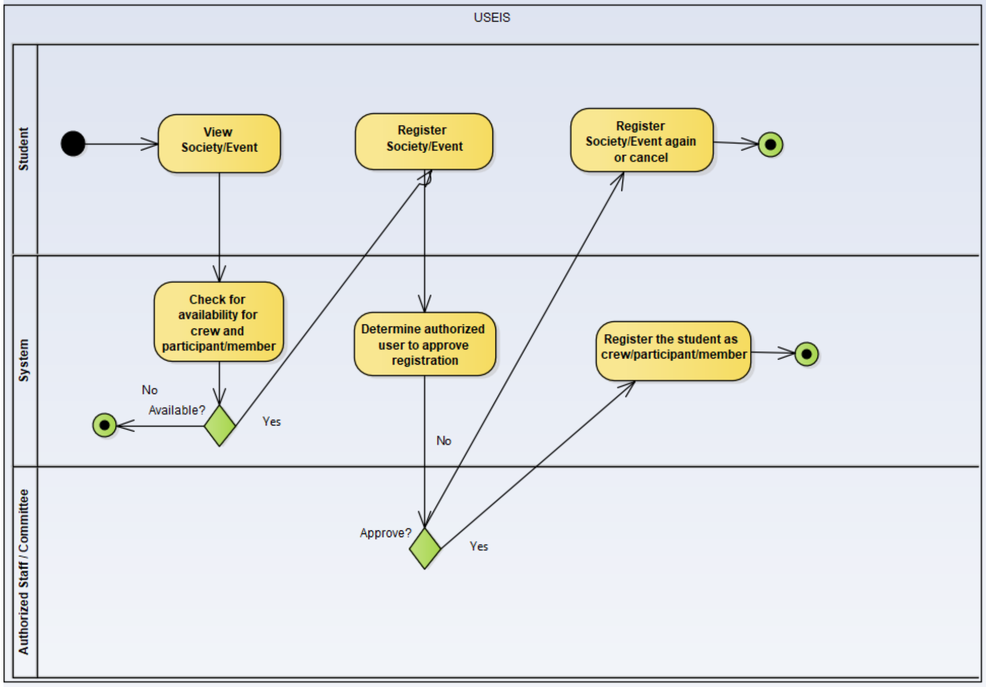

USEIS
Student
System
Authorized Staff / Committee
View Society/Event
Register Society/Event
Register Society/Event again or cancel
Check for availability for crew and participant/member
No
Available?
Yes
Determine authorized user to approve registration
No
Register the student as crew/participant/member
Approve?
Yes

Figure 6.4: Student registration for society / event

6.3 Component Hierarchy

Component hierarchy shows all components which have been created in the front-end development using React framework.

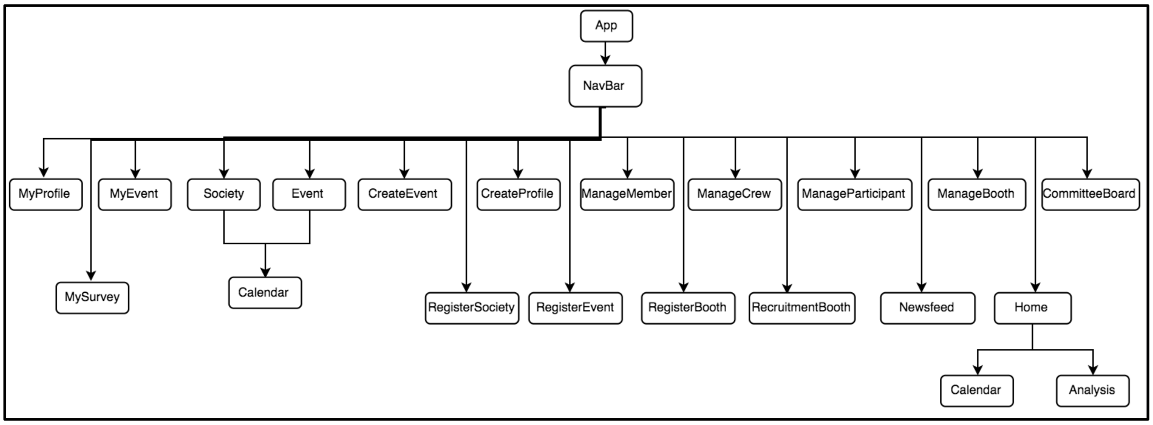


Figure 6.5: Component Hierarchy

6.4 RESTful Route Design

Routes are created for the each different action based on the tables in database design. The prefix of route is *"HOST_URL/api/v1"* for AWS API Gateway and *"HOST_URL/api/v1"* for AWS Beanstalk. The HOSR_URL represents the URL of the server which is *"https://b2auwy2dql.execute-api.ap-southeast-1.amazonaws.com/v1"*. The semicolon in some routes represents the path parameter such as route/type/:id. For instance, route/society/1 which means the ID of the society is 1.

GET method (/type)

Table 6.1: Route Design for *GET* method - 1

Possible Input	**Description**
/society	Return a list of societies.
/event	Return a list of all events
/eventInMonth	Return a list of upcoming events in a month
/calendarEvent	Return a list of upcoming events in calendar
/societyRole	Return a list of society roles
/eventRole	Return a list of event roles
/newsfeeds	Return a list of news feeds
/societyEvent	Return a list of events of all societies
/societyBooth	Return a list of booths of all societies
/eventBooth	Return a list of booths of all events
/allBooths	Return a list of all booths of societies and events
/recommendedSocieties	Return a list of recommended societies

GET Method (/type/id)

Table 6.2: Route Design for *GET* method - 2

Possible Input	Description
/society/:id	Return society based on society ID.
/event/:id	Return event based on event ID.
/studentSociety/:id	Return student's societies based on student ID.
/staffSociety/:id	Return staff's societies based on staff ID.
/studentEvent/:id	Return student's events based on ID.
/staffEvent/:id	Return staff's events based on staff ID.
/societyCrew/:id	Return society's crew based on society ID.
/eventCrew/:id	Return event's crew based on event ID.
/eventParticipant/:id	Return event's participants based on event ID.
/societyMember/:id	Return society's members based on society ID.
/crewPosition/:id	Return society's available crew positions based on society ID.
/eventAnalysis/:id	Return event's analysis based on ID.
/recommendation/:id	Return society based on ID.

GET method (/type/id/userId): where id is either society or event id

Table 6.3: Route Design for *GET* method - 3

Possible Input	Description
/studentSociety/:id/:userId	Return student's societies based on society ID and student ID.

/staffEvent/:id/:userId	Return staff's events based on event ID and staff ID.
/studentEvent/:id/:userId	Return student's events based on event ID and student ID.
/staffEvent/:id/:userId	Return staff's events based on event ID and staff ID.

POST Method (/login)

Table 6.4: Route Design for *POST* method - 1

Possible Input	**Description**
/login	Login user and return result to indicate the login is success.

POST Method (/type)

Table 6.5: Route Design for *POST* method - 2

Possible Input	**Description**
/society	Create new society and return result to indicate the creation of society is success.
/event	Create new event and return result to indicate the creation of event is success.
/newsfeeds	Create new news feed and return result to indicate the creation of news feed is success.
/studentEventRating	Create new student's event rating and return result to indicate the creation of event rating is success.

POST Method (/register/type)

Table 6.6: Route Design for *POST* method - 3

Possible Input	**Description**
/register/studentSociety	Create a new society entry for student.
/register/staffSociety	Create a new society entry for staff.
/register/eventCrew	Create a new event entry for student as crew.
/register/studentEvent	Create a new society entry for student as participant.
/register/staffEvent	Create a new event entry for staff as participant.

PUT Method (/type)

Table 6.7: Route Design for *PUT* method - 1

Possible Input	**Description**
/societyCrew	Update student’s status as society crew after approval.
/eventCrew	Update student’s status as event crew after approval.
/member	Update student’s status as society member after approval.
/studentParticipant	Update student’s status as event participant after approval.
/staffParticipant	Update staff status as event participant after approval.
/rejectSocietyCrew	Update student’s status as society crew after rejection.
/rejectEventCrew	Update student’s status as event crew

	after rejection.
/rejectStudentEvent	Update student’s status as event participant after rejection.
/rejectStaffEvent	Update staff status as event participant after rejection.
/resubmitStudentParticipant	Update student status as event participant after resubmission.
/cancelStudentSociety	Update student status as society member after cancellation.
/cancelStudentEvent	Update student status as event participant after cancellation.

PUT Method (/type/id)

Table 6.8: Route Design for *PUT* method - 2

Possible Input	**Description**
/society/:id	Update society based on society ID.
/event/:id	Update event based on event ID.
/studentRating/:id	Update student’s event rating based on student ID.
/staffRating/:id	Update staff’s event rating based on staff ID.

DELETE Method (/type/id)

Table 6.9: Route Design for *DELETE* method - 1

Possible Input	**Description**
/sNewsfeeds/:id	Delete society's news feed based on news feed ID.
/eNewsfeeds/:id	Delete event's news feed based on news feed ID.
/event/:id	Delete event based on news event ID.

DELETE Method (/type/id/eventId)

Table 6.10: Route Design for *DELETE* method - 2

Possible Input	**Description**
/studentEvent/:id/:eventId	Delete student's participation in event based on student ID and event ID.
/staffEvent/:id/:eventId	Delete staff's participation in event based on staff ID and event ID.
/studentSociety/:id/:eventId	Delete student's participation in society based on student ID and society ID.
/staffSociety/:id/:eventId	Delete staff's participation in society based on student ID and society ID.

6.5 Role-based Access Control Diagram

Role-Based Access Control is implemented in this information system to ensure that every user only authorized to certain rights according to their roles. There are a few rules for the authorization in this system as shown in the figure below.

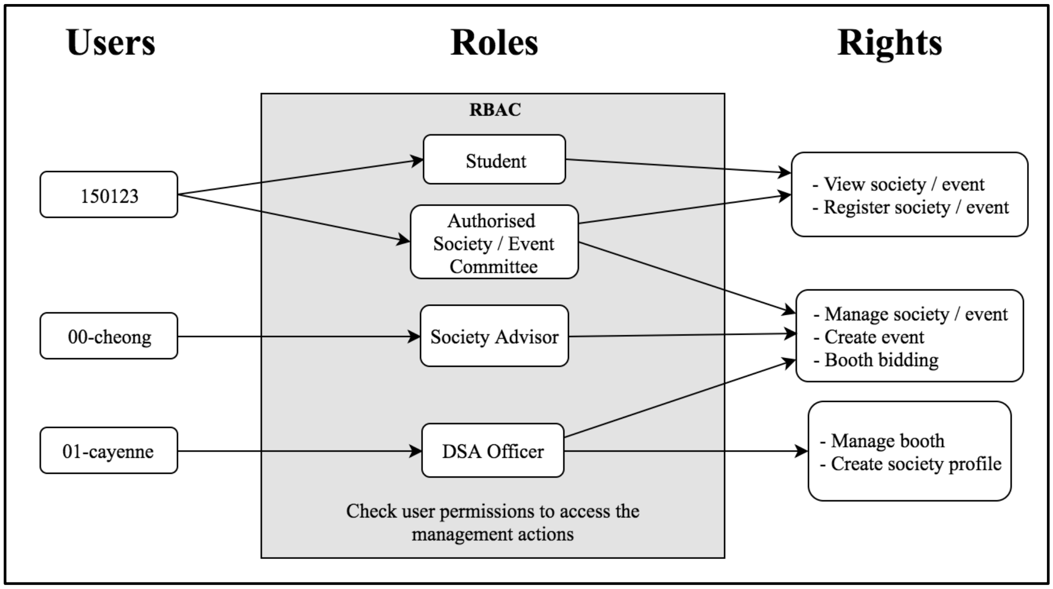
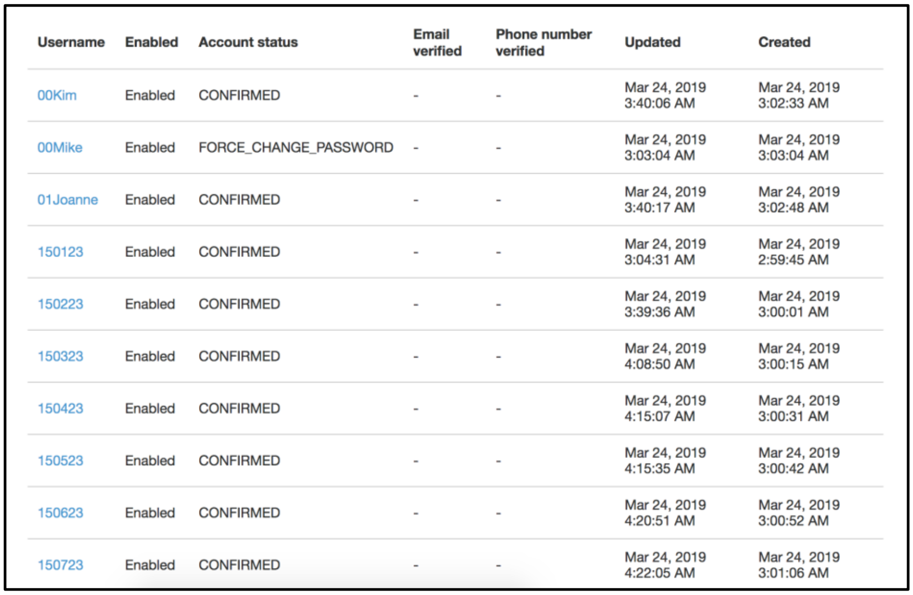


Figure 6.6: Role-based Access Control Diagram

6.6 Implementation of cloud services in Amazon Web Services

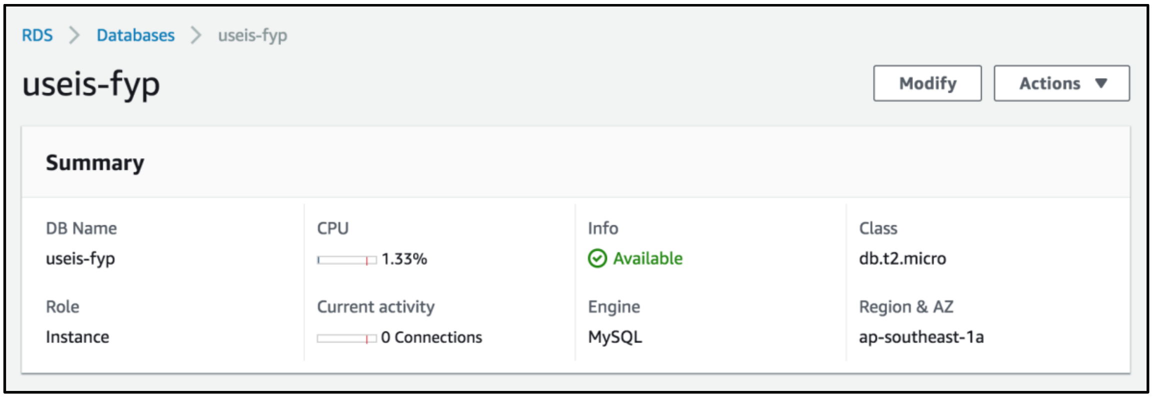
6.6.1 Amazon Cognito



Username	Enabled	Account status	Email verified	Phone number verified	Updated	Created
00Kim	Enabled	CONFIRMED	-	-	Mar 24, 2019 3:40:06 AM	Mar 24, 2019 3:02:33 AM
00Mike	Enabled	FORCE_CHANGE_PASSWORD	-	-	Mar 24, 2019 3:03:04 AM	Mar 24, 2019 3:03:04 AM
01Joanne	Enabled	CONFIRMED	-	-	Mar 24, 2019 3:40:17 AM	Mar 24, 2019 3:02:48 AM
150123	Enabled	CONFIRMED	-	-	Mar 24, 2019 3:04:31 AM	Mar 24, 2019 2:59:45 AM
150223	Enabled	CONFIRMED	-	-	Mar 24, 2019 3:39:36 AM	Mar 24, 2019 3:00:01 AM
150323	Enabled	CONFIRMED	-	-	Mar 24, 2019 4:08:50 AM	Mar 24, 2019 3:00:15 AM
150423	Enabled	CONFIRMED	-	-	Mar 24, 2019 4:15:07 AM	Mar 24, 2019 3:00:31 AM
150523	Enabled	CONFIRMED	-	-	Mar 24, 2019 4:15:35 AM	Mar 24, 2019 3:00:42 AM
150623	Enabled	CONFIRMED	-	-	Mar 24, 2019 4:20:51 AM	Mar 24, 2019 3:00:52 AM
150723	Enabled	CONFIRMED	-	-	Mar 24, 2019 4:22:05 AM	Mar 24, 2019 3:01:06 AM

Figure 6.7: Registered users in Amazon Cognito

6.6.2 Amazon Relational Database Service



useis-fyp

Summary

DB Name
useis-fyp

Role
Instance

CPU
1.33%

Current activity
0 Connections

Info
Available

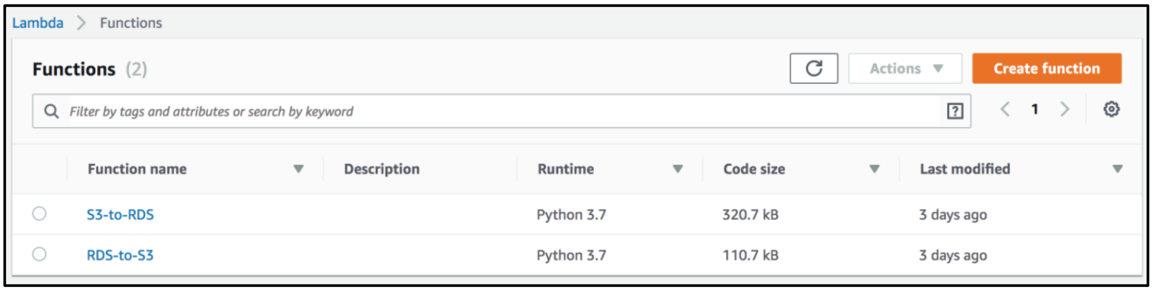
Engine
MySQL

Class
db.t2.micro

Region & AZ
ap-southeast-1a

Figure 6.8: Database Configuration of RDS

6.6.3 AWS Lambda

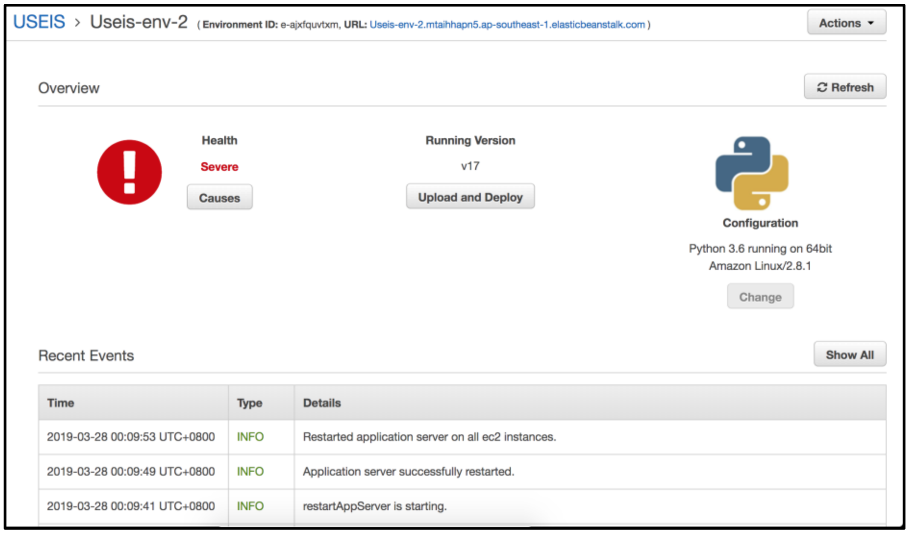


Functions (2)

	Function name	Description	Runtime	Code size	Last modified
○	S3-to-RDS		Python 3.7	320.7 kB	3 days ago
○	RDS-to-S3		Python 3.7	110.7 kB	3 days ago

Figure 6.9: Two running functions in AWS Lambda

6.6.4 AWS Elastic Beanstalk



(**Environment ID:** e-ajxfquvtxm, **URL:** Useis-env-2.mtaihhapn5.ap-southeast-1.elasticbeanstalk.com)

Health: Severe

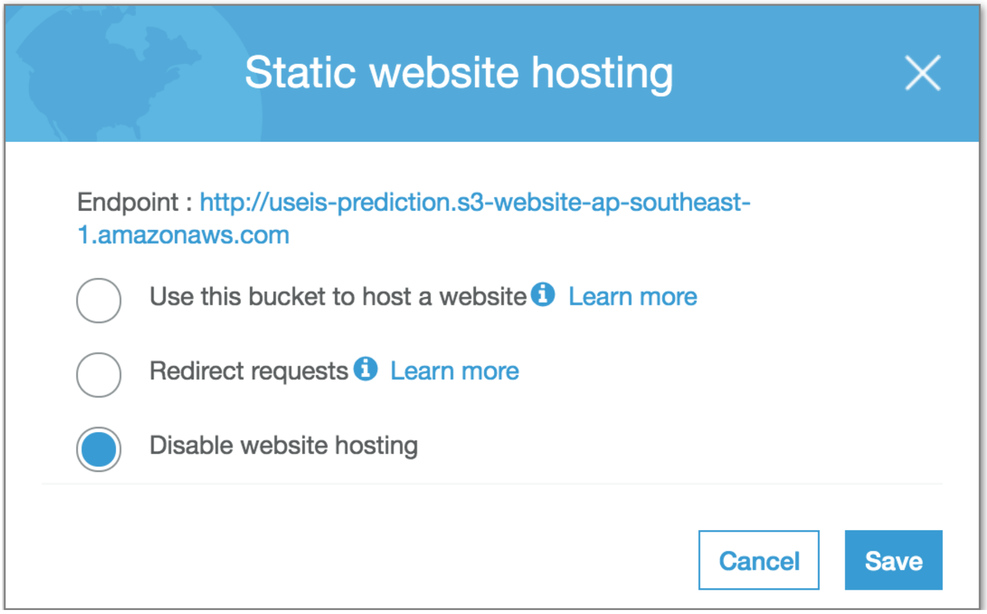
Running Version: v17

Configuration: Python 3.6 running on 64bit Amazon Linux/2.8.1

Time	Type	Details
2019-03-28 00:09:53 UTC+0800	INFO	Restarted application server on all ec2 instances.
2019-03-28 00:09:49 UTC+0800	INFO	Application server successfully restarted.
2019-03-28 00:09:41 UTC+0800	INFO	restartAppServer is starting.

Figure 6.10: Running instance in AWS Elastic Beanstalk in Python 3.6

6.6.5 Amazon Simple Storage Service (S3)



Static website hosting

Endpoint : http://useis-prediction.s3-website-ap-southeast-1.amazonaws.com

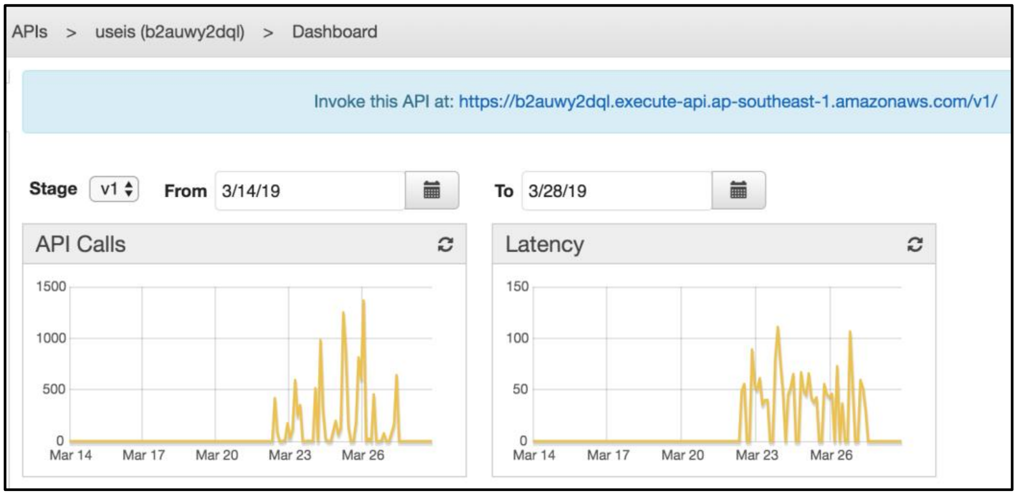
○ Use this bucket to host a website

○ Redirect requests

● Disable website hosting

Figure 6.11: Static Hosting of React in S3

6.6.6 Amazon API Gateway



Invoke this API at: https://b2auwy2dql.execute-api.ap-southeast-1.amazonaws.com/v1/

Stage v1 From 3/14/19 To 3/28/19

Figure 6.12: Dashboard of Amazon API Gateway

CHAPTER 7

TESTING AND EVALUATION

This chapter describes various types of testing which have been conducted for this information system.

7.1 Functional Testing

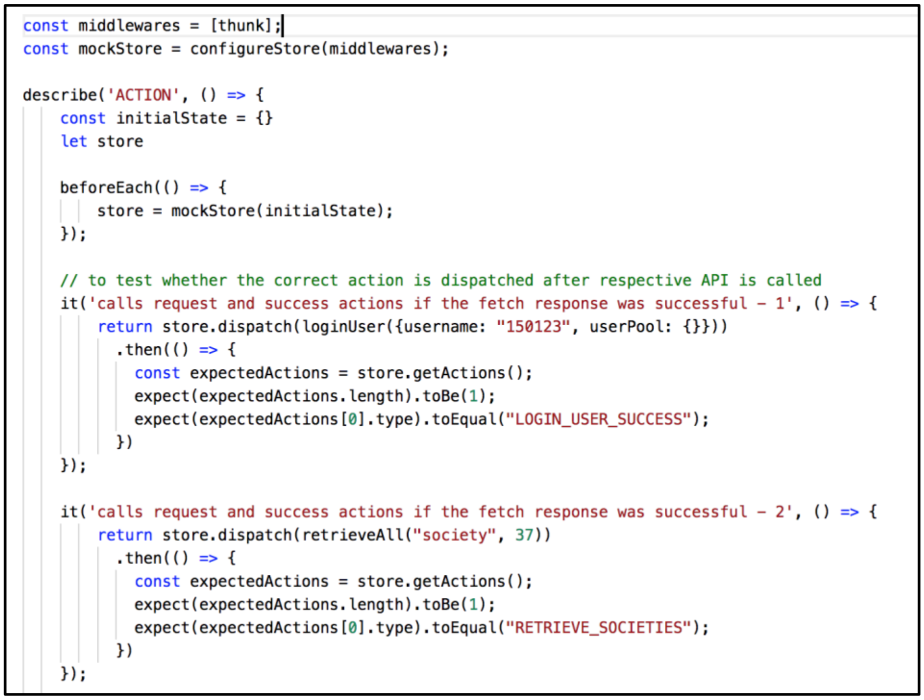
Functional Testing is type of testing where the requirements of a system is being tested by giving input and evaluate the output. This is to ensure that the system satisfies the functional requirements.

7.1.1 Unit Testing

Unit testing is being used to test each individual component or module of the software application. The purpose of Unit Testing is to ensure that each component is well performing according to the design plan.

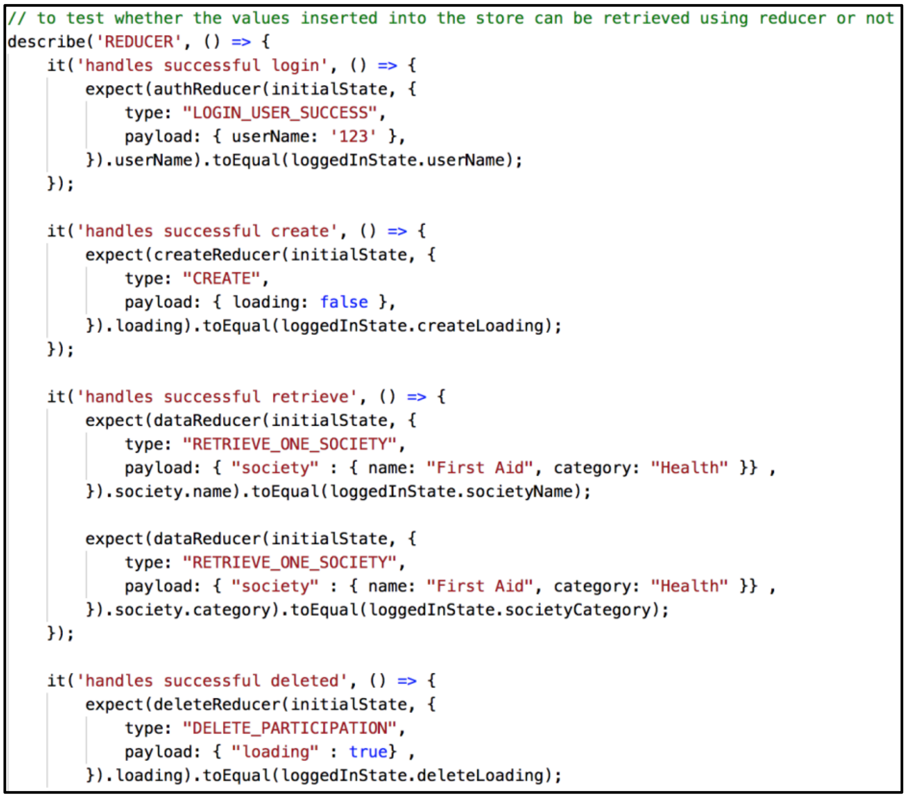
(a) Unit Testing for React

Unit Testing has been performed in front-end development using Enzyme and Jest, which are the libraries designed to test React application. Enzyme is only applicable to React but Jest is applicable for all Javascript applications. There are two types of unit testing implemented in front-end development, testing for redux (actions and reducers) and React components. The figures below shows parts of the test code for Unit Testing.



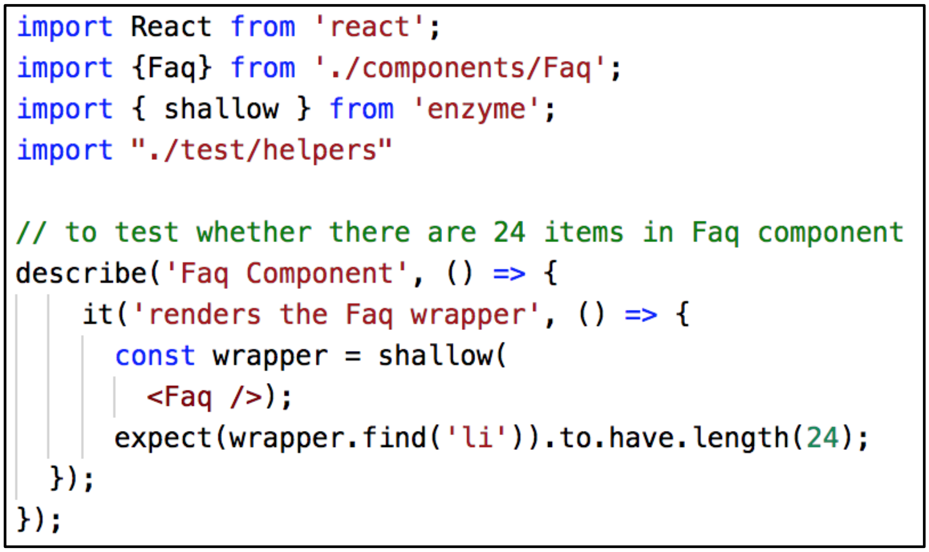
```
const middlewares = [thunk];
const mockStore = configureStore(middlewares);

describe('ACTION', () => {
    const initialState = {}
    let store

    beforeEach(() => {
        store = mockStore(initialState);
    });

    // to test whether the correct action is dispatched after respective API is called
    it('calls request and success actions if the fetch response was successful - 1', () => {
        return store.dispatch(loginUser({username: "150123", userPool: {}}))
          .then(() => {
            const expectedActions = store.getActions();
            expect(expectedActions.length).toBe(1);
            expect(expectedActions[0].type).toEqual("LOGIN_USER_SUCCESS");
          })
    });

    it('calls request and success actions if the fetch response was successful - 2', () => {
        return store.dispatch(retrieveAll("society", 37))
          .then(() => {
            const expectedActions = store.getActions();
            expect(expectedActions.length).toBe(1);
            expect(expectedActions[0].type).toEqual("RETRIEVE_SOCIETIES");
          })
    });
```

Figure 7.1: Test script for redux action using Jest and react-thunk



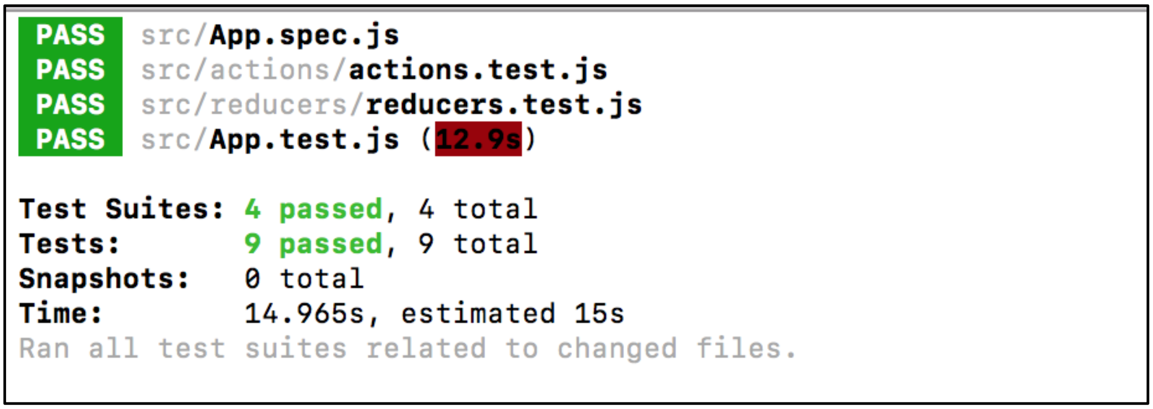
```
// to test whether the values inserted into the store can be retrieved using reducer or not
describe('REDUCER', () => {
    it('handles successful login', () => {
        expect(authReducer(initialState, {
            type: "LOGIN_USER_SUCCESS",
            payload: { userName: '123' },
        }).userName).toEqual(loggedInState.userName);
    });

    it('handles successful create', () => {
        expect(createReducer(initialState, {
            type: "CREATE",
            payload: { loading: false },
        }).loading).toEqual(loggedInState.createLoading);
    });

    it('handles successful retrieve', () => {
        expect(dataReducer(initialState, {
            type: "RETRIEVE_ONE_SOCIETY",
            payload: { "society" : { name: "First Aid", category: "Health" }} ,
        }).society.name).toEqual(loggedInState.societyName);

        expect(dataReducer(initialState, {
            type: "RETRIEVE_ONE_SOCIETY",
            payload: { "society" : { name: "First Aid", category: "Health" }} ,
        }).society.category).toEqual(loggedInState.societyCategory);
    });

    it('handles successful deleted', () => {
        expect(deleteReducer(initialState, {
            type: "DELETE_PARTICIPATION",
            payload: { "loading" : true} ,
        }).loading).toEqual(loggedInState.deleteLoading);
```

Figure 7.2: Test script for redux reducer using Jest and react-thunk



```
import React from 'react';
import {Faq} from './components/Faq';
import { shallow } from 'enzyme';
import "./test/helpers"

// to test whether there are 24 items in Faq component
describe('Faq Component', () => {
    it('renders the Faq wrapper', () => {
      const wrapper = shallow(
        <Faq />);
      expect(wrapper.find('li')).to.have.length(24);
  });
});
```

Figure 7.3: Test script for component rendering using Enzyme

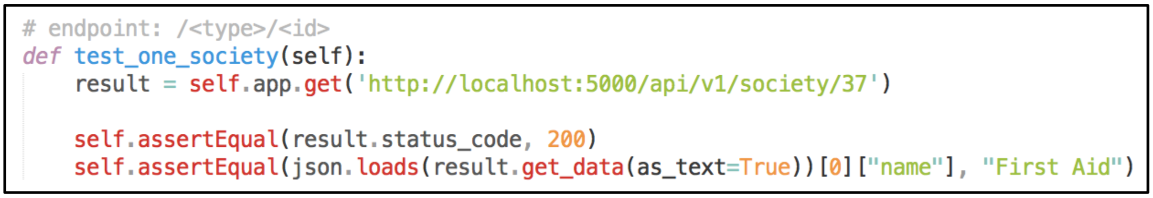


```
 PASS  src/App.spec.js
 PASS  src/actions/actions.test.js
 PASS  src/reducers/reducers.test.js
 PASS  src/App.test.js (12.9s)

Test Suites: 4 passed, 4 total
Tests:       9 passed, 9 total
Snapshots:   0 total
Time:        14.965s, estimated 15s
Ran all test suites related to changed files.
```

Figure 7.4: Unit Test Result for React

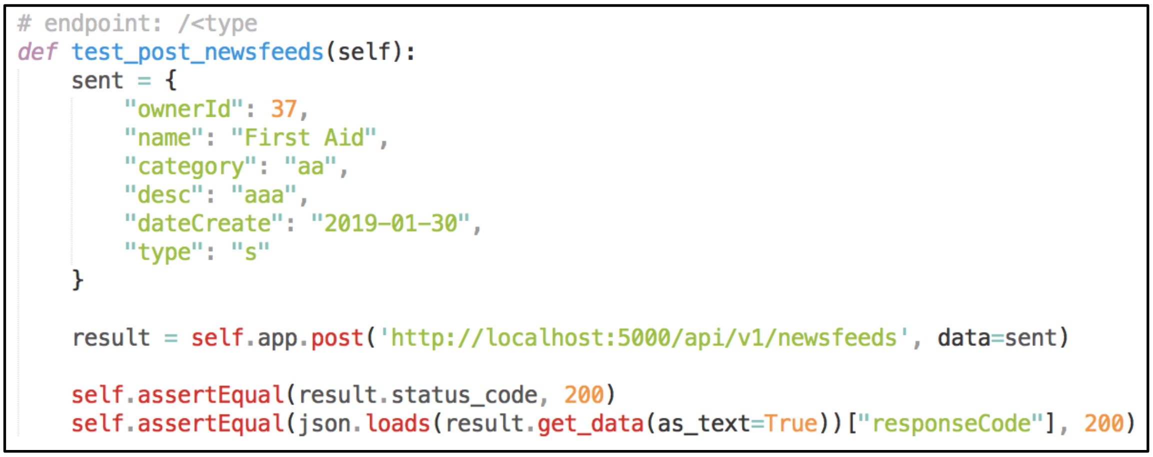
(b) Unit Testing for Flask

Werkzeug test Client which is provided by Flask has been implemented for API endpoint testing in back-end development. All endpoint methods which are *GET, POST, PUT and DELETE* has been fully tested.



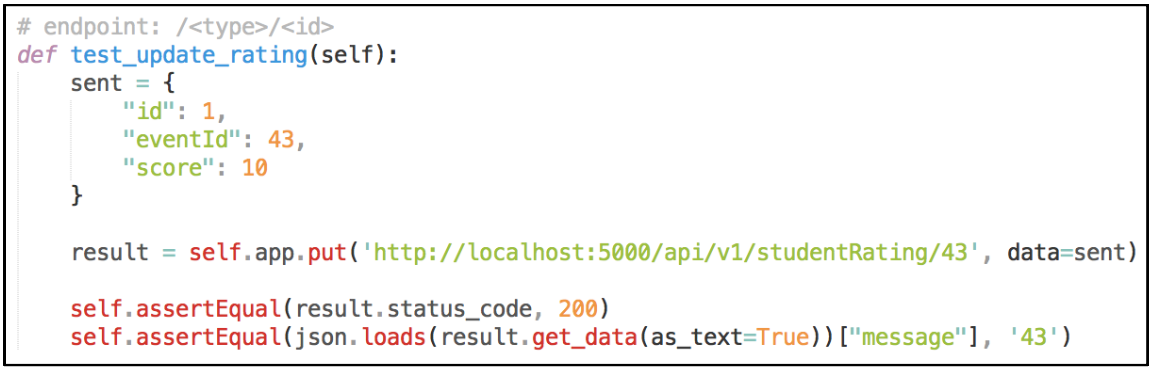
```
# endpoint: /<type>/<id>
def test_one_society(self):
    result = self.app.get('http://localhost:5000/api/v1/society/37')

    self.assertEqual(result.status_code, 200)
    self.assertEqual(json.loads(result.get_data(as_text=True))[0]["name"], "First Aid")
```

Figure 7.5: Test script for *GET* method



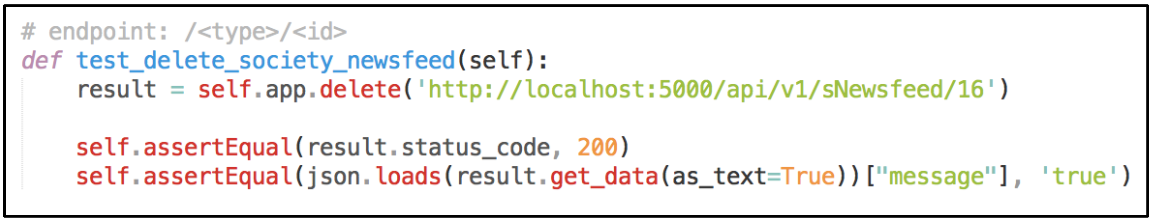
```
# endpoint: /<type
def test_post_newsfeeds(self):
    sent = {
        "ownerId": 37,
        "name": "First Aid",
        "category": "aa",
        "desc": "aaa",
        "dateCreate": "2019-01-30",
        "type": "s"
    }

    result = self.app.post('http://localhost:5000/api/v1/newsfeeds', data=sent)

    self.assertEqual(result.status_code, 200)
    self.assertEqual(json.loads(result.get_data(as_text=True))["responseCode"], 200)
```

Figure 7.6: Test script for *POST* method



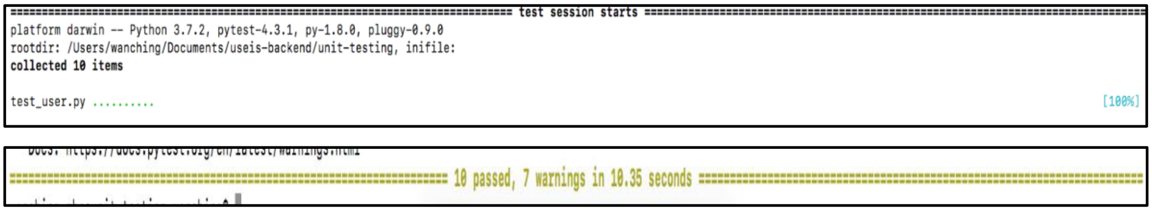
```
# endpoint: /<type>/<id>
def test_update_rating(self):
    sent = {
        "id": 1,
        "eventId": 43,
        "score": 10
    }

    result = self.app.put('http://localhost:5000/api/v1/studentRating/43', data=sent)

    self.assertEqual(result.status_code, 200)
    self.assertEqual(json.loads(result.get_data(as_text=True))["message"], '43')
```

Figure 7.7: Test script for *PUT* method



```
# endpoint: /<type>/<id>
def test_delete_society_newsfeed(self):
    result = self.app.delete('http://localhost:5000/api/v1/sNewsfeed/16')

    self.assertEqual(result.status_code, 200)
    self.assertEqual(json.loads(result.get_data(as_text=True))["message"], 'true')
```

Figure 7.8: Test script for *DELETE* method



```
=============================================== test session starts ================================================
platform darwin -- Python 3.7.2, pytest-4.3.1, py-1.8.0, pluggy-0.9.0
rootdir: /Users/wanching/Documents/useis-backend/unit-testing, inifile:
collected 10 items

test_user.py ..........                                                                                     [100%]
```

```
  Docs: https://docs.pytest.org/en/latest/warnings.html
=========================================== 10 passed, 7 warnings in 10.35 seconds ===========================================
```

Figure 7.9: Test result for Flask

7.1.2 Integration Testing

Integration Testing is being used to test the combined individual units or modules of a system. The purpose of this testing is to investigate the faults when groups of units are integrating together. The table in Appendix G: Integration Test Cases shows the integration test cases of every main feature of this information system.

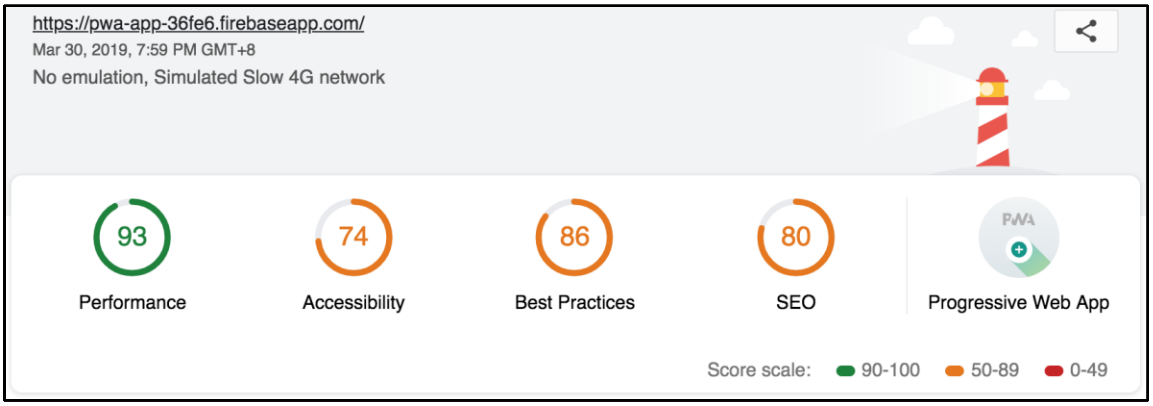
Please refer to “Appendix G: Integration Test Cases”.

7.2 Non-Functional Testing

Non-functional testing is type of testing where the non-functional requirements of a system is being tested such as the operation of the system instead of its specific behaviours. The non-functional requirements include usability, performance, reliability and scalability. There are two non-functional requirements being tested for this system which are performance and usability.

7.2.1 Performance Testing

Performance testing is a testing practice to evaluate the performance of a system in the aspect of stability and responsiveness under certain level of workload. In this information system, performance, progressive web application, accessibility and SEO has been audited using Lighthouse. Lighthouse is an automated tool used for improving web page quality. The figure below shows the audited result. The result for application performance is not satisfied due to the speed and time issue. The result for accessibility, best practices and SEO is quite satisfying. Lastly, the audited result below shows that this is a Progress Web App. Please refer to “Appendix H: Performance Testing Report (Lighthouse)” for the full report.



https://pwa-app-36fe6.firebaseapp.com/
Mar 30, 2019, 7:59 PM GMT+8
No emulation, Simulated Slow 4G network

93 Performance
74 Accessibility
86 Best Practices
80 SEO
PWA Progressive Web App

Score scale: ● 90-100 ● 50-89 ● 0-49

Figure 7.10: Audited result using Lighthouse

7.2.2 User Acceptance Testing

User Acceptance Testing is the last level of every software testing phase. UAT will inform real users to test and evaluate the system to determine whether the system behaviour is compliance to the real-world scenarios. In this information system, UAT has been conducted by 10 users which included current society committees and normal students who have not been society committees before. The final score of UAT for every question are around average. For Q1, which is the importance of this information system towards user scores the highest. This shows that this information is preferable by students. The table below shows the score accumulated from all user feedback forms.

Table 7.1: Usability Test Average Score

Questions	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
User 1	4	4	3	4	3	1	3	3	4	4
User 2	3	2	3	4	3	1	3	2	3	1
User 3	3	3	3	2	3	3	3	1	3	3
User 4	4	2	4	1	4	2	4	1	3	2
User 5	4	3	3	2	3	2	3	3	3	1
User 6	5	2	4	3	4	1	4	1	4	1
User 7	5	4	2	2	3	5	2	4	3	3
User 8	4	3	3	2	3	2	4	3	4	2
User 9	3	1	3	1	4	3	4	3	3	2
User 10	4	3	3	3	3	2	3	2	3	2
Total	39	27	31	24	33	22	33	23	33	21
Average	3.9	2.7	3.1	2.4	3.3	2.2	3.3	2.3	3.3	2.1

CHAPTER 8

CONCLUSION AND DISCUSSIONS

This chapter summarises how UTAR Societies and Events' Information System solves the issues and achieves the project objectives. Limitations and recommendations of this information are also described as shown below.

9.1 Conclusion

After 6 months of Software Development Life Cycle of planning, analysis, design, development and testing, a cocu-curriculum information system is delivered successfully at the end of this project. This system has fulfilled the initial planned objectives which include:

1. To plan the methodology to be used to conduct this project.
2. To design a final solution of this project.
3. To implement the project according to the final solution.
4. To test and evaluate the effectiveness and efficiency of the completed solution.

9.2 Limitations

Despite achievement all of the initial planned objectives in this project, there are still some limitations in this information system due to time and scope constraints. The list below shows the limitations in this information system:

1. This project is not able to get students and staffs' information from the shared database in UTAR.
2. This project cannot be accessible in mobile phone due to the implementation of websocket is unsecure using http or ws. The RESTful API in this system is secure due to implementation in Amazon API Gateway. However, websocket which is a TCP connection is not supported in Amazon API Gateway. This project is not using secure websocket due to the limitation in applying SSL certificate for secure communication which require a domain.

9.3 Recommendations for future improvement

A future improvement for this project has been planned. The additional features shown below are not implemented in current situation, but it will be very useful to be added into this project in the future.

1. Payment feature which allows users to pay for society or event registration fee.
2. Email notification will be sent to users as a reminder about actual date of participated events.
3. Chatbot feature for students to communicate with society committees of staffs if there is any enquiry.
4. Submission of event proposal by student and management (either approve, reject or deferred) of event proposal by staff.
5. The Machine Learning algorithm used for recommender system can be further improved to increase the accuracy of event recommendation. For example, parameter tuning can be done on the existing algorithm. Besides, new algorithm can be introduced instead of using Alternating Least Square (ALS) model from Spark ML.

REFERENCES

Ashwini, A., 2017. *Should you use NoSQL or SQL Db or both?* [online] Available at: <https://medium.com/swlh/should-you-use-nosql-or-sql-db-or-both-349cb26c9add> [Accessed 21 June 2018].

Amazon Web Services, n.d. Amazon Web Services. [online] Available at: <https://aws.amazon.com/> [Accessed 15 January 2019].

Apache Spark, n.d. Machine Learning Library (MLlib) Guide. [online] Available at: <https://aws.amazon.com/s3/> [Accessed 15 January 2019].

Buna, S., 2017. *All the Fundamental React.js concepts.* [online] Available at: <https://medium.freecodecamp.org/all-the-fundamental-react-js-concepts-jammed-into-this-single-medium-article-c83f9b53eac2> [Accessed 03 July 2018].

C# Corner, n.d. What is ReactJS and Why should we use it? [online] Available at: <https://www.c-sharpcorner.com/article/what-and-why-reactjs/> [Accessed 15 January 2019].

CloudAcademy, n.d. *Introduction to the Google Cloud Platform.* [online] Available at: <https://cloudacademy.com/course/introduction-to-the-google-cloud-platform/> [Accessed 3 July 2018].

Duncan, S., 2017. *Angular vs React - A Side-By-Side Comparison.* [online] Available at: <https://www.pluralsight.com/guides/angular-vs-react-a-side-by-side-comparison> [Accessed 21 June 2018].

Firebase, n.d. Firebase. [online] Available at: <https://firebase.google.com/docs/> [Accessed 16 January 2019].

Firebase, 2018. *Cloud Firestore.* [online] Available at: <https://firebase.google.com/docs/firestore/> [Accessed 28 June 2018].

Google Developers, n.d. Progressive Web Apps. [online] Available at: <https://developers.google.com/web/progressive-web-apps/> [Accessed 15 January 2019].

Isinkaye, F.O., Folajimi, Y.O. and Ojokoh, B.A., 2015. Recommendation System: Principles, methods, and evaluation. *Egyptian Informatics Journal*. [online] Available at: <https://www.sciencedirect.com/science/article/pii/S1110866515000341> [Accessed 26 July 2018].

ISTQB Exam Certification, n.d. *What is Agile model - advantages, disadvantages and when to use it?* [online] Available at: <http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/> [Accessed 20 June 2018].

ISTQB Exam Certification, n.d. *What is Waterfall model - advantages, disadvantages and when to use it?* [online] Available at: <http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/> [Accessed 20 June 2018].

Jamsheer, K., n.d.*12 Best Software Development Methodologies with Pros and Cons*. [online] Available at: <https://acodez.in/12-best-software-development-methodologies-pros-cons/> [Accessed 20 June 2018].

Little, G., 2018. *Benefits of using Online Management System*. [online] Available at: <http://www.clixsounds.com/benefits-of-using-online-management-system/> [Accessed 24 June 2018].

McLeod, S., 2018. *Questionnaire.* [online] Available at: <https://www.simplypsychology.org/questionnaires.html> [Accessed 02 July 2018].

Michael, J.P., & Daniel, B., n.d. *Content-based Recommendation Systems*. California: FXPAL.

Polepeddi, L., 2013. *An Introduction to Python's Flask Framework.* [online] Available at: <https://code.tutsplus.com/tutorials/an-introduction-to-pythons-flask-framework--net-28822> [Accessed 3 July 2018].

Ratnmala, R. and Haresh, M., 2013. Comparative Study of Various Process Model in Software Development. *International Journal of Computer Applications*. [online] Available at: <https://www.researchgate.net/profile/Haresh_Rathod/publication/260632268_Comparative_Study_of_Various_Process_Model_in_Software_Development/links/54f57ca00cf2eed5d737a9c0/Comparative-Study-of-Various-Process-Model-in-Software-Development
.pdf> [Accessed 25 July 2018].

Real Python, n.d. Flask Tutorials. [online] Available at: <https://realpython.com/tutorials/flask/> [Accessed 16 January 2019].

Rouse, M., 2005. *Prototyping Model*. [online] Available at: <https://searchcio.techtarget.com/definition/Prototyping-Model> [Accessed 24 June 2018].

Sharma, L., 2016. *What is Waterfall Model*. [online] Available at: <http://toolsqa.com/software-testing/waterfall-model/> [Accessed 20 June 2018].

Thorpe, S., 2018. *Cloud Services Comparison: AWS Vs. Google Vs. Azure*. [online] Available at: <https://caylent.com/aws-google-azure-cloud-comparison/#Comp> [Accessed 21 June 2018].

Tutorials Point, n.d. *SDLC - RAD Model.* [online] Available at: <https://www.tutorialspoint.com/sdlc/sdlc_rad_model.htm> [Accessed 20 June 2018].

Tutorials Point, n.d. *ReactJS - JSX.* [online] Available at: <https://www.tutorialspoint.com/reactjs/reactjs_jsx.htm> [Accessed 02 July 2018].

W3Schools, n.d. *SQL Tutorial.* [online] Available at: <https://www.w3schools.com/sql/> [Accessed 2 July 2018].
W3Schools, n.d. *Bootstrap 3 Tutorial.* [online] Available at: <https://www.w3schools.com/booTsTrap/default.asp> [Accessed 03 July 2018].

Weebly, .n.d. *The Computer's Impact on Society*. [online] Available at: <http://thecomputersimpact.weebly.com/the-positive-and-negative-effects-of-the-internet.html> [Accessed 24 June 2018].

Willoughby, J., 2017. *5 Benefits of React.js to Brighten a Cloudy Day.* [online] Available at: <https://dzone.com/articles/5-benefits-of-reactjs-to-brighten-a-cloudy-day> [Accessed 28 June 2018].

Xiaoyuan, S. and Taghi, M.K., 2009. A Survey of Collaborative Filtering Techniques. *Advances In Artificial Intelligence.* [online] Available at: <https://www.hindawi.com/journals/aai/2009/421425/> [Accessed 26 July 2018].

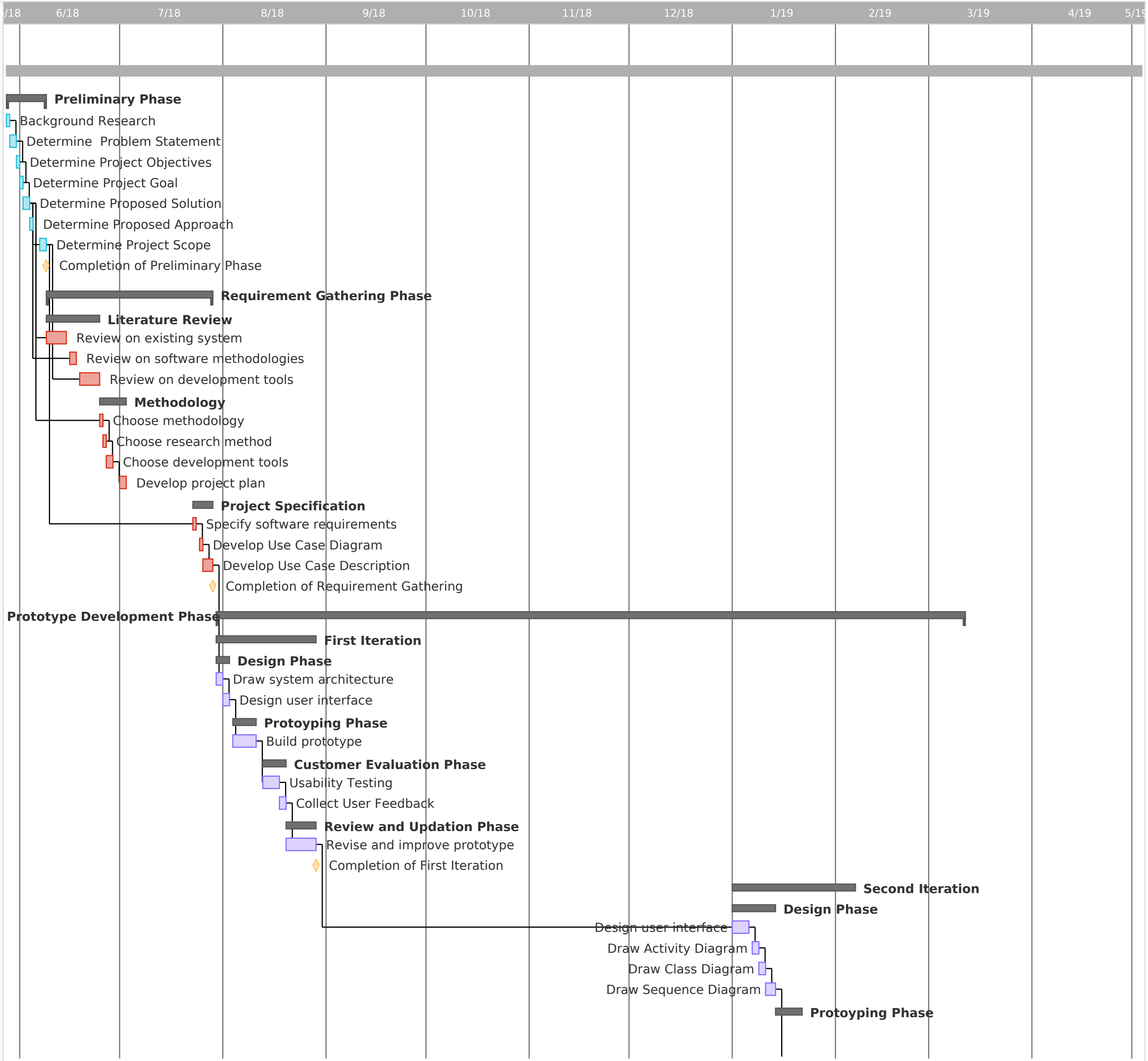
APPENDICES

Appendix A: Work Breakdown Structure and Gantt chart

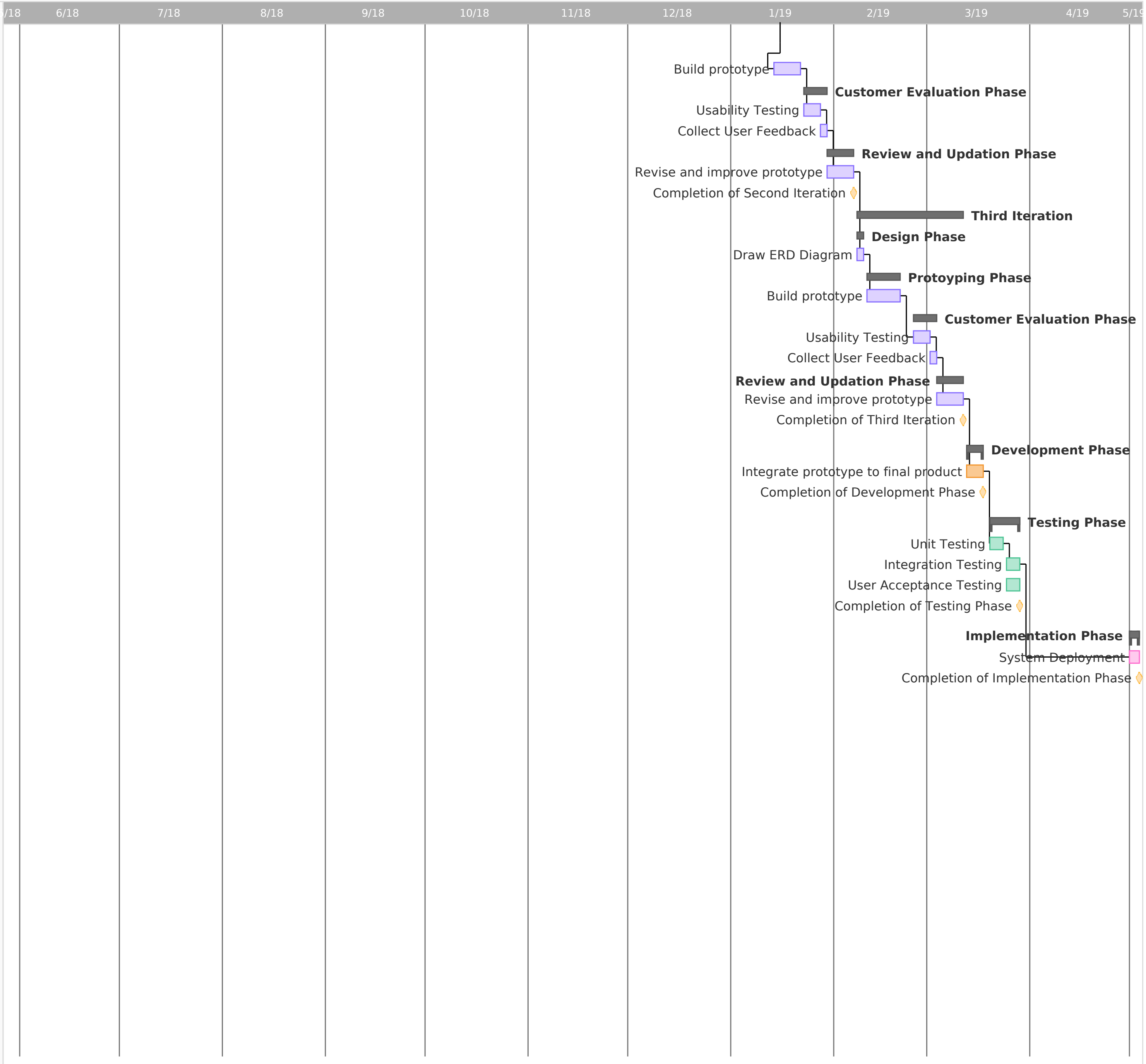
Task Name	Duration	Start	Finish
Preliminary Phase	**12 days**	**Mon 28/5/18**	**Fri 8/6/18**
Background Research	1 day	Mon 28/5/18	Mon 28/5/18
Determine Problem Statement	2 days	Tues 29/5/18	Wed 30/5/18
Determine Project Objectives	1 day	Thurs 31/5/18	Thurs 31/5/18
Determine Project Goal	1 day	Fri 1/6/18	Fri 1/6/18
Determine Proposed Solution	2 days	Sat 2/6/18	Sun 3/6/18
Determine Proposed Approach	1 day	Mon 4/6/18	Mon 4/6/18
Determine Project Scope	2 days	Thurs 7/6/18	Fri 8/6/18
Requirement Gathering Phase	**60 days**	**Sat 9/6/18**	**Sun 29/7/18**
Literature Review	**15 days**	**Sat 9/6/18**	**Fri 24/6/18**
Review on existing system	6 days	Sat 9/6/18	Thurs 14/6/18
Review on software methodologies	2 days	Sat 16/6/18	Sun 17/6/18
Review on development tools	6 days	Tues 19/6/18	Sun 24/6/18
Methodology	**6 days**	**Mon 25/6/18**	**Mon 2/7/18**
Choose methodology	1 day	Mon 25/6/18	Mon 25/6/18
Choose research method	1 day	Tues 26/6/18	Tues 26/6/18
Choose development tools	2 days	Wed 27/6/18	Thurs 28/6/18
Develop project plan	2 days	Sun 1/7/18	Mon 2/7/18
Project Specification	**5 days**	**Mon 23/7/18**	**Sat 28/7/18**
Specify software requirements	1 days	Mon 23/7/18	Mon 23/7/18
Develop Use Case Diagram	1 days	Wed 25/7/18	Wed 25/7/18
Develop Use Case Description	3 days	Thurs 26/7/18	Sat 28/7/18
Prototype Development Phase	**216 days**	**Mon 30/7/18**	**Thurs 28/2/19**
First Iteration	**25 days**	**Mon 30/7/18**	**Sun 26/8/18**
Design Phase	**4 days**	**Mon 30/7/18**	**Thurs 2/8/18**
Draw system architecture	2 days	Mon 30/7/18	Tues 31/7/18
Design user interface	2 days	Wed 1/8/18	Thurs 2/8/18
Protoyping Phase	**7 days**	**Sat 4/8/18**	**Fri 10/8/18**
Build prototype	7 days	Sat 4/8/18	Fri 10/8/18
Customer Evaluation Phase	**7 days**	**Mon 13/8/18**	**Sun 19/8/18**
Usability Testing	5 days	Mon 13/8/18	Fri 17/8/18
Collect User Feedback	2 days	Sat 18/8/18	Sun 19/8/18
Review and Updation Phase	**7 days**	**Mon 20/8/18**	**Sun 26/8/18**
Revise and improve prototype	7 days	Mon 20/8/18	Sun 26/8/18
Second Iteration	**39 days**	**Tues 1/1/19**	**Thurs 31/1/19**
Design Phase	**14 days**	**Tues 1/1/19**	**Tues 15/1/19**
Design user interface	5 days	Tues 1/1/19	Sat 5/1/19
Draw Activity Diagram	2 days	Mon 7/1/19	Tues 8/1/19
Draw Class Diagram	2 days	Wed 9/1/19	Thurs 10/1/19
Draw Sequence Diagram	3 days	Fri 11/1/19	Sun 13/1/19
Protoyping Phase	**8 days**	**Mon 14/1/19**	**Sat 21/1/19**
Build prototype	8 days	Mon 14/1/19	Sat 21/1/19

Customer Evaluation Phase	**7 days**	**Wed 23/1/19**	**Tues 29/1/19**
Usability Testing	5 days	Wed 23/1/19	Sun 27/1/19
Collect User Feedback	2 days	Mon 28/1/19	Tues 29/1/19
Review and Updation Phase	**8 days**	**Wed 30/1/19**	**Sun 6/2/19**
Revise and improve prototype	8 days	Wed 30/1/19	Sun 6/2/19
Third Iteration	**27 days**	**Fri 8/2/19**	**Wed 27/2/19**
Design Phase	**2 days**	**Fri 8/2/19**	**Sat 9/2/19**
Draw ERD Diagram	2 days	Fri 8/2/19	Sat 9/2/19
Protoyping Phase	**10 days**	**Mon 11/2/19**	**Wed 20/2/19**
Build prototype	10 days	Mon 11/2/19	Wed 20/2/19
Customer Evaluation Phase	**7 days**	**Mon 25/2/19**	**Sun 3/3/19**
Usability Testing	5 days	Mon 25/2/19	Fri 1/3/19
Collect User Feedback	2 days	Sat 2/3/19	Sun 3/3/19
Review and Updation Phase	**8 days**	**Mon 4/3/19**	**Mon 11/3/19**
Revise and improve prototype	8 days	Mon 4/3/19	Mon 11/3/19
Development Phase	**5 days**	**Wed 13/3/19**	**Sun 17/3/19**
Integrate prototype to final product	5 days	Wed 13/3/19	Sun 17/3/19
Testing Phase	**13 days**	**Wed 20/3/19**	**Fri 5/4/19**
Unit Testing	4 days	Wed 20/3/19	Sat 23/3/19
Integration Testing	4 days	Mon 25/3/19	Thurs 28/3/19
User Acceptance Testing	5 days	Mon 1/4/19	Fri 5/4/19
Implementation Phase	**3 days**	**Wed 1/5/19**	**Fri 3/5/19**
System Deployment	3 days	Wed 1/5/19	Fri 3/5/19

project1	start	end
Preliminary Phase	**28/05/18**	**08/06/18**
Background Research	28/05	28/05
Determine Problem Statement	29/05	30/05
Determine Project Objectives	31/05	31/05
Determine Project Goal	01/06	01/06
Determine Proposed Solution	02/06	03/06
Determine Proposed Approach	04/06	04/06
Determine Project Scope	07/06	08/06
Completion of Preliminary Phase	08/06	08/06
Requirement Gathering Phase	**09/06/18**	**28/07/18**
Literature Review	**09/06/18**	**24/06/18**
Review on existing system	09/06	14/06
Review on software methodologies	16/06	17/06
Review on development tools	19/06	24/06
Methodology	**25/06/18**	**02/07/18**
Choose methodology	25/06	25/06
Choose research method	26/06	26/06
Choose development tools	27/06	28/06
Develop project plan	01/07	02/07
Project Specification	**23/07/18**	**28/07/18**
Specify software requirements	23/07	23/07
Develop Use Case Diagram	25/07	25/07
Develop Use Case Description	26/07	28/07
Completion of Requirement Gathering	28/07	28/07
Prototype Development Phase	**30/07/18**	**11/03/19**
First Iteration	**30/07/18**	**28/08/18**
Design Phase	**30/07/18**	**02/08/18**
Draw system architecture	30/07	31/07
Design user interface	01/08	02/08
Protoyping Phase	**04/08/18**	**10/08/18**
Build prototype	04/08	10/08
Customer Evaluation Phase	**13/08/18**	**19/08/18**
Usability Testing	13/08	17/08
Collect User Feedback	18/08	19/08
Review and Updation Phase	**20/08/18**	**28/08/18**
Revise and improve prototype	20/08	28/08
Completion of First Iteration	28/08	28/08
Second Iteration	**01/01/19**	**06/02/19**
Design Phase	**01/01/19**	**13/01/19**
Design user interface	01/01	05/01
Draw Activity Diagram	07/01	08/01
Draw Class Diagram	09/01	10/01
Draw Sequence Diagram	11/01	13/01
Protoyping Phase	**14/01/19**	**21/01/19**

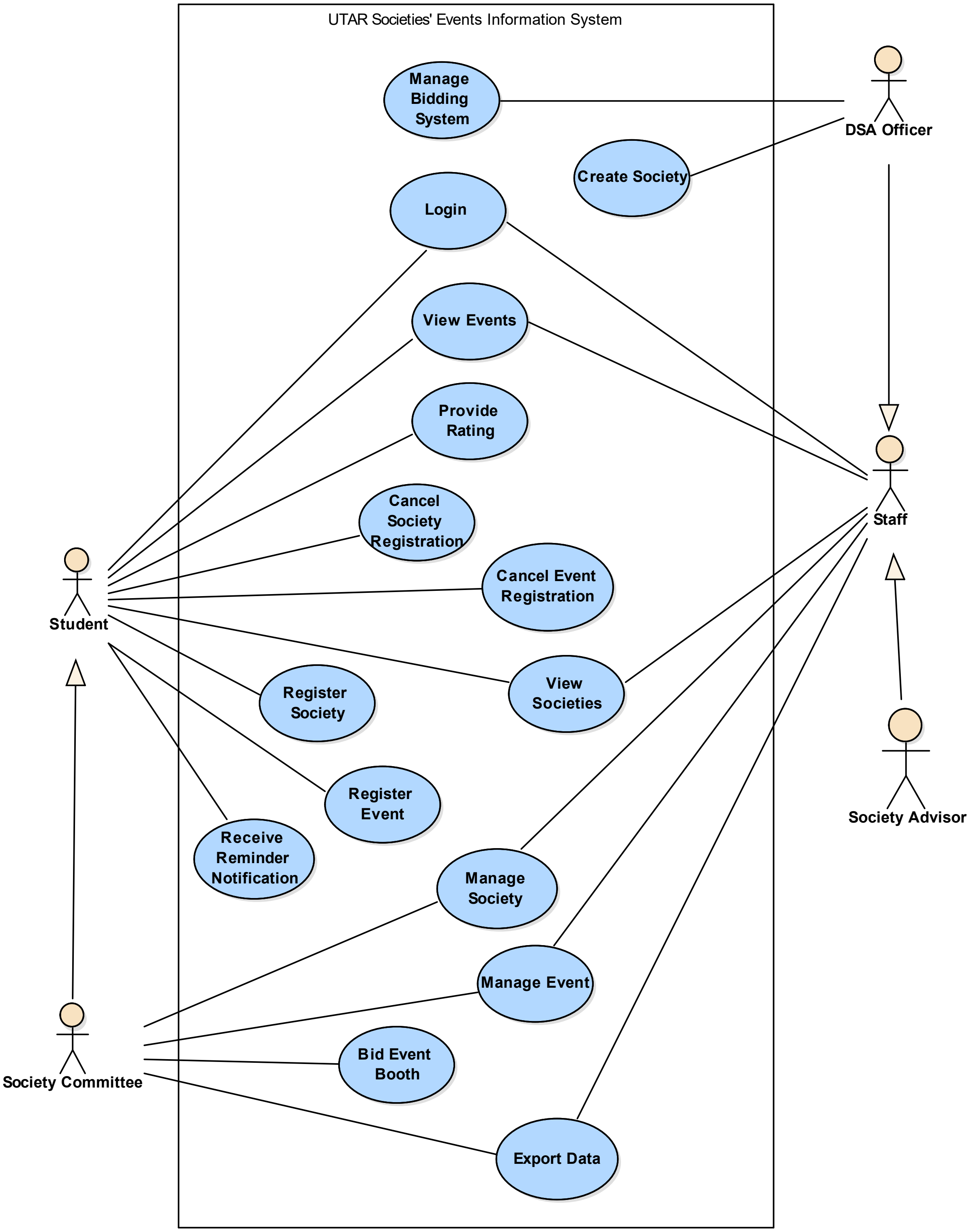


Build prototype	14/01	21/01
Customer Evaluation Phase	**23/01/19**	**29/01/19**
Usability Testing	23/01	27/01
Collect User Feedback	28/01	29/01
Review and Updation Phase	**30/01/19**	**06/02/19**
Revise and improve prototype	30/01	06/02
Completion of Second Iteration	06/02	06/02
Third Iteration	**08/02/19**	**11/03/19**
Design Phase	**08/02/19**	**09/02/19**
Draw ERD Diagram	08/02	09/02
Protoyping Phase	**11/02/19**	**20/02/19**
Build prototype	11/02	20/02
Customer Evaluation Phase	**25/02/19**	**03/03/19**
Usability Testing	25/02	01/03
Collect User Feedback	02/03	03/03
Review and Updation Phase	**04/03/19**	**11/03/19**
Revise and improve prototype	04/03	11/03
Completion of Third Iteration	11/03	11/03
Development Phase	**13/03/19**	**17/03/19**
Integrate prototype to final product	13/03	17/03
Completion of Development Phase	17/03	17/03
Testing Phase	**20/03/19**	**28/03/19**
Unit Testing	20/03	23/03
Integration Testing	25/03	28/03
User Acceptance Testing	25/03	28/03
Completion of Testing Phase	28/03	28/03
Implementation Phase	**01/05/19**	**03/05/19**
System Deployment	01/05	03/05
Completion of Implementation Phase	03/05	03/05



APPENDICES

Appendix B: Use Case Diagram


UTAR Societies' Events Information System
Manage Bidding System
DSA Officer
Create Society
Login
View Events
Provide Rating
Staff
Cancel Society Registration
Cancel Event Registration
Student
View Societies
Register Society
Society Advisor
Register Event
Receive Reminder Notification
Manage Society
Manage Event
Society Committee
Bid Event Booth
Export Data

APPENDICES

Appendix C: Feedback Survey Form

Co-curriculum Portal Survey Form

Good day and thank you for agreeing to take part in this survey. I am Wan Ching, Software Engineering student from University Tunku Abdul Rahman (Sungai Long campus). I am currently working on my Final Year Project which is developing a UTAR Societies Events Information System. This system is mainly focus on the management and information of all societies and events such as event registration, member registration and etc. Today, I would like to conduct this survey with you to obtain more suggestions upon your preference if a new UTAR co-curriculum portal will be developed. This survey will only take less than 10 minutes. For your information, the answer you provided will be taken into consideration for my project and the information you provided might be included in my project report.
Your help is much appreciated and kindly try your best to complete this survey form. Thank you!

* Required

1. **Name** *

____

2. **Course (Eg: Software Engineering)** *

____

3. **Year & Trimester (Eg: Y1S1)**

____

4. **Gender**

Mark only one oval.

○ Male

○ Female

5. **Are you active in participating co-curriculum activities in UTAR? Kindly share the reason (for Yes or No).**

____

____

____

____

____

6. **How do you usually acknowledge about the various kinds of UTAR co-curricular activities?**

Check all that apply.

□ Facebook

□ Whatsapp

□ Friends

□ Booths at campus

□ Other: ______________________________

7. **Do you feel that it is inconvenient to acknowledge about scattered information of co-curricular activities through social media?**
Mark only one oval.

○ Yes
○ No

8. **Regarding your answer for the previous question, kindly provide a simple reason (for Yes or No).**

____
____
____
____
____

9. **Do you have the experience of forgetting to attend the events that you have registered or you remember it on one day before the actual date of the event?**
Mark only one oval.

○ Yes
○ No

10. **How many times do you encountered the above experience?**
Mark only one oval.

○ 1 - 3 times
○ 4 - 6 times
○ more than 6 times
○ Never
○ Other: ____

11. **If a notification about the event that you have registered will be sent to your email and SMS text message, will this helps you to solve the problem above?**
Mark only one oval.

○ Yes
○ No
○ Maybe

12. **Which way of notification do you prefer?**
Check all that apply.

□ SMS text message
□ Email
□ Other: ____

13. **Since UTAR portal is existing now for academic purpose, do you feel that a co-curriculum portal is also necessary for co-curriculum purpose?**
Mark only one oval.

○ Yes
○ No

14. **If there is a UTAR co-curriculum portal which is used to manage co-curricular stuff, which features do you wish to add into the system for your convenience?** *

* Kindly provide other features that are upon your preferences.
Check all that apply.

□ Students are able to register for their interested societies

□ Students are able to register for their interested events

□ Students are able to make payment for society / event registration fee

□ Students are able to rate the events that they had participated

□ Society committees are able to manage their own societies (such as viewing member name list, manage new member registration)

□ Staffs are able to manage their respective society (such as society advisor is able to see all information about the respective society)

□ Other: ____________________

15. **If there is a rating system for every event, are you willing to rate for every event that you had participated through is this co-curriculum portal?**

Mark only one oval.

○ Yes
○ No

16. **Is this kind of co-curriculum portal able to increase your involvement in UTAR co-curricular activities due to the high convenience and efficiency?**

Mark only one oval.

○ Yes
○ No

17. **Do you have any other expectation regarding this UTAR co-curriculum portal?**

____
____
____
____
____

Powered by Google Forms

Co-curriculum Portal Survey Form on University / College

Thank you for agreeing to take part in this important survey. I am Wan Ching, Software Engineering student from University Tunku Abdul Rahman (Sungai Long campus). I am currently working on my Final Year Project which is developing a UTAR Societies Events Information System.
This system is mainly focus on the management of all societies and events such as event registration, member registration and etc. Today, I would like to conduct this survey with you to obtain more information about the existing and similar system in your University or College.
This survey will only take less than 10 minutes. For your information, the answer you provided will be taken into consideration for my project and the information you provided might be included in my project report.
Your help is much appreciated and kindly try your best to complete this survey form. Thank you!

* Required

1. **Name** *

   ____

2. **University / College** *

   ____

3. **Gender**

   Mark only one oval.

   ○ Male

   ○ Female

4. **Is it students are required to login into the current existing co-curriculum portal in your University / College?**

   Mark only one oval.

   ○ Yes

   ○ No

5. **What are the features provided by the co-curriculum portal?** *

   Check all that apply.

   □ Students are able to register for their interested societies

   □ Students are able to register for their interested events

   □ Students are able to make payment for society / event registration fee

   □ Students are able to rate the events that they had participated

   □ Society committees are able to manage their own societies (such as viewing member name list, manage new member registration)

   □ Staffs are able to manage their respective society (such as society advisor is able to see all information about the respective society)

   □ Other: ____________________________

6. **Any other features (which are not stated above) that are provided in the co-curriculum portal?**

7. **What are the other features that you wish to suggest to add into the current existing co-curriculum portal in your University / College upon your personal convenience?**

8. **Is this existing co-curriculum portal able to increase your involvement in co-curricular activities due to the high convenience?**

Mark only one oval.

○ Yes

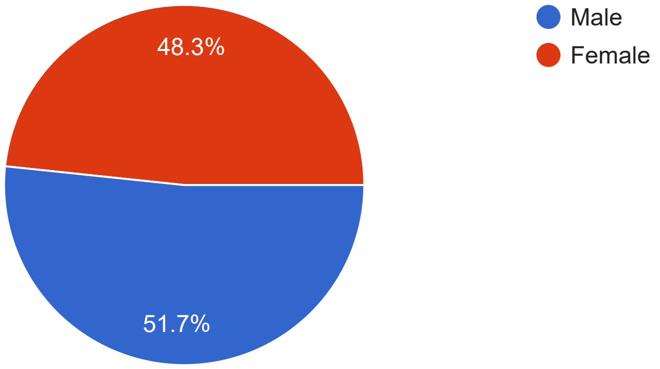
○ No

Powered by
Google Forms

APPENDICES

Appendix D: Survey Form Result

60 responses



Are you active in participating co-curriculum activities in UTAR? Kindly share the reason (for Yes or No).

60 responses

No (20)

Yes (13)

yes (2)

NO (2)

Yes (2)

yes, because of my frds

Yes, just to have fun with friends and gain some social knowledge

Yes.

Before yes now no, I think enough liao

No, I am busy with studies.

Yes. Because I personally found a group of people with same goal

Yes, I would like to extend my network to new people, and also can develop some of the skills that will be useful for me in the future.

yes, because my frds ask me to join

Yes, I can only join this kind of co-cu in university, so why not?

Yes, it is very fun.

Yee , to learn more soft skills abilities

Yes, Interesting.

No.

Sometimes will be very active

Partially 😂 Not every event I would be interested.

No. No time

YES

No

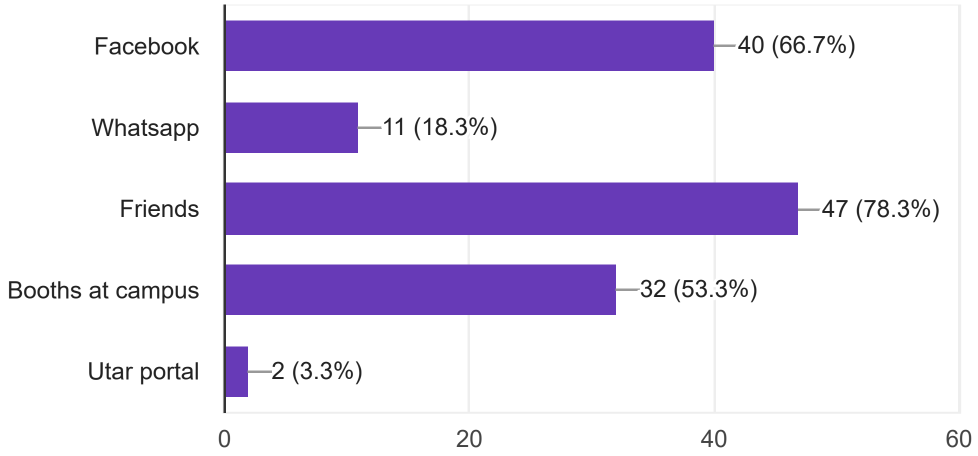
No due to tight schedule

No. Seldom get information for the activities.

No, no time

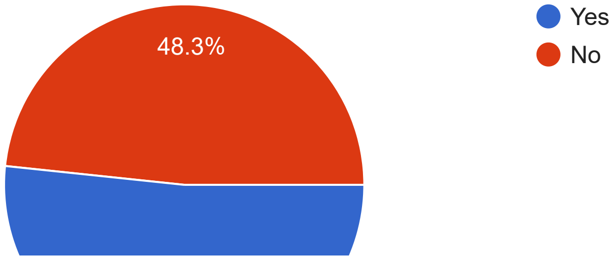
How do you usually acknowledge about the various kinds of UTAR co-curricular activities?

60 responses



Do you feel that it is inconvenient to acknowledge about scattered information of co-curricular activities through social media?

60 responses



Regarding your answer for the previous question, kindly provide a simple reason (for Yes or No).

44 responses

Because everyone got at least one social media app,it makes admin to spread the update to everyone

Bcoz it's convenient

more ez to know

No

sometimes i cannot find the info of new events

Coz it's convenient

It is time consuming as we have to check every social media platform

Without clear information

Sometimes too many messages, too annoying

Everyone gt social media app easier for admin to contact them

It is like a form of advertisement

Information is not complete

Because there will be more chamce to choose the event that you like to join

Haha

I think that now social media is enough for the students to discover the co-curricular activities. If the students are really interested to join the activity, they can always approach the PIC in the information online.

Hard to find a specific information especially when you dont remember the full name of the event.

so that they know got this kind of event and go search it

Promoting during class, for some big event

People now are more active in social media

Easily flooded on Facebook when group of committees start to share the same post can know further information

Social media is a very good platform to let us know what events are going on currently.

Only my friends event will appears on my facebook

Better for us to know the latest information.

-

It's rather hard to get some official information about certain event as we do not have a designated platform to publish it.

Sometimes the information that acknowledge through social media is not accurate

Because it easy to access the detail

Unable to get to know all on-going co-curricular activities

All cocu information send to all types of platform, Sometimes it’s confusing and troublesome

Since I use social media a lot, and I would agree that it is more convenient to know the cocu information

some information is missing in the post such as venue & registration method

I think it is quite okay

As expose with variety of event that recruiting member, we can choose the one we like.

The source too scattered like you mentioned.

May miss out information

Because I open whatsapp everday

Due to busy schedule, I rarely go to level 1. I usually acknowlegde some events through facebook and friends. Therefore, having a portal would be better for me to check it out.

Easier

No time and transport

Is better to have a single source for all co-curricular activities for ease of use.

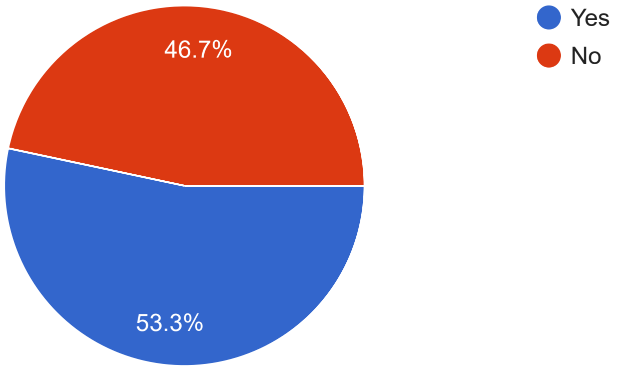
Have to update myself to the social media always

We always using social media

Actually it is quite easy to know all the information through social media

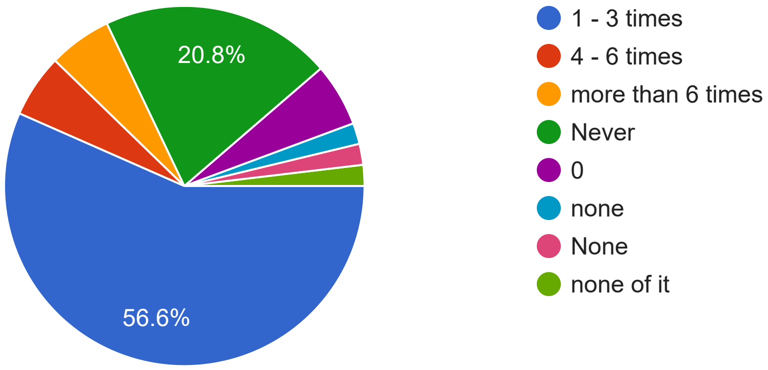
Do you have the experience of forgetting to attend the events that you have registered or you remember it on one day before the actual date of the event?

60 responses



How many times do you encountered the above experience?

53 responses

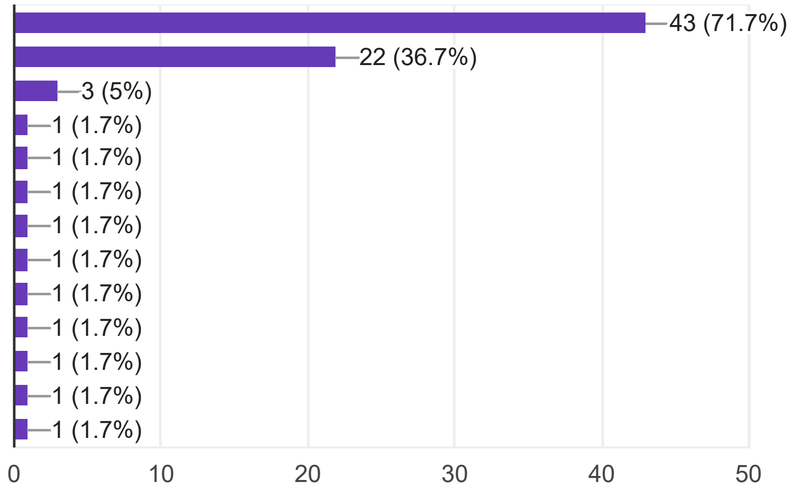


If a notification about the event that you have registered will be sent to your email and SMS text message, will this helps you to solve the problem above?

59 responses

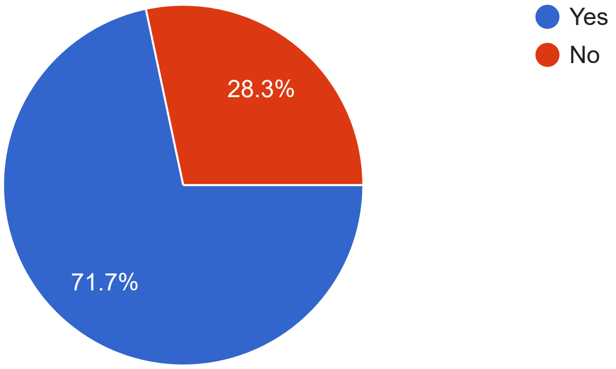
Which way of notification do you prefer?

60 responses



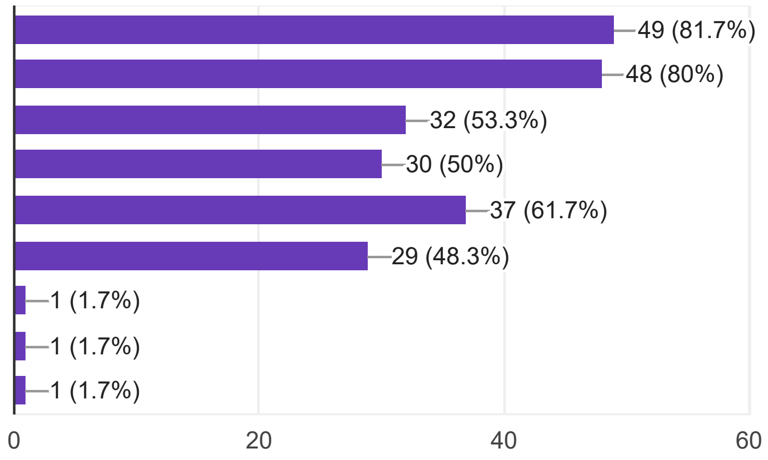
Since UTAR portal is existing now for academic purpose, do you feel that a co-curriculum portal is also necessary for co-curriculum purpose?

60 responses



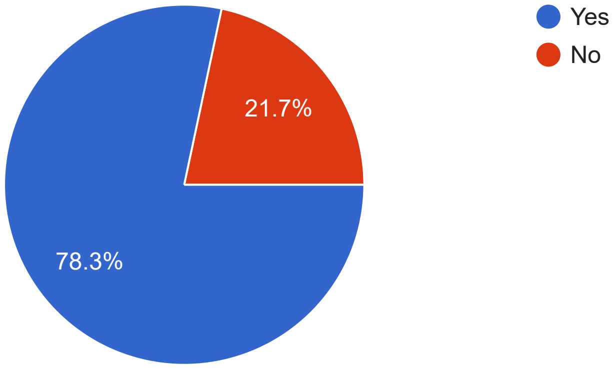
If there is a UTAR co-curriculum portal which is used to manage co-curricular stuff, which features do you wish to add into the system for your convenience?

60 responses



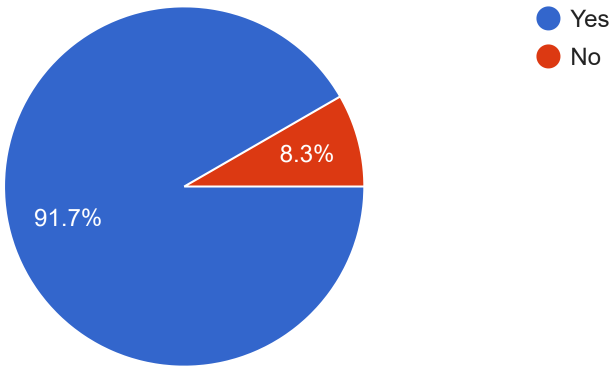
If there is a rating system for every event, are you willing to rate for every event that you had participated through is this co-curriculum portal?

60 responses



Is this kind of co-curriculum portal able to increase your involvement in UTAR co-curricular activities due to the high convenience and efficiency?

60 responses



Do you have any other expectation regarding this UTAR co-curriculum portal?

39 responses

No (16)

- (2)

no

chat function to communicate with club for more information

Able to show any of friend will be participating the event would be nice.

If possible, make a forum at the portal to let utar students to discuss and maybe contribute their ideas

Probably No

no

Email student about any upcoming event register date, since some of student may revisit the site few weeks, but email everyday

Unless this portal can be developed as an app it may be convenient to them, else students may not surf to the portal as it's not compulsory but optional to them

Please make it happen before May 2019!

Mo

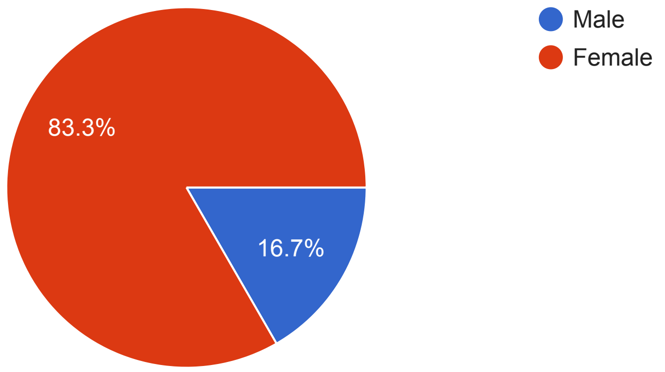
Able to know all the information about the society's activities and events and have a FAQ section

All cocu are organized using just a platform

More user friendly interface will help

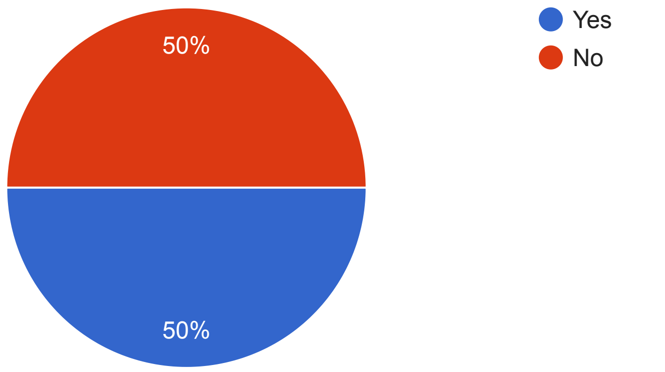
Discounts

6 responses



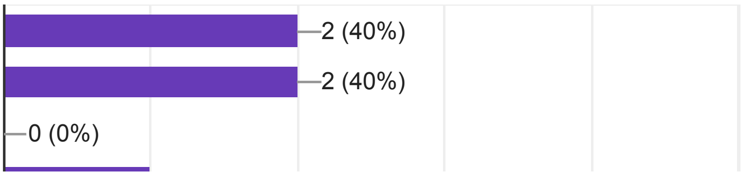
Is it students are required to login into the current existing co-curriculum portal in your University / College?

6 responses



What are the features provided by the existing co-curriculum portal in your university / college?

5 responses



Any other features (which are not stated above) that are provided in the co-curriculum portal?

3 responses

Details of all upcoming events are provided (date, emails of person on charge etc)

General event announcement

Stated in Other section

What are the other features that you wish to suggest to add into the current existing co-curriculum portal in your University / College upon your personal convenience?

3 responses

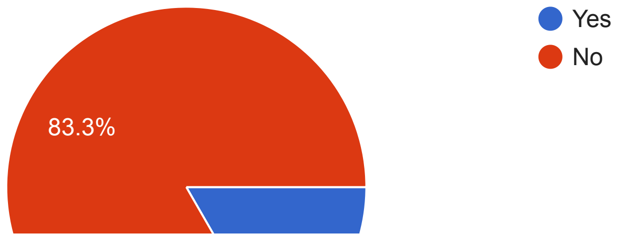
Activities, events, photos etc.

More updates in term of events that are happening in uni

Direct message to ask further inquiry, instead of send email, so can get reply instantly.

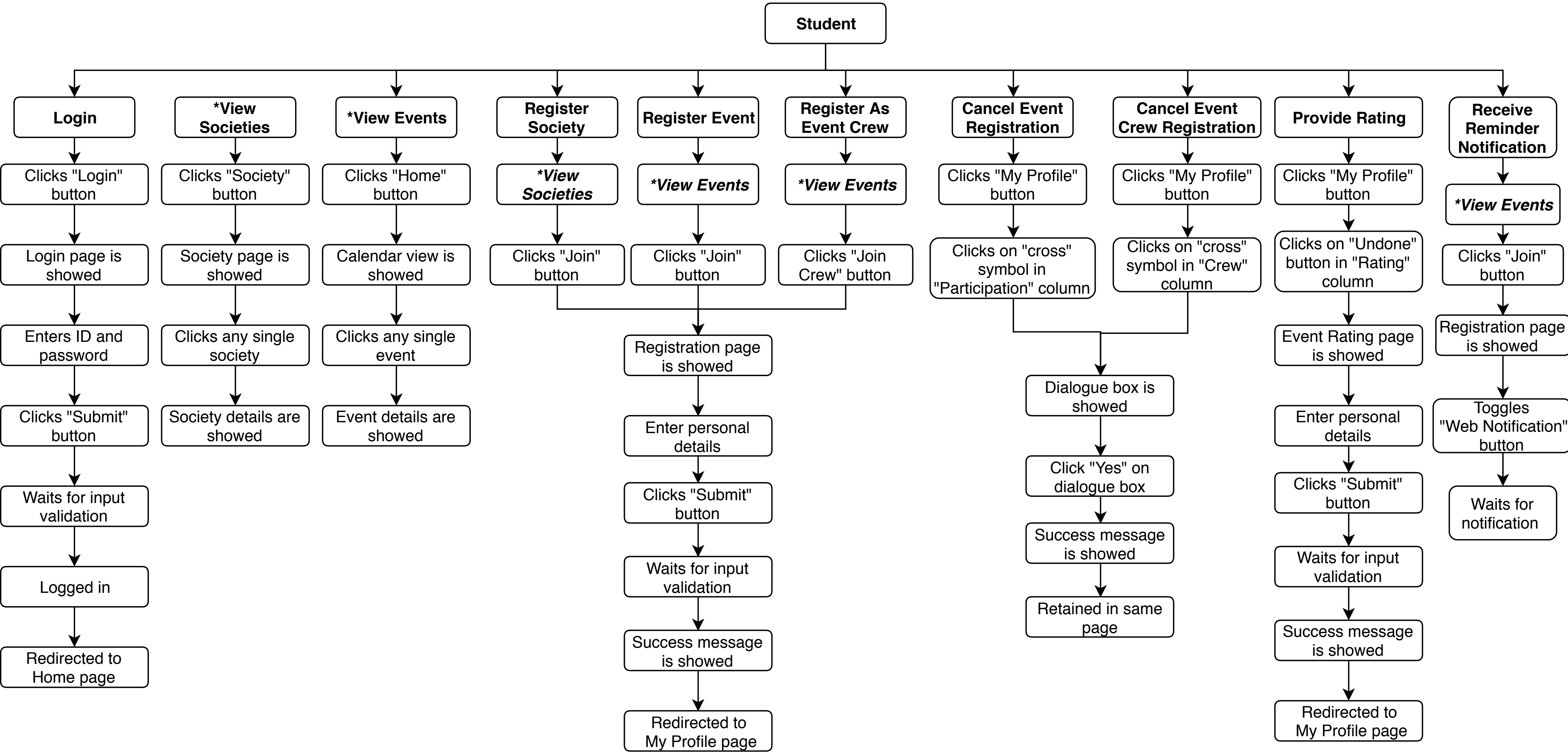
Is this existing co-curriculum portal able to increase your involvement in co-curricular activities due to the high convenience?

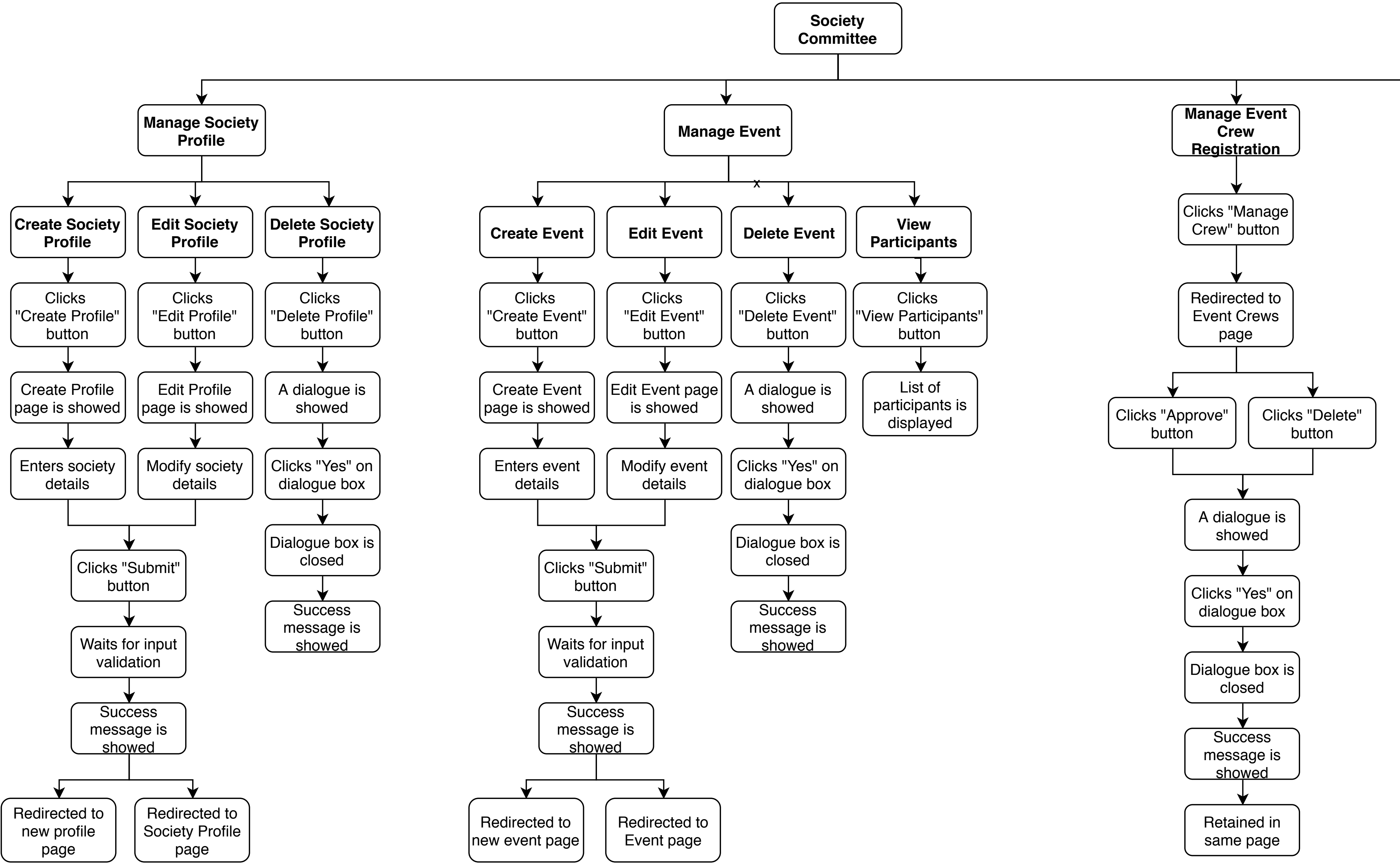
6 responses

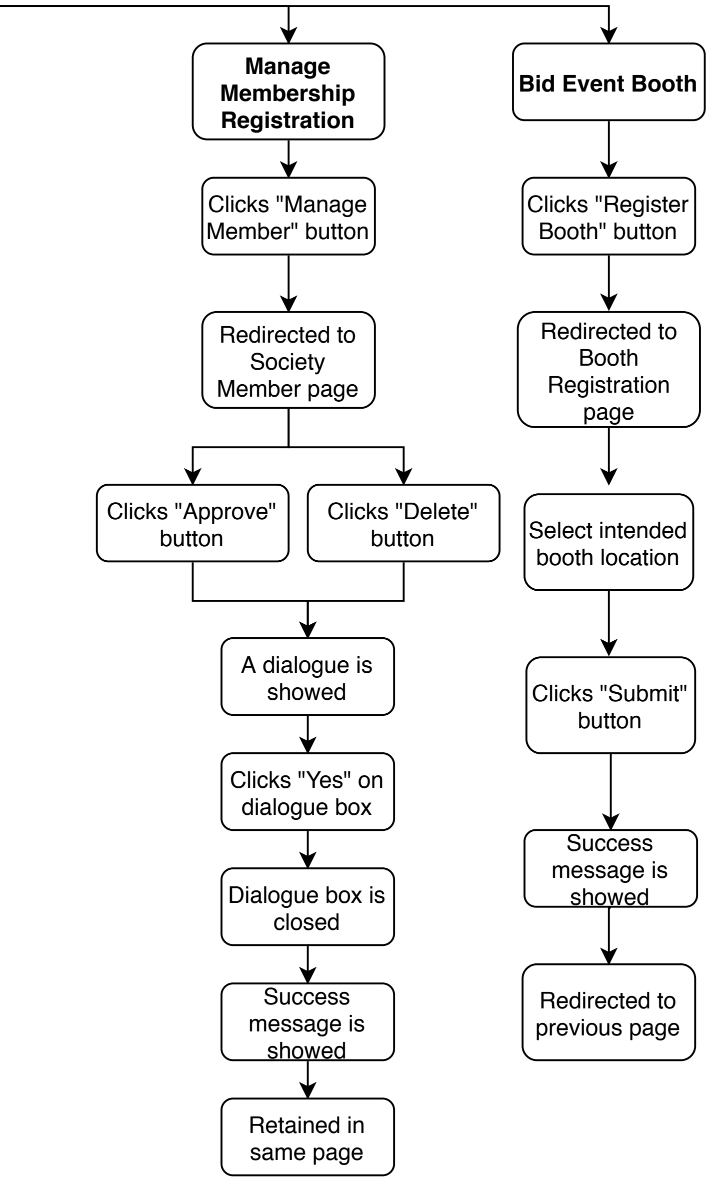

Yes
No
83.3%

APPENDICES

Appendix E: Flow Chart


Student
Login
Clicks "Login" button
Login page is showed
Enters ID and password
Clicks "Submit" button
Waits for input validation
Logged in
Redirected to Home page
*View Societies
Clicks "Society" button
Society page is showed
Clicks any single society
Society details are showed
*View Events
Clicks "Home" button
Calendar view is showed
Clicks any single event
Event details are showed
Register Society
*View Societies
Clicks "Join" button
Register Event
*View Events
Clicks "Join" button
Register As Event Crew
*View Events
Clicks "Join Crew" button
Registration page is showed
Enter personal details
Clicks "Submit" button
Waits for input validation
Success message is showed
Redirected to My Profile page
Cancel Event Registration
Clicks "My Profile" button
Clicks on "cross" symbol in "Participation" column
Cancel Event Crew Registration
Clicks "My Profile" button
Clicks on "cross" symbol in "Crew" column
Dialogue box is showed
Click "Yes" on dialogue box
Success message is showed
Retained in same page
Provide Rating
Clicks "My Profile" button
Clicks on "Undone" button in "Rating" column
Event Rating page is showed
Enter personal details
Clicks "Submit" button
Waits for input validation
Success message is showed
Redirected to My Profile page
Receive Reminder Notification
*View Events
Clicks "Join" button
Registration page is showed
Toggles "Web Notification" button
Waits for notification


Society Committee
Manage Society Profile
Create Society Profile
Clicks "Create Profile" button
Create Profile page is showed
Enters society details
Edit Society Profile
Clicks "Edit Profile" button
Edit Profile page is showed
Modify society details
Clicks "Submit" button
Waits for input validation
Success message is showed
Redirected to new profile page
Redirected to Society Profile page
Delete Society Profile
Clicks "Delete Profile" button
A dialogue is showed
Clicks "Yes" on dialogue box
Dialogue box is closed
Success message is showed
Manage Event
x
Create Event
Clicks "Create Event" button
Create Event page is showed
Enters event details
Edit Event
Clicks "Edit Event" button
Edit Event page is showed
Modify event details
Clicks "Submit" button
Waits for input validation
Success message is showed
Redirected to new event page
Redirected to Event page
Delete Event
Clicks "Delete Event" button
A dialogue is showed
Clicks "Yes" on dialogue box
Dialogue box is closed
Success message is showed
View Participants
Clicks "View Participants" button
List of participants is displayed
Manage Event Crew Registration
Clicks "Manage Crew" button
Redirected to Event Crews page
Clicks "Approve" button
Clicks "Delete" button
A dialogue is showed
Clicks "Yes" on dialogue box
Dialogue box is closed
Success message is showed
Retained in same page


Manage Membership Registration
Clicks "Manage Member" button
Redirected to Society Member page
Clicks "Approve" button
Clicks "Delete" button
A dialogue is showed
Clicks "Yes" on dialogue box
Dialogue box is closed
Success message is showed
Retained in same page
Bid Event Booth
Clicks "Register Booth" button
Redirected to Booth Registration page
Select intended booth location
Clicks "Submit" button
Success message is showed
Redirected to previous page

APPENDICES

Appendix F: Interview Result

INTERVIEW FORM

Name of DSA Officer: Ms Choong Yee Chin

Location: UTAR DSA Office

Objective:
To enquire about UTAR Standard Operating Procedure (SOP) on co-curricular activities.

Feedbacks:

1. Is it possible if all staffs (includes DSA Officer and Advisor of every societies) are given the authorization to manage all societies and events? Example of management activities are editing society or event details, managing society or event participants and managing bidding system.
   No. The authorization should be only given to those DSA officers and advisors who have in-charged on certain societies.

2. Is it necessary that the committee name list of each event to be kept confidential?
   No. The name list should be open for everyone to view and acknowledge.

3. Is it the available positions in every society and event are standardized?
   Yes. Those available positions are standardize and should not have too much difference among all societies and events.

APPENDICES

Appendix G: Integration Test Cases

Test Case	Test Title	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Register Society / Event	- Click on “Societies” link in top navigation bar. - Select interested category of society. - Select interested society. - Click on “Register member” or “Register crew” button. - Fill in all required details and click on “Submit” button.	Student’s credential	- The registered society / event will be shown in My Profile or My Event page with “pending” status.	Same as expected result	Pass
2	Cancel registration for society / event	- Click on “Societies” link in top navigation bar. - Select interested category of society. - Select interested society. - Click on “Register member” or “Register crew” button. - Fill in all required details and click on “Submit” button.	Student’s credential	- The cancelled society / event will be shown in My Profile or My Event page with a cross sign.	Same as expected result	Pass
3	Rate for the participated events	- Click on “My Profile” link in top navigation bar. - Click on “My Event” button. - Select “undone” link on the target event to rate. - Fill in all required details and click on “Submit” button.	Student’s credential	- A dialog box will be shown to notify user that the submission is completed. User is redirect to “My Event” page.	Same as expected result	Pass

4	Create society profile.	- Click on “Create Profile” link in top navigation bar. - Fill in all required details and click on “Submit” button.	Student’s credential Staff’s credential	- The society profile cannot be created because the link is not accessible by unauthorized users.	Same as expected result.	Pass
			DSA officer’s credential	- The society profile can be created successfully due to authorization.	Same as expected result	Pass
5	Create Event	- Click on “My Profile” link in top navigation bar. - Select targeted society to create event. - Click on the floating button at bottom right corner and click on button with “+” sign. - Fill in all required details and click on “Submit” button.	Student’s credential Staff’s credential	- The event cannot be created because the link is not accessible by unauthorized users.	Same as expected result.	Pass
			DSA officer’s credential	- The event can be created successfully due to authorization.	Same as expected result.	Pass
6	Manage available booth for bidding system.	- Click on “Manage” link in top navigation bar. - Click on “Manage Booth” link. - Fill in all required details and click on “Submit” button.	Student’s credential Staff’s credential	- The changes on booth cannot be made because the link is not accessible by unauthorized users.	Same as expected result	Pass
			DSA officer’s credential	- The changes on booth can be made successfully due to authorization.	Same as expected result	Pass

7	Approve / reject registration of society members and crew / event participants and crew	- Click on “My Profile” link in top navigation bar. - Select targeted society to manage members. - Click on the floating button at bottom right corner and click on button with a “human” sign. - Click on “+” sign to approve the targeted member. - Click on “Yes” button to confirm.	Student’s credential Staff’s credential	- The approval on registration cannot be made because the link is not accessible by unauthorized users.	Same as expected result	Pass
			DSA officer’s credential	- The approval on registration can be made due to authorization.	Same as expected result	Pass

APPENDICES

Appendix H: Performance Testing Report (Lighthouse)

https://pwa-app-36fe6.firebaseapp.com/

Mar 30, 2019, 7:59 PM GMT+8

No emulation, Simulated Slow 4G network

93	74	86	80	PWA
Performance	Accessibility	Best Practices	SEO	Progressive Web App

Score scale: ● 90-100 ● 50-89 ● 0-49

Performance

93

Metrics

First Contentful Paint	0.0 s ✓	First Meaningful Paint	3.3 s ⓘ
Speed Index	2.8 s ✓	First CPU Idle	3.5 s ✓
Time to Interactive	3.5 s ✓	Estimated Input Latency	2,160 ms ⚠

Values are estimated and may vary.

Diagnostics

More information about the performance of your application.

1	Ensure text remains visible during webfont load	⚠
2	Minimize main-thread work	4.1 s ⚠
3	Reduce JavaScript execution time	3.1 s ⓘ
4	Minimize Critical Requests Depth	4 chains found

✓ Passed audits

18 audits

Accessibility

74

These checks highlight opportunities to [improve the accessibility of your web app](). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

Color Contrast Is Satisfactory

These are opportunities to improve the legibility of your content.

1 Background and foreground colors do not have a sufficient contrast ratio. ⚠

Elements Use Attributes Correctly

These are opportunities to improve the configuration of your HTML elements.

2 Image elements do not have `[alt]` attributes ⚠

Additional items to manually check 12 audits

✓ **Passed audits** 8 audits

⊖ **Not applicable** 24 audits

Best Practices

86

1 Browser errors were logged to the console ⚠

2 Displays images with incorrect aspect ratio ⚠

✓ **Passed audits** 13 audits

SEO

80

These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. Learn more.

Content Best Practices

Format your HTML in a way that enables crawlers to better understand your app's content.

1 Document does not have a meta description ⚠

Crawling and Indexing

To appear in search results, crawlers need access to your app.

2 robots.txt is not valid 1 error found ⚠

Additional items to manually check 2 audits

✓ **Passed audits** 8 audits

⊖ **Not applicable** 1 audits

Progressive Web App

These checks validate the aspects of a Progressive Web App. [Learn more](#).

Fast and reliable

1 Page load is fast enough on mobile networks ✓

2 Current page does not respond with a 200 when offline ⚠

3 start_url does not respond with a 200 when offline ⚠
Unable to fetch start URL via service worker.

Installable

4 Uses HTTPS ✓

5 Registers a service worker that controls page and start_url ✓

6 Web app manifest meets the installability requirements ✓

PWA Optimized

7 Redirects HTTP traffic to HTTPS ✓

8 Configured for a custom splash screen ✓

9 Sets an address-bar theme color ✓

10 Content is not sized correctly for the viewport ⚠
The viewport size is 931px, whereas the window size is 1338px.

11 Has a `<meta name="viewport">` tag with `width` or `initial-scale` ✓

12 Contains some content when JavaScript is not available ✓

Additional items to manually check 3 audits

Runtime settings

- **URL:** `https://pwa-app-36fe6.firebaseapp.com/`
- **Fetch time:** `Mar 30, 2019, 7:59 PM GMT+8`

- **Device:** `No emulation`
- **Network throttling:** `150 ms TCP RTT, 1,638.4 Kbps throughput (Simulated)`
- **CPU throttling:** `4x slowdown (Simulated)`
- **User agent (host):** `Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36`
- **User agent (network):** `Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.86 Safari/537.36`
- **CPU/Memory Power:** `532`

Generated by **Lighthouse** 4.0.0 | File an issue

APPENDICES

Appendix I: User Acceptance Test Agreement Form

User Acceptance Test Agreement Form

Please read this form before you sign it.

I agree to participate in this user acceptance test and I understand that participation in this user acceptance test is voluntary. I know that

In this user acceptance test:

- I will be asked to perform certain tasks on the web application.
- I will be given the opportunity to ask questions, and will have my questions answered to my satisfaction.
- I will have to evaluate the web application regarding to the tasks that I performed by filling in the survey form.
- My suggestion may be used to help improve the usability of the web application.
- I have the right to withdraw my consent to this user acceptance testing and to discontinue participation at any time, without prejudice to my future treatment.

I have read and understood the information on this form and any questions that I have about this user acceptance testing have been answered. My signature below may be taken as affirmation of all above statements; it was given prior to my participation in this user acceptance test.

User's name : ___________________________

Signature : ___________________________

Date : ___________________________

I appreciate your participation. Thank you!

Project Title:	UTAR Societies Events Information System
Student Name:	TEOH WAN CHING
Supervisor Name:	Ooi Ean Huat
Moderator:	Dr Sugumaran

Key Assessment for Project Proposal	Supervisor Comments/Remarks	Moderator Comments/Remarks
Project Description - Is the problem or need to be addressed clearly presented? - Is the proposed approach or solution clearly presented and justified?	**Organize the problem statement in point form.**	**1. Suggest to separate the problem statement by sections.**
Project Scope and Objectives - Is the scope of the project clearly defined? - Are the objectives of the project clearly specified? - Are the project scope and objectives appropriate for a final year project?	**Relook at the objectives to ensure only relevant ones are included, and must be measureable**	**1. Suggest limit to 4 objectives only.**
Literature Review / Fact Finding for Benchmarking / Verification of Project - Are sources for literature review / fact finding appropriate? - Is information from literature review / fact finding relevant and adequate? - Is information from literature review / fact finding clearly presented and discussed?		
Research/Development Methodology and Development Tools - Is the methodology for the project clearly described and discussed? - Are the required development tools clearly described and discussed? - Are the stated methodology and development tools appropriate?		
Project Plan - Are the phases and tasks of the project properly defined and planned? - Are the phases and tasks consistent with the methodology of the project?		
Initial Deliverables - Are deliverables (e.g. use case diagrams and descriptions) of initial phases of the project plan included in the report?		
Report Structure and References - Is the report organised in a logical structure? - Are references listed in accordance to Harvard format?		
Language and Clarity of Writing - Are the sentences concise and understandable? - Are there spelling and grammar issues?		