

UTARoid – Android-based Announcement Portal for Students

By

Chin Wei Kit

Final Year Project II

Supervisor

MISS YAP SEOK GEE

A FINAL REPORT

SUBMITTED TO

University Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEM (HONS) BUSINESS INFORMATION
SYSTEM

Faculty of Information and Communication Technology

(Perak Campus)

Jan 2013

REPORT STATUS DECLARATION FORM

Title: UTARoid – Android-based Announcement Portal for Students

Academic Session: 2013/01

I CHIN WEI KIT

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

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

Verified by,

(Author's signature)

(Supervisor's signature)

Address:

No. 5, Jalan SP 3/4,

Bandar Saujana Putra,

42610 Jenjarom, Kuala Langat, Selangor.

Date: 8th April 2013

Ms. Yap Seok Gee

Date: 8th April 2013

Declaration of Originality

I declare that this report entitled “UTARoid – Android-Based Announcement Portal for Students” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : _____

Name : Chin Wei Kit

Date : 8th April 2013

Acknowledgement

Throughout the process of preparing and completing this project, I have received many assistance and guidance from various parties. Without these individuals who are willing to share their experiences and time to give me a helping hand, I may not have completed the project on time or in a better quality. Thus, in this section, I would like to express my heartfelt gratitude to all of these individuals who had supported me.

Firstly, I would like to thank my supervisor, Miss Yap Seok Gee who has guided me on how to prepare high-quality documentation and presentation for the project. She had also inspired me with her ideas on how to further enhance the application which is being developed. Thanks to her concerns on the project scopes that I proposed, I could also make a better evaluation on whether or not each proposed scope is feasible. Plus, with her constructive comments and suggestion, I could clearly see where my mistakes are and improve myself further throughout the processes in developing UTARoid.

Next, I would also like to thank my family members who have constantly provided me financial and mental support especially during hard times. Without their encouragement, I may not be able to withstand the stress being faced and eventually become demotivated.

Special thanks are also conveyed to friends especially Chong Jia Zheng who had also assisted me in the document-writing process by sharing their knowledge and references besides sharing his experience in Android application development and PHP website development. Besides, I would also like to express my appreciation to friends who are willingly to spend their time for being models of my project poster namely Cheah Tung Seng, Chok Lin Jiun, Lim Shu Shin, Tan Jia Ming, Tan Kuan Ong, Tong Woei Chyuan and Yap Eng Li and those who do not hesitate to be the participants of my survey.

Last but not least, I would also like to thank Faculty of Information Communication and Technology for providing facilities such as computer labs and Internet connection which had facilitated my progress toward finishing the project.

Abstract

This project is initiated as one of the solutions for a major problem faced by the staffs of Universiti Tunku Abdul Rahman which is to communicate announcements or alerts especially last minute ones to the students effectively and efficiently. Since smart phone users especially in the student community is growing tremendously, the platform chosen to be used as an announcement portal for this project is the Android platform.

With the availability of this application named UTARoid, students will be able to receive announcements and information on events happening in UTAR anytime anywhere as long as they have an Android smart phone with Internet capabilities in hand. Thus, chances of them to miss out any information or golden opportunities especially those which are essential for their studies, talents, lifestyle, achievements and future career will be greatly reduced.

With the flexibility of UTARoid, students will be able to be notified in a timely manner whenever new announcements or updated bulletins are posted. Simple forms which are needed to be filled-in physically could now be completed digitally; thus, proving that UTARoid also supports environment-friendliness. As UTARoid also aims highly to increase the loyalty of student to frequently visit this announcement portal, many other features which could enhance the attractiveness, interactivity, user-friendliness and categorization of the portal are also implemented.

Table of Contents

Title	i
Report Status Declaration Form	ii
Declaration of Originality	iii
Acknowledgements	iv
Abstract	v
Table of Contents	vi-ix
List of Figures	ix-xi
List of Tables	xii-xiii
List of Abbreviations	xiv
Chapter 1 Introduction	1
1.1 Project Background	1
1.2 Problem Statements	2-3
1.3 Objectives	3
1.4 Contributions	4
1.5 Scopes	4-6
1.6 Technology and Tools	7-8
1.7 Chapter Summary	8-9
Chapter 2 Literature Review	10
2.1 Android	10-13
2.2 System Development Methodology – Agile Method	13-14

2.3 Related Works	14-17
2.4 Requirement Gathering Techniques	18-21
2.5 Rewarding Experiences Boosting Up Loyalty	21-22
2.6 The Usage of JavaScript Object Notation (JSON)	22-23
Chapter 3 Methodology	24
3.1 Agile Method	24-27
3.2 Gantt Chart	28
Chapter 4 Requirement Documentation	29
4.1 Requirement Gathering Techniques Used	29-30
4.2 Scope Requirements	31-37
Chapter 5 Interface Design	37
5.1 Login Screen	37-38
5.2 Home Screen	39-40
5.3 Edit Profile Screen	40-41
5.4 UTARoid List Layout	41-42
5.5 UTARoid Multiline List Layout	43-44
5.6 View Announcement Screen	44-45
5.7 Post Comment Screen	46
Chapter 6 System Design	47
6.1 Use Case of UTARoid	47
6.2 Activity Diagrams of UTARoid	48-53

Chapter 7 Database Design	54
7.1 ERD Diagram	54
7.2 Data Dictionary	55-62
Chapter 8 Implementation & Deployment	63
8.1 Language Used	63-65
8.2 Software Used	66
8.3 Critical Factors & Vital Steps in Development	66-73
8.4 Special Features	73-76
8.5 Screens & Files Involved	77-85
Chapter 9 Testing	86-97
Chapter 10 Future Enhancement	98
10.1 Trigger of Notifications	98
10.2 Automatic Sign-In & Update	99
10.3 Add-On of Other New Modules	100
10.4 Sharing Announcement to Social Networking Sites	100
10.5 Filter of Harsh Words in Posted Comments	100-101
Chapter 11 Conclusion	101
References	102-104
Appendix A: Poster	A1-A2
Appendix B: Sample of Survey Sheet	A3-A7
Appendix C: Survey Results	A8-A15
Appendix D: User Manual	A16-A21

LIST OF FIGURES

Figure Number	Title	Page
Figure 2.1.1	Android Architecture Frameworks	11
Figure 2.2.1	Screen shots of several main tabs of Lim Kok Wing University Android Application	15
Figure 2.2.2	One of the User Interfaces of University of Virginia iPhone Application	16
Figure 2.2.3	The Only Display for Auto WiFi MMU Application	17
Figure 2.6.1	Example of How Information is Stored in JSON	22
Figure 2.6.2	Example of How Information is Stored in Array in JSON	23
Figure 5.1.1	Sketch of Login Screen	37
Figure 5.1.2	Sketch of Login Screen with Error Message Displayed	38
Figure 5.2.1	Sketch of UTARoid's Home Screen	39
Figure 5.3.1	Sketch of Edit Profile Screen	40
Figure 5.4.1	Sketch of UTARoid List Layout	42
Figure 5.5.1	Sketch of UTARoid Multiline List Layout	43
Figure 5.6.1	Sketch of View Announcement Screen	44
Figure 5.7.1	Sketch of Post Comment Screen	46

Figure 6.1.1	Use Case of UTARoid	47
Figure 6.2.1	Edit Profile Activity Diagram for UTARoid	48
Figure 6.2.2	Announcement & Comment Activity Diagram for UTARoid	49
Figure 6.2.3	Time Table & Event Activity Diagram for UTARoid	50
Figure 6.2.4	Bus Schedule Activity Diagram for UTARoid	51
Figure 6.2.5	Past Year Examination Activity Diagram for UTARoid	52
Figure 6.2.6	Reward Redemption Activity Diagram for UTARoid	53
Figure 7.1.1	ERD Diagram for UTARoid	54
Figure 8.1.1	Java Environment in Development Phase	64
Figure 8.1.2	Partial XML Coding Example	64
Figure 8.1.3	Example of Contents of PHP File Written in Notepad	65
Figure 10.1.1	Notifications in Android Shown to User via Status Bar	98
Figure 10.2.1	How to Acknowledge User on New, Updated & Unread Posts by Using Twitter App Example	99
Figure A1	Pie Chart on Amount of Students who Possess at least an Android Smart Phone	A-9
Figure A2	Bar Graph showing Frequency of Visit of Respondents to Existing Announcement Platforms	A-11

Figure A3	Bar Chart of Respondents' Opinions on Existing Platforms based on Characteristics	A-12
Figure A4	Pie Chart of Respondents' Favourite Announcement Platforms	A-13

LIST OF TABLES

Table Number	Title	Page
7.2.1	Data Dictionary for Student Table	55
7.2.2	Data Dictionary for Announcement Table	56
7.2.3	Data Dictionary for Read_Status Table	56
7.2.4	Data Dictionary for Comment Table	57
7.2.5	Data Dictionary for Department Table	57
7.2.6	Data Dictionary for Course Table	58
7.2.7	Data Dictionary for Staff Table	58
7.2.8	Data Dictionary for Event Table	59
7.2.9	Data Dictionary for Bus Schedule Table	59
7.2.10	Data Dictionary for Subject Table	60
7.2.11	Data Dictionary for Slot Table	60
7.2.12	Data Dictionary for Student_Reg Table	61
7.2.13	Data Dictionary for Past_Year_Exam Table	61
7.2.14	Data Dictionary for Reward Table	62
7.2.15	Data Dictionary for Reward_Trans Table	62
9.1.1	Test Plan	86-97

Table A1	Results on which Existing Platform that Respondents Usually Visit for Announcements	A-11
Table A2	Ranking of Most and Least Important UTARoid Features Expected by Respondents	A-16

LIST OF ABBREVIATIONS

E

ERD Entity-Relationship Diagram

F

FICT Faculty of Information Communication and Technology

G

GPS Global Positioning System

J

JAD Joint Application Development

JSON JavaScript Object Notation

M

MMU Multimedia University, Malaysia

S

SDK Software Development Kit

SLDC System Development Life Cycle

SQL Structured Query Language

U

UTAR Universiti Tunku Abdul Rahman

W

WBLE Web-Based Learning Environment

Wi-Fi Wireless Fidelity

X

XML Extensible Markup Language

Chapter 1: Introduction

In this chapter, the background of the project, problems faced and the objectives are described. Then, the contribution in which this project is expected to produce is briefly elaborated. The scopes, tools and technology of the project are also explained in details under this chapter. Finally, a chapter summary is provided as a guideline for readers on what contents are available in each chapter of the report.

1.1 Project Background

Announcements are frequently made to students of a university to notify them on various events such as competitions, workshops, enactment of new rules and regulations, release of examination results and perhaps cancellation of classes. In Universiti Tunku Abdul Rahman (UTAR), the delivery of these announcements have been made through various media namely the use of e-mail, intranet, official website, web-based learning environment (WBLE), Facebook, Twitter and forums except for one – Android application.

The use of Android smart phones has been increasing enormously due to its variety of features which come in various design and prices. As a proof, based on an analysis of global fourth quarter 2011 smart phone sales made by Gartner, Android has been dominating the market share (50.9%) followed by iOS (23.8%). With that report, it can be seen that Android has remain at the top of the smart phone market, defeating other operating systems namely Symbian, Research in Motion (RIM) and Bada. As technology will never cease to improve, Android platform will be frequently updated into newer and better version to attract new consumers and maintain the loyalty of existing ones. (Egham 2012)

Thus, making use of the capabilities of Android technology to convey public statements or alerts to students in an effective and efficient way should not be missed. Under this project, a deep research will be performed on the technology behind which can be used to create an Android application that fulfill all the scopes mentioned. As this method of conveying announcements is a pioneered method, the target audience aimed for this project involves students of UTAR Kampar Campus who owns a smartphone installed with Android operating system and has their own Internet data plan (or WiFi enabled). By the end of this project, the aforementioned prototyped Android application is to be completely developed, tested and presented.

1.2 Problem Statements

For years, communicating vital information and announcements to the student has been a major problem faced by most faculties in UTAR including FICT. Many approaches have been used previously but still messages were not delivered effectively and efficiently to students due to various reasons. As a result, students are unaware of essential messages especially last minute updates causing inconveniences and perhaps causing students to miss out golden opportunities for their career or study life.

The major reason students are not likely to view these announcements published in most media is due to attractiveness. The announcements are poorly organized and categorized leading to students having difficulties in finding important messages or alerts only related to them. Besides, the formality of some channels also caused the use of unattractive fonts, colours and less multimedia elements thus, decreasing student's loyalty to assess to these announcements regularly.

Plus, notifications of announcement updates are also not acknowledged by students effectively. In other words, students will most likely be unaware of newly posted announcements or bulletins unless being notified by friends, lecturers or any other coincidental situation.

Announcements posted in the existing approaches also do not promote interactivity. For instance, students could not make any enquiries, comments or suggestions for any announcements posted in the Intranet or the official website. Hence, the students' doubts or confusion could not be cleared on the spot leading them to be uninterested of getting details of the particular announcement. Even approaches such as Facebook and Twitter provides a platform for interactivity, many third parties are involved, causing a high probability for the announcement posted to lose its credibility and accuracy.

Certain tasks such as editing personal details also cause a hassle for students as they need to travel back and forth to their own faculty to complete them. With the existence of an Android application specially tailored for student use, the stated tasks could most probably be performed in a timelier manner. Plus, usage of unnecessary forms could be reduced; promoting an environmental-friendly culture.

Finally, as there are thousands of students being divided between courses available in UTAR, students may only communicate and socialize with other students only within their same group. Thus, sharing of knowledge and skills of different fields between students is hard to be achieved. With the online capabilities of smartphones nowadays, communicating with each other especially through the proposed UTAR android application could most probably solve the stated problem. Plus, if this project is to be successful and expanded, students from all branches could befriend and communicate ideas with each other easily.

1.3 Objectives

The following are the objectives which are to be achieved from the development of the UTARoid application:-

- a) Delivering messages, announcements and last-minute updates to students effectively and efficiently.
- b) Being able to attract or encourage students to frequently use the application thus ruling out chances for them to miss any important bulletins or alerts
- c) Providing a more flexible and interactive announcement platform compared to existing ones for students
- d) Allowing students to access to list of related announcements with just a few simple clicks anytime anywhere as long as the application is installed and Internet is available
- e) Reducing the gap between students of different courses within UTAR thus enhancing a friendly study environment
- f) Reducing the use of paper forms or documents by replacing them with digital ones; thus promoting environment-friendliness

1.4 Contribution

As mentioned, with the existence of this application, staffs of UTAR will be able to post announcements, alerts or last minute updates anytime with higher chances that students are aware of them and will consequently read them.

Students, meanwhile, will benefit by ruling out chances of missing golden opportunities which are vital for their studies or future career. Plus, they will not miss being notified with their latest favourite or subject-related announcements as long as they install the application and are constantly signed in to the platform. Since interactivity is promoted within this application, besides being able to expand students' social circle and share knowledge within the faculty, any doubts regarding an unclear message could also be clarified swiftly.

Plus, as forms can be completed digitally and other documents such as past year papers and bus schedules can be downloaded and saved directly to the student's smartphone, the existence of UTARoid is also said to contribute to the environment in a positive manner.

1.5 Scopes

The following are the scopes and requirements of the chosen final year project:-

a) User Verification

In order to further explore the contents of the application, a verification process will firstly be initiated to determine if the user is a registered student or a valid staff in UTAR. This verification process will be performed with the aid of a database. Successful verification of a student will eventually bring him/her to the main screen where all the personalized and customized settings are being displayed. As for staffs that are responsible to channel out announcements, a fixed screen which allows them to do so will be displayed after successful login. Failure in verification will result in an error message.

b) Auto-Filter and Categorization of Announcement

For a more organized and user-friendlier view of announcements, announcements will be categorized based on departments or classes depending on the type of announcement. For general announcements meant to be spread virally either targeted to a group of students or every student of UTAR, auto-categorization functionality will be implemented in such a way that these announcements will be categorized according to the department it originates from. A function shall also be found within UTARoid where students can view related lecture, tutorial or practical classes' announcements without any confusion triggered. As for auto-filtering functionality, whenever there is no announcement being channeled from a particular department, the department itself will never be listed in the categorized list; saving more space and avoiding unnecessary mobile processes. With auto-filter and categorization, one of the objectives of UTARoid which is to enable students to find related announcements easily and efficiently shall be accomplished.

c) Attractiveness and Simplicity

The application will be designed in such a way that it incorporates attractive and creative elements which could eventually promote user exploration and maintain student's loyalty towards the application. In contrast with the complexity of existing platforms in displaying announcement to students, announcements presented via this application will be done in a much simpler manner in terms of organization and ease of access. Every module available in UTARoid shall be accessible via fewer clicks when compared to existing platforms.

d) Downloadable Contents

Some category of announcements/information should also be able to be downloaded directly to the user's phone such as Bus Schedule and Past Year Papers. The downloadable contents could be in the form of PDF or any other formats supportable by Android devices. The application shall be able to detect if the user's phone is installed with a suitable document reader and error message will be prompted out if no such reader is found within the phone. If a suitable document reader is found, a progress bar shall be displayed showing the users the download progress of the respective file. Downloaded files should be readable even in offline state.

e) Profile Edit & Digital Forms

Within the application, students can also update their personal information such as home address and e-mail address anytime anywhere as long as there is an Internet connection, thus, avoiding the need to travel back and forth to the faculty just for profile updates. Any other simple registration or transactions will also be made available in the application, replacing the need of paper forms to digital forms.

f) Interaction-Enabled Announcements

For every announcements made, students should be able to enquire, comment or make suggestions using their own profile ID. Administrator or the staff who is responsible in posting that particular announcement will then respond appropriately towards the statements made by the students. Other students are also allowed to participate if there are any discussions being held in which they can share their knowledge.

g) Reward Points System

In order to increase the loyalty of students towards the developed application, a reward points system should be implemented in such a way that every announcement read will entitle the particular student an amount of points each. As soon as enough points are accumulated, student then can exchange these points with rewards provided by any party. Initial suggestions include soft skill points, soft skill incentives or affordable hardware or software. For this functionality, the application shall be able to detect if a particular student has read an announcement previously to prevent any duplication of points gained.

h) Platform to Post or Update Announcement

There will be a single way which will be provided to the administrators or staff who wish to post or update announcements namely via website itself. This method requires user verification process before users can actually proceed to the next displayable. As soon as login is successful, the particular admin or staff will be able to check previously posted announcement, each announcements' comments and also post new comments. As the UTARoid application is more focused in this project, dummy websites will be developed for this functionality.

1.6 Technology and Tools

The following are the technology and tools which will be used in the development of this project:-

a) Android Software Development Kit (SDK)

The Android SDK which is freely downloadable via Android's main website provides developers the API libraries and tools necessary to build, test and debug apps for Android. Plus, the Android library includes specifications for the use of multitasking, drag and drop and swiping between screens. Thus, if the Android application for this project is to be developed, this Android SDK is vital to be used. Since different computer platforms require different type of SDK, a proper checking in terms of compatibility and system requirements should be performed before downloading and installing a SDK. The Eclipse IDE, a platform which allows programmers to code in Java, will be installed first before plugging-in the chosen Android SDK.

b) Java Programming Language

Most of the important features and interfaces in the UTAR Android application will be developed using Java programming language since most Android devices support this language. Since Android implements the open source platform, available code samples and supports from fellow developers can be easily accessed thus, finding, sharing and the enhancement of solutions for problems can be easily performed.

c) Structured Query Language (SQL)

Most of the functionalities in UTARoid require the retrieval, process, storage and usage of data. Hence, database is required and a standard language for accessing and manipulating this database is to be chosen. Since the knowledge of SQL is easier to grasp and matches well with the use of XAMPP (web server for UTARoid), this query language will be used for the development of UTARoid. Among the capabilities of SQL include executing queries, retrieving, inserting, updating and deleting data, creating database and tables and setting permissions for each element of the database.

d) XAMPP

XAMPP, a cross-platform web server solution stack package, will be installed and used in UTARoid's development machine since it is free and open-source. Plus, since PHP will be used as the main programming language for the development of websites used to manipulate data for UTARoid, XAMPP is deemed to be the best package to be installed since it supports and incorporates well with the aforementioned language. Among the included tools in the package are Apache HTTP software and MySQL database. An important module within XAMPP is the phpMyAdmin where all manipulation of UTARoid database will be performed

1.7 Chapter Summary

In Chapter 1, the background of the project which include why this project is importantly needed is explained clearly. Besides, the problem statements, objectives, scopes, contributions of the project are listed and described also in this chapter. Finally, the technology and tools which is expected to be used for UTARoid application is also listed and elaborated.

In the next chapter which is the Literature Review, all reviews of any topics made from journals, articles, websites and any other sources will be written. The topics which include the Android platform, Related Works, System Development Life Cycle (SDLC) Methods, Agile Methodology and Requirement Gathering Techniques will be reviewed in terms of their definition, background, pros and cons.

In the third chapter of the project named Methodology, the name of the SDLC method selected and the reason behind why this methodology is chosen for the project will be explained. Besides, all activities which is to be undergone throughout this project is described in phases starting from the Planning stage up to the Maintenance stage. A Gantt Chart is also included as a reference or schedule on which activity is to be completed during which period of the project.

In Chapter 4, requirement gathering techniques used will be explained here. The explanation will include why the techniques, namely Survey and Interview are used, who are the participants and what are the expected outcomes. Besides, requirements for each scope listed in Chapter 1 is also listed under this chapter named Requirement Documentation.

Chapter 5 consists of interface designs where all initial designs for screens in UTARoid will be sketched and explained thoroughly. The standard size of logos, banners and other visual elements will also be included in this chapter. This chapter serves as a guideline for future programmers and designers especially when new modules are to be included into the application. It can also be acted as a storyboard for the application.

Chapter 6 mainly consists of all the diagrams vital for system designs. Among the diagrams which will be found in this chapter includes Use Cases and Activity diagrams. Use Cases depicts the relationship between the system and the users of the application whereas activity diagrams show the flow of performing a particular activity within the application. Meanwhile, Chapter 7 fully focuses on the database design of UTARoid which include ERD diagrams, Data Dictionary and CRUD.

The next chapter will be the Implementation and Deployment chapter which specifically explains the development of the application from the scratch to the end. Among the information which could be found are languages and procedures used, special features (with partial codes being shown) and variable standards. For each module, related Java files, XML files and PHP files will be listed. Screen shots will also be included along the explanation for a better understanding on how the functions are implemented. Other warnings and extra information are also highlighted to avoid any critical errors to arise from the application itself.

Chapter 9 is written for the Testing phase of the application. In this chapter, test plans and results of each test plan will be displayed and elaborated. Chapter 10 suggests all future enhancements which could be incorporated into UTARoid with the objective of making the application more sophisticated and versatile.

The final chapter which is the Conclusion concludes the contents of the documentation from Chapter 1 to Chapter 10.

The Appendix section will include five major sections which are the Poster Design, Sample of Survey Sheet, Survey Results, User Manual and Turnitin Report.

Chapter 2: Literature Review

This chapter reviews on topics of Andoid, SDLC methods, Agile Methodology, Related Works and Requirement Gathering Techniques from various sources such as journals, articles, conference papers and websites. The reviews include the topic's background, functionalities, pros and cons.

2.1 Android

Android platform can be generally defined as a software bunch for mobiles which include an operating system, middleware and key applications. Google which acquire Android Inc. in 2003 has been constantly researching and updating the versions of Android to fix bugs faced by previous versions besides adding in more functionalities and specifications. The latest version of Android which can be found in the market is Android 4.1, Jelly Bean which was released mid-2012. Among the improved functionalities implemented on this latest version are enhanced accessibility, bi-directional text and other language support, expandable notifications, resizable app widgets, high resolution contact photos, Android Bean and multichannel audio. (Developer.android.com 2012)

Android platform can be divided into five primary layers namely:-

- a) **Applications** - Set of core application that comes with Android distribution such as Calendar, Map Browser, Web Browser and Contacts application
- b) **Application Framework** - Facilitate reuse of components in Android. With this framework, developers could build applications to execute on Android Kernel.
- c) **Libraries** – Includes System C Library, surface manager, 2D and 3D graphic engines, Media Codecs, SQLite and web browser engine, LibWebCore
- d) **Android Runtime** - Consist of a set of Core libraries that provides most Java functionalities and Dalvik virtual machine which act as a translator between application side and operating system
- e) **Linux Kernel** - For core system services

With all these layers integrated, Android platform is said to be successful in offering many features to the end user which covers the area of application development, internet, media and connectivity. (Maji, Kangli, Sultana & Bagchi 2010)

The following diagram shows a clearer picture of the Android layers:

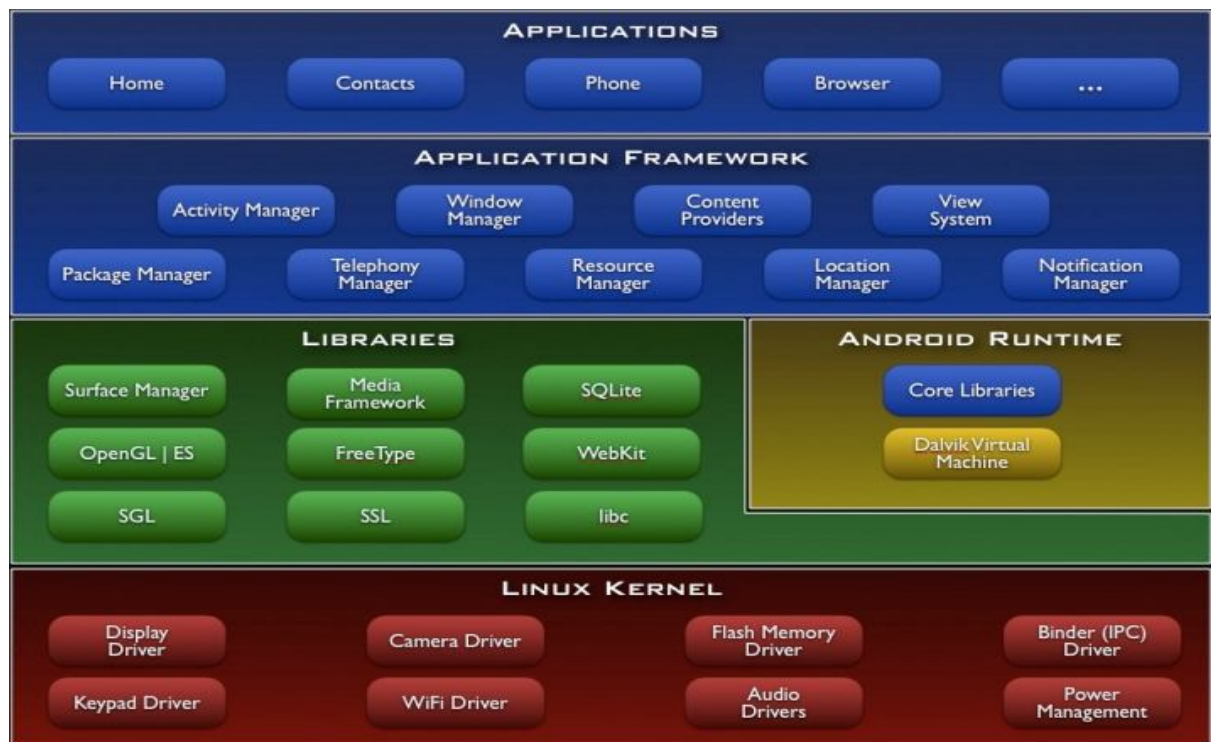


Figure 2.1.1 – Android Architecture Frameworks (Developer.android.com 2012)

There are many strengths of Android which are worth to be applauded for. For instance, developers may most likely prefer creating apps for Android devices since Android offers an open development platform, allowing them to use third-party tools to create the particular app. With the implementation of open source platform, anyone with the right skills and knowledge can create an app specially tailored for Android devices without tight restrictions or approvals. Besides, this strength of Android also enhances flexibility since it allows developers to update their apps anytime to accommodate a range of different devices and play around with many features and functions of their app during the development process. (Priya Viswanathan n.d.)

Developing an Android app may also be easy especially for Java programmers since all applications under Android mainly use Java, a commonly used programming language in current days. Hassell (2010) also commented that since Android implements the open source platform, available code samples and supports from fellow developers can be easily accessed

thus, attracting more new potential developers to learn and create apps either with basic or complex functionalities.

Meanwhile, an excellent testing environment is also provided by Android that encourages developers to test and debug apps properly before presenting them to the Android Market. Android Inc. has also make developers' life easier by providing downloadable System Development Kit (SDK) in its website which comes with the API libraries and developer tools necessary to build, test and debug apps. (Developer.android.com 2012)

In contrast with its major competitor's application market, Apple's App Store which rejects application submissions that are deemed too simple or are similar to existing apps, Android Marketplace doesn't censor its apps. In other words, apps developed by any developers will have equal chances to enter and to be published in the market but of course, the best one will only succeed. Meanwhile, since Android is an open platform, manufacturers have the flexibility to pair the OS with any hardware they wish, thus providing end users a higher number of device selections in terms of specifications, design, screen and other physical appearances. (Escallier 2010)

However, Android does have its cons too. Though open source platform is part of Android major strengths, this technique has also raised security and quality control concerns to app developers. Since anyone can develop and present new apps in the market without much restrictions and approvals, end users may purchase low-quality apps numerous times thus wasting their resources. With the open source concept, developers may also face piracy threats and loss of potential profits since anyone could easily access to the same learning resources and produce similar app. Meanwhile, hackers could also easily mimic popular apps and trick end users into using the fake app which may contain malicious elements. (Dempsey 2012)

Since Android is being licensed to multiple phone manufacturers such as Samsung, Huawei and HTC, apps developers also face problem in successfully presenting their apps across all Android devices since each manufacturer has its own configurations and specifications (especially processing power) being implemented on its respective smart phones. In other words, there may be an incident in which some devices installed with certain versions of Android will not be able to run applications that work fine either on the same device or another device of different Android version. With such problem, fixing bugs and

updating apps will also cause headache to developers as they have to make sure that the app could be supported by all Android devices as soon as an update is performed. (Rodriguez 2012)

2.2 System Development Methodologies – Agile Methods

Association of Modern Technologies Professionals (2012) defines System Development Life Cycle (SDLC) as a conceptual model used in project management that describes the stage involved in developing an information system, starting from initial feasible studies. As time passes, SDLC is no longer only incorporated in Waterfall Methodology (the traditional way of SDLC) but is also applied in other various methodologies including Rapid Application Development, Object-Oriented Analysis & Design and Agile Methodology in which each methodology has its own pros and cons and accommodates certain situation of a system development.

Meanwhile, Wiras Adi (2008) stated that SDLC is a process of gradual refinement in which each development phase continues and refines what is done in the previous phase. The commonly known development phases are namely Planning (understanding why system has to be built and its requirements are determined), Analysis (identifying and analysing problems besides predicting problems which may arise in the future), Design (determines how exactly system operates in its environment), Implementation (stage where system is built, tested and installed) and lastly, Maintenance.

Agile Methodology has existed since in the 90s solving most frustrations met when the waterfall approach of the SDLC is being commenced. As mentioned, it is one of the approaches to project management but what differ it from the rest is that it helps team respond to unpredictable situations while building a software or application through incremental and iterative work cadences. Plus, agile methods also emphasizes on collaboration among customers and development teams especially when requirements and solution are to be determined. (Haughey D. 2009; Gatherspace.com n.d.)

Besides having the benefit of being incremental and iterative unlike the traditional approach which does not allow developers to return from the current phase back to the previous phase, agile methods also promote real-time communication over documents. Thus, better understanding of the requirements and retrieval of customer reviews and feedbacks can

be performed swiftly. Plus, since customers are majorly involved and testing is performed in a continuous manner, chances of reworking certain phases of the project will be minimized. (Haughey D. 2009)

In fact, there was also a project performed by Andrei Christian Spataru in which he researched the suitability of agile methods for mobile development and even bring a set of improvements to established agile method called Mobile-D. In his report, he did quoted that agile methods are unique in the sense of individuals and interactions are more important than processes and tools, working software is more valuable than comprehensive documents, customer collaboration is more preferred than contracts and adaptability is valued higher than accordance to plans. (Spataru A. C. 2010)

2.3 Related Works

a) Limkokwing Android Application

Lim Kok Wing University of Creative Technology has developed an Android application as a means for a mobile version of its student portal. With this application, students are able to access the latest information such as events, class schedules and exam results besides receiving reminders on outstanding fees or assignments. (Limkokwing University of Creative Technology n.d.)

Before being able to access the portal, student login is firstly needed. The following screens upon successful login are then generally organized in tabs. The “Today” tab generally summarizes student’s time table for the day itself with details of each class including its time, lecturer and location. Meanwhile, the “Calendar” tab is then further divided into two sub-navigation tabs namely “Timetable” and “Events” in which “Timetable” displays weekly class schedule with day’s classes highlighted whereas “Events” shows the events that are happening in the university. (Limkokwing University of Creative Technology n.d.)

The third tab which can be found in the navigation tab upon successful login is the “Module” tab. The function of this screen is to display an overview of all the subjects that the student is currently taking for the semester and the days these subjects are held. Upon selecting any of the displayed subjects, student will be able to view more details such as location, time and lecturer as well as downloading course materials for the particular subject.

The next main tab will be the “Profile” tab which is yet again divided into three sub-navigation tabs namely “Information”, “Results” and “Fees”. As the “Information” tab is being pressed, the student details which include Student ID, Passport/IC No., Date of Birth and Course will be displayed. Meanwhile, the “Results” tab will show student’s exam results for the previous semester and the “Fees” tab displays the outstanding fees that the student owes the university. Other modules and functions can be accessed by the students whenever the “More” tab is being selected. (Limkokwing University of Creative Technology n.d.)

The following diagram shows the screen shots of several main tabs of the Android application specially catered for Lim Kok Wing University students:-

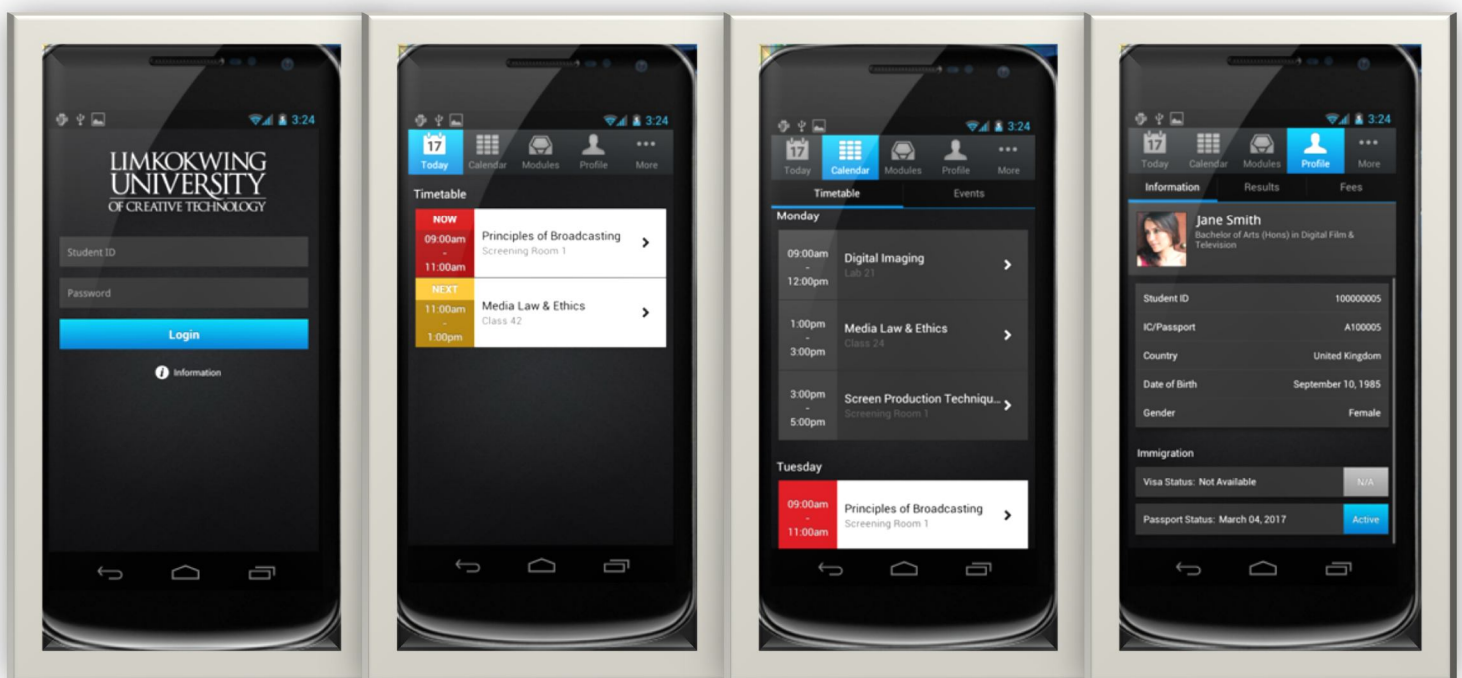


Figure 2.2.1 – Screen shots of several main tabs of Lim Kok Wing University Android application (Limkokwing University of Creative Technology n.d.)

b) University of Virginia Android/iPhone Application

University of Virginia, a public research university located in the United States of America has eventually selected WillowTree Apps as the main developer of its Android application. One of the application sophisticated features includes an augmented reality view

of the campus in which students can use it to search for buildings on grounds besides identification of a building. A campus map is also readily available in the app and with the integration of a Global Positioning System (GPS), students could pinpoint which location of the campus they currently are at.

The application also provides Athletics feature where students could access to updated scores, schedules, videos and photos of a particular sport event and even the profile of an athlete that they favour. Other brilliant features include Courses where student could search for classes and find out in-depth information of a selected class, Alumni where graduates are updated with Homecomings and Reunion schedules, Health System where students could directly make an appointment with health consultants of the university, Student Support where medical advice and other motivational efforts are provided and Alerts where students will receive push notifications whenever there are new stories or announcements in the university itself. (WillowTree Apps, Inc 2012)

The following diagram shows a screen shot of one of the user interfaces of the application:-



Figure 2.2.2 – One of the user interfaces of University of Virginia iPhone application
(WillowTree Apps, Inc 2012)

c) Auto WiFi MMU

A small group of Multimedia University, Malaysia (MMU) students has took the initiative to develop an Android application specially tailored to other students and staffs of the same university to automatically connect to MMU wireless network with WiFi SSID “MMU”. In other words, students and staff will no longer need to manually open up the browser and key in username and password just to connect to the WiFi provided within MMU campus, thus promoting convenience. There are only two requirements needed for the particular application to work which are the standard password for the hotspot which can be easily gained from around and a student or staff MMU Webmail account. (Yap W. J. 2012)

The following diagram shows the only user interface for the application:-

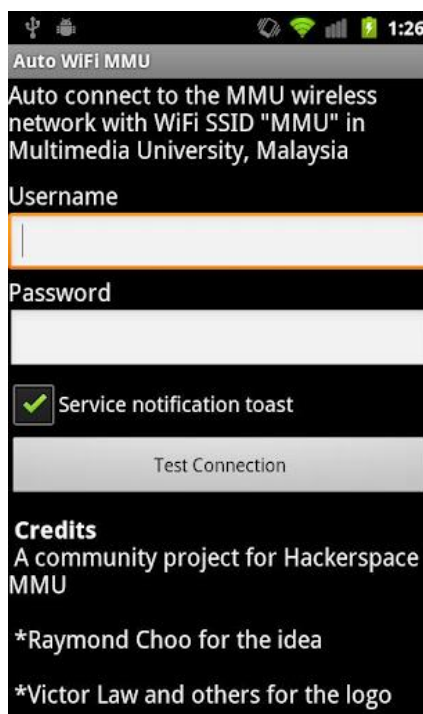


Diagram 2.2.3 – The only display for Auto WiFi MMU application (Yap W. J. 2012)

2.4 Requirement Gathering Techniques

For every project especially those which involves development of applications and systems, gathering the right requirements during the analysis phase of the SDLC is highly compulsory. In fact, every developers especially senior developers and team leaders, should at least possess the basic skills of gathering good requirements for their project. If requirements are vague, unclear or undefined at the earlier stage, risks of redoing the whole project will be high which eventually cause wastage of time, financial resources and energy. Plus, quality of the final deliverable may be badly affected too. Every developers should bear in mind that the final solution of the project should meet the targeted user needs. A project will still deemed as bad or of low quality no matter how attractive or high-tech the system is if requirements of users are not satisfied. (Mochal, T. 2001)

There are many approaches which are commonly used to gather user requirements including Interview, JAD Session, Survey Method, Prototyping and Observations. Each of these methods have their own pros and cons and developers are advised to wisely choose the best technique depending on their current time constraints and situation. For every project, combination use of more than one technique is encouraged especially when there are diverse set of clients and stakeholders. (Mochal, T. 2008)

The interview technique can be defined as a method which involves conversations, particularly question and answer sessions, with stakeholders to validate needs and requirements including high-level ones. Interview is also said to be a technique which could discover discrepancies between needs of stakeholders and to be used as a medium to obtain approval from stakeholders on already defined needs, requirements or changes on them. (Samuel90 2010)

Interview sessions can be divided into two types namely one-on-one interviews and group interviews. One-on-one interviews generally involve only one client or stakeholder to be questioned during the session whereas group interviews involve more than one interviewee – usually two to four. Compared to one-on-one interviews, group interviews requires more preparation and formality to obtain information since interviewees may have contradict opinions or requirements within themselves. However, if the group being interviewed could remain focus, richer set or requirements can be determined swiftly. (Mochal, T. 2008) For every session, asking the right interviewees (clients or stakeholders)

the right questions are highly vital. Open-ended questions allows interviewee to freely interpret their ideas and provide details in any way he/she likes whereas context-free questions provide a better insight into requirements since there are no boundaries involved besides being able to detect requirements which are not defined in earlier stage. (Jones, M. 2011)

Among the major advantages of using interview sessions include minimal preparations are needed, thus making this technique easy to be conducted besides providing an opportunity to explore or clarify topics in a more detailed manner. However, interviews have risks in which interviewees may have preconceived or biased statements which will affect the accuracy of the requirements gathered. When a large number of stakeholders are involved, interview sessions are also deemed as inefficient and time-consuming. Plus, additional interviews are needed to resolve conflicts and inconsistencies between stakeholder information, thus, consuming extra time just to clarify the needs for the project. (Samuel90 2010)

Meanwhile, Joint Application Development (JAD) technique is an extended facilitated workshop which involves stakeholders and system analysts to closely collaborate together to identify needs and requirements. In Mochal, T.'s article (2008), he clearly states that participants will stay in the JAD session until a complete set of requirements is documented and agreed to and session objectives are all met. The JAD participants must include an executive sponsor, facilitator, recorder and other stakeholders who are either directly or indirectly involved with the project in discussion. (Samuel90 2010)

The JAD technique may produce relatively high amount of best quality information within a small period of time and resolve discrepancies immediately with the aid of facilitator but for all these advantages to be experienced fully, significant planning and scheduling effort and commitment of stakeholders to attend and participate in the sessions are needed. Plus, facilitation and recording processes also requires trained and experienced personnel who must also possess basic technology skills in terms of use of hardware and software. (Samuel90 2010)

Survey method involves use of electronic or paper-based medium to garner needs and requirements from clients or stakeholders. This method is best used when there is a need to gather information from up to hundreds to thousands of stakeholders and time is a limiting

factor. Plus, with the use of survey, developers or team leaders may retrieve focused data efficiently and validate assumptions effectively. In the process of conducting a survey, little scheduling effort and stakeholder time and commitment are required, thus making the process an easy-going process. For data which are required to be tabulated to gain statistical preferences information, survey is highly suitable to be used. (Samuel90 2010)

However, survey method does have its cons like any other techniques. If the survey is made to be very large, responses from participants may most likely to be low. Questions being asked in the survey may also reflect developer's preconceived ideas or assumptions, thus reducing the chance of getting information on what user really need. The use of jargon words and improper type of questions being asked could also risk the quality of the survey results. In other words, a well-made survey requires trained and experience individual to develop. If survey is developed in a low-quality manner, chances of getting bad quality information is high and eventually affect the requirement gathering process. As results of the survey are gathered, survey designers may face conflicts and inconsistencies in information from stakeholders and extra time is required to perform additional analysis as a solution. (Samuel90 2010)

The observation technique, meanwhile, requires project team members to constantly visit the site where the future developed system will most likely be implemented and observe how targeted users use the existing system. From the observation, observers may be able to witness and experience the current problems faced by the users and obtain a clearer view on what they really need from the new system. The disadvantages of this technique, however, include the need of high amount of time spent and commitment from the observers as visits need to be frequently performed until requirements are adequately gathered. Besides, there may also be cases in which workers being observed do not act like what they usually do in the site, thus, affecting the quality of information being retrieved. (Jones, M. 2011)

A more modern approach named prototyping is also introduced to gather requirements for a project. By using this approach, preliminary requirements are being gathered via the use of a prototype. In other words, an initial version of the solution will be developed first before being demonstrated to the targeted users. Then, evaluation will be made and the requirements will be obtained from this evaluation results. Though prototyping excel in communicating ideas to potential users and validating if the functions developed are

what they really need, high cost, time and commitment are needed since prototyping is an iterative process. With that said, prototypes may be developed continuously throughout the life cycle until the business needs are confirmed and satisfied. Besides, since the prototypes may not be the final deliverable, most of them may just be disposed after evaluation, thus resulting in wastage of financial resources. (Mochal, T. 2008)

2.5 Rewarding Experience Boosting Up Loyalty

It has been a trend in nowadays mobile apps to provide rewarding experience to its users in order to gain higher chances for their comeback. In other words, with loyalty programs being incorporated in a mobile application even if it is of gaming, sales or informative types, users will be most likely to re-visit the application and consequently, aid in fulfilling the businesses of the application and strengthening the loyalty between the two parties. In order to survive in the ever-growing mobile application market, it is estimated that there will be a dramatic growth in loyalty programs being instilled in the industry at the year 2013. In fact, consumers will also most likely experience reward programs which do not only limit in rewarding them after purchases are made. (Clay, K., 2012)

There are three rewarding ways which are deemed to be creative in respect of building loyalty between customers and mobile applications. The first way is as easy as gaining reward points which is already been used by thousands of brands worldwide. This particular program which attracts mostly the generation of younger age (the largest community in the use of websites and applications) not only encourages users to perform activities within the application more often but is also vital for a particular business to gain information of customer's trends. (Brady, A., 2012)

Another creative way of loyalty programs is going "card-less". With this functionality enabled within an application, a user could transform all his/her reward cards digitally on his/her phone; thus, preventing the need to bring heavy wallets. Another benefit of such method is that the digital cards can be used both offline in physical shops and online during online transactions are made. (Brady, A., 2012)

There are customers who prefer instant rewards rather than slowly gaining points for redeem purposes which takes more time and effort. Thus, some mobile applications has put this factor into consideration and subsequently, introduce an instant reward technique within

the app itself. FrontFlip is the best example where after a particular in-store or online purchase is made and a QR code is being scanned, the member will directly gain a prize ranging from 25% off for the next purchase, free entry vouchers to special promotions. (Brady, A., 2012)

2.6 The Usage of JavaScript Object Notation (JSON)

JSON is the abbreviation for Javascript Object Notation which is famously used by programmers and application developers for its organized, logical and easy-to-access mechanism to store information. Plus, with the implementation of JSON in the background of the system, a set of collected data could be projected out in a human-readable format besides being able to be manipulated easily. (Lengstorf, J., 2009)

JSON is also deemed as a vital tool for nowadays development involving retrieve and storage of data as it supports the loading of data in a quick and asynchronous manner without delaying any page rendering especially in mobile applications and websites. JSON can be implemented in various programming standards such as JAVA and PHP, thus, providing flexibility and acting as a glue between a single server to multiple clients at once. (Lengstorf, J., 2009)

The following figure shows an example on how a set of information of an individual is stored in JSON format:-

```
1 var jason = {  
2     "age" : "24",  
3     "hometown" : "Missoula, MT",  
4     "gender" : "male"  
5 };
```

Diagram 2.6.1 – Example on How Information is Stored in JSON Format (Lengstorf, J., 2009)

Based on the diagram above, the data of an object (age, hometown and gender) named Jason is being stored. Between the curly brackets after the object name is initiated, any number of properties can be declared using the standards “property-name”:“property-value”. Each property of the object is separated by a comma. This function is especially useful whenever there is a need in a mobile application to read and display data of a single object

queried from a PHP file using SQL syntax. In other words, if a SELECT statement manages to read data of an object from the database, all the data (columns of a record) will be stored as seen in the above format, sent to the mobile application background upon calling a few methods and thus, processed according to the application need. The update, insert, and deletion of data of a single object will also use the above format as information storage means. (Tamada, R., 2012)

Multiple data of different objects can also be stored within a single array of JSON format as illustrated in the below diagram:-

```
1 var family = [{
2     "name" : "Jason",
3     "age" : "24",
4     "gender" : "male"
5 },
6 {
7     "name" : "Kyle",
8     "age" : "21",
9     "gender" : "male"
10 }];
```

Diagram 2.6.2 – Example on How Information is Stored in an Array of JSON Format
(Lengstorf, J., 2009)

In the figure above, the variable “family” stores two objects by enclosing them with square brackets thus, forming an array. This is especially useful when a mobile application requires to read information of more than one objects from the database server. If a SELECT statement manages to read data of multiple objects from the database, all the data (columns of the records) will be stored as seen in the above format, sent to the mobile application background upon calling a few methods and thus, processed according to the application need. The update, insert, and deletion of data of multiple objects will also use the above format as information storage means. (Tamada, R., 2012)

JSON does have its limitations when it comes to the support of data types. Data types that JSON supports are only limited to strings, numbers, Booleans and nulls. For Date types, developers are only given the choices to either transform them into string uses unix timestamp as integer. (Nitschinger, M., 2010)

Chapter 3: Methodology

In this chapter, the chosen methodology for the project will be described. The description will include which methodology is chosen and the reasons for the choice made. For each phase of the methodology life cycle, activities which are planned to be executed will also be explained. A Gantt Chart is included in this chapter as an initial schedule of tasks to be completed throughout the project life cycle.

3.1 Agile Method

For this project, the methodology which will be used to develop this application will be the agile method. As mentioned, the agile method promotes collaboration between both the customers and developers to retrieve immediate feedback and to acknowledge their changing requirements swiftly. As this Android application's objective is to solve the main problem of student's decreasing loyalty towards the existing announcement platforms, interaction between them and me, as a developer is highly essential. Plus, with their immediate feedback or additional requirements during the testing stages, modification of the features and contents of application could be performed without any doubts. Since the time for the entire project to be completed is only a year and at least an estimated of 2-3 weeks are needed to build up a scope, agile methodology will be a more efficient method when compared to the others. Lastly, since agile method is highly iterative, developers could easily refine each of the development phases either the ones which are completed or the one which just started anytime as appropriate.

As mentioned earlier in Chapter 2.2, Agile methodology incorporates the SDLC phases starting from the Planning phase up to the Maintenance phase. Activities which are to be performed in each phase are described in a detailed manner as follows:-

Phase 1 – Planning

In this first phase, a major problem statement which has affected a group of individuals is to be identified. The problem identified should be able to be resolved using current readily available technology. Next, a suitable title for the project which is to be developed to solve the identified problem is proposed. Both the objectives and the scopes of the project were then clearly defined. Meanwhile, the technology, platform and tools to be used to solve the particular problem were also determined at this phase. These steps are highly important as the

objectives, scopes and selected tools and technology will affect the entire project outcome besides influencing if the end product is capable enough to solve the identified problem in an effective and efficient way.

Under this phase, a background research is also performed to investigate what are the existing or similar solutions used to solve the respective problem. The current project should be unique and must not have exactly the same approaches used in the researched existing solutions. A work breakdown structure is also carefully designed to specifically clarify which task needs to be done at which period throughout the life cycle of the project. At least a few basic requirements on what the final deliverable should satisfy must also be identified.

Deliverables of Planning Phase: Gantt Chart, Preliminary Report

Phase 2 – Analysis

In this phase, a further analysis on the chosen platform, development tools and other technology during the planning phase is performed. The pros and cons for these elements are analysed with great detail to identify what are the uncertainties and potential risks which may be faced during the development process. The literature review section is also further enhanced with add-on of advantages and disadvantages of existing solutions.

Besides, under this phase, requirement gathering process will also be undergone to better understand the exact requirements and needs of the targeted user i.e. the students. Two techniques were used for this process namely Interview and Survey Method. The reason, flow and purpose of using these techniques are clearly explained in Chapter 4.2. With the results of the survey and interview, a requirement documentation is then written.

Deliverables of Analysis Phase: Requirement Documentation

Phase 3 – Design

In this phase, all the requirements which are defined in the analysis phase is transformed into a complete and detailed system design specifications. All necessary diagrams including Unified Modelling Language (UML) diagrams such as use case, class and activity diagrams and Entity Relationship diagram (ERD) are to be drawn during this phase with the requirements set at the previous phase as references.

Besides, mock-up interfaces planned to be implemented into the application are also to be designed using graphical software and evaluated by supervisor and targeted users before the real development of the interfaces proceed. The mock-up interfaces are designed in a detailed environment especially in terms of the screen size and position of each element found within the interface to ease the development process in the following SDLC stage.

Meanwhile, the basic functionalities of the proposed system are also developed out as prototypes to determine if the functions are feasible and could satisfy the requirements as identified in the previous few phases. From this activity, the potential risks may be faced in the implementation phase could be recognized too.

Deliverables of Design Phase: System Design Specifications, UML Diagrams, ERD Diagrams, Mock-Up Interfaces, Throwaway Prototypes

Phase 4 – Implementation

In this stage, the coding and development process is officially started. All functions and interfaces as planned in the previous phases are to be coded and developed completely with the aid of the design phase final deliverables (system design specifications, UML diagrams, ERD diagrams etc.)

During implementation, there are high risks of facing errors or infeasible situations. However, since agile methodology supports iteration, returning to the design phase and reconstructing the deliverables of this phase to eliminate risks, errors and infeasibility is possible. Otherwise, trials and debugs will continuously be performed until the errors are gone and solutions for the faced problems are to be searched using online resources.

When the application is completely developed and designed which at the same time fulfills all the requirements as listed in the analysis phase, it can then be deployed to the targeted users for testing purposes.

Deliverable of Implementation Phase: Full-function Application with Complete Interfaces

Phase 5 – Testing

As soon as the application is developed, a number of students will be asked to test and evaluate it to check whether it fulfills all their needs and requirements besides satisfying the

criteria of being attractive, categorized, user-friendly, interactive, timely, secure and usable. Since agile method is iterative, returning to the implementation stage and enhancing the functionalities based on tester's comments, suggestions and criticisms are allowed.

The testing phase will only end entirely in the SDLC when the design and development of the application are finalized and no further enhancements, improvements or repairs are needed.

Deliverable of Testing Phase: Evaluation of Implemented Application, Finalized Application

Phase 6- User Manual/Maintenance

The final phase of the life cycle is an on-going phase as long as the application is still being used in the market and has not stopped its services. As soon as the application is completely developed and finalized, a user manual is to be written. The user manual should consist step-by-step procedures for each and every function implemented in the application as a guidance for future users especially those who are new to the application and does not have the basics of using an Android phone.

Under this phase, there will also be a documentation which is specially written for developers, programmers or analysts which are directly involve in the maintenance of this application. With the availability of this document, fixing errors and performing future version updates could be performed easily. Since I am mainly involved in this project, I may also be required to assist in maintaining the entire application.

Deliverables for User Manual/Maintenance Phase: User Manual, Documentation for Maintenance Purposes

3.2 Gantt Chart

Chapter 4: Requirements Documentation

In this chapter, the requirement gathering techniques used will be explained specifically. The explanation will include the reasons these techniques are chosen, who the participants are and what are the expected results. Next, the requirements which are needed for each scope listed in Chapter 1 are listed down. This sub-chapter of scope requirements is important for future references during development phases conducted by programmers, analysts and designers.

4.1 Requirement Gathering Techniques Used

As part of the requirement gathering process, I have used two major techniques, namely the survey method and the interview method.

4.1.1 Survey Method

Since UTARoid is targeted to be used by students and there are huge amount of students which are potentially available to be participants, the survey method is assumed to be the best way to gather their requirements in a short period of time. Being able to avoid too much scheduling effort is also an added advantage since time is a limiting factor. Plus, by using the survey method, I could easily gather specified information on the number of students possessing a smartphone, their current usage and opinion of existing announcement platforms and their expectations on the upcoming UTARoid. At the end of the requirement gathering process, a total of 30 respondents has participated.

The objectives of the survey are as listed as below:

- a) To calculate the percentage of students who possess at least one Android smart phone
- b) To determine how frequent the students access to the existing announcement platforms
- c) To investigate which categories of announcements students view the most
- d) To determine the problems of the existing announcement platforms that could not communicate announcements to students effectively and efficiently
- e) To determine the expectations or requirements of the newly developed UTARoid application

- f) To retrieve ideas and suggestions from students on how they would increase their loyalty towards the newly developed announcement portal

The survey is divided into three sections; Section A – Android Smartphone Users Among Students, Section B – Study on Existing Platforms and Section C – UTARoid Expectations and Requirements. Each section consist of mainly close-ended questions where options are given to respondents to choose from. With the aid of close-ended questions, students are more likely to spend their time to participate the survey and statistical data can be easily tabulated. However, there are still a few questions which are of open-ended type to request for respondent's comments, opinions and suggestions. A sample of the survey question can be viewed in Appendix A.

After the survey forms are gathered, the data tabulation process is started. For each close-ended question, an appropriate chart will be develop and inferences are made from all of these charts. Meanwhile, review is also made towards open-ended questions and a selected few constructive opinions, suggestions or comments will be given top priority while developing the application. The survey results can be viewed in Appendix C.

4.1.2 Interview Method

An one-on-one interview session was also conducted during the requirement gathering process. On the fourth week of the project timeline, I had interviewed the Deputy Dean (Student Development and Industrial Training), Dr. Choo Peng Yin. He was selected as my interviewee as he is one of upper management staff who finds it difficult to communicate announcements especially last minute updates and Industrial Training information to FICT students. He was also the one who had high concerns on this existing problem and had inspired me for the development of this project.

Before the interview session, I had properly planned the questions to be asked which could aid in my objective and requirement definitions. During the session, Dr. Choo has managed to voice out his problems and provide ideas and suggestions related to the needs of communicating messages to the students. Whenever there are topics which are needed to be clarified, questions can be thrown directly and chances of getting a highly detailed information is very high. The overall interview session was conducted in less than one hour in Dr. Choo's office.

4.2 Scope Requirements

As mentioned, requirements for each scope listed in Chapter 1.4 will be explained under this section. This section will be especially useful for programmers, designers and analysts to be used as guidance or references on what needs to be accomplished for each module involved in the project.

a) User Verification

- i. The User Verification scope consists of two primary parts namely the student verification module and staff verification module.
- ii. Under the student verification module, every student needs to perform the sign-in process before being able to proceed to his/her personalized announcement portal.
- iii. Upon entering the UTARoid application, a screen which requires the input of Student ID and Password will be displayed. The Student ID field only accepts numerical values (alphabetical characters are not required) while characters entered in Password field should be displayed in a confidential and secure manner.
- iv. As the Student ID and Password which should be exactly the same with the ones used in UTAR Portal/WBLE are being keyed-in by a student, a matching process will be performed in which the combination of inputted Student ID and Password will be matched with the combination saved in the appropriate database table.
- v. Once a match is found in the database, the particular student will then be granted access to his/her own personalized announcement portal. The Main Screen of the portal should be displayed right after verification process is completed.
- vi. If no match was found, an error message stating invalid combination of Student ID and Password will be displayed.
- vii. No registration for the use of UTARoid is required since the Student ID is already given to each student by default whenever they enrol their studies in UTAR.
- viii. Meanwhile, the staff verification module will be applied on platforms that enable staffs to post, update or delete announcements associated to UTARoid.

- ix. Similarly to the student verification module, an initial screen which requires the input Staff ID and Password will be displayed whenever the mentioned platforms are being accessed. The Staff ID field only accepts numerical values (alphabetical characters are not required) while characters entered in Password field should be displayed in a confidential and secure manner.
- x. As the Staff ID and Password are being keyed-in by a staff, a matching process will be performed in which the combination of inputted Staff ID and Password will be matched with the combination saved in the related database table.
- xi. Once a match is found in the database, the particular staff will then be granted access to the mentioned platform and permission to post, update and delete announcements related to him/her. The Main Screen of the platform should be displayed right after verification process is completed.
- xii. If no match was found, an error message stating invalid combination of Staff ID and Password will be displayed.
- xiii. Both verification modules should have the “Remember Me” functionality in which the previous correctly matcher ID and Password will be automatically entered and displayed during a new user verification session. This function is optional.

b) Auto-Filter and Categorization of Announcement

- i. In the main screen of the announcement, users should be able to tap on the Announcement button to access list of announcements.
- ii. As the Announcement button is tapped, the auto-filter and categorization function should be implemented whereby a list of departments will be displayed within a screen. Auto-filter filters out any departments with no announcements, thus, removing them from the list. Meanwhile, auto-categorization categorizes announcements based on departments.
- iii. As a result, in the department listing screen, only departments with announcements will be listed out and a number label will also be included next to the name of the department indicating the total number of announcements posted by the respective department. For example FICT (5) depicts that there are five announcements posted by FICT officials.

- iv. As an individual clicks on a department, the list of announcements posted by the faculty itself will be displayed in date order (latest being shown on top and old announcements being shown in the bottom of the list)
- v. The same functionality applies in the viewing of upcoming and latest events in UTAR.
- vi. As for time table module, classes are automatically filtered according to subjects that the student registers and sorted according to time (subject with the earliest starting time being displayed on top of the list).
- vii. Announcements for each lecture, tutorial and practical class are also differentiated accordingly to avoid any confusion.

c) Attractiveness and Simplicity

- i. For every displayable of the application, attractive and appropriate multimedia elements should be used which could enhance student's loyalty.
- ii. Fonts chosen should be readable and at the same time, match with the theme and objectives of the application.
- iii. Layout chosen should also match with the theme and objectives of the application.
- iv. Foreground colour and layout of the application should match with each other and promotes readability.
- v. The right icons or images should be used whenever appropriate to replace usage of texts since they could also improve memorability. Icons or images can also be used as links.
- vi. Videos, sounds and animation are optional elements and should be minimized to maintain formality of the announcement portal and to avoid huge memory usage for the application.
- vii. The design of each screen should be as consistent as possible where the usage of texts, colours, images and other media are adequately similar.
- viii. Announcements, bulletins or alerts should be listed in a simpler manner where student could access them with as less number of touches as possible.
- ix. Announcements, bulletins or alerts should be listed from the latest to the oldest.
- x. Settings for the application should be easy to be learned, accessed and manipulated

d) Downloadable Contents

- i. Bus Schedule and Past Year Exams should be downloadable to the Android phone's memory storage for offline usage.
- ii. Downloaded contents will be saved in .pdf format.
- iii. Downloadable contents, however, can still be viewed online via the application.
- iv. During the downloading process, a progress bar is to be included on the screen indicating the progress of the file download (percentage of file successfully downloaded or remaining time/size left)
- v. The application must detect if the phone accessing the application contains a suitable document reader and subsequently prompts out an error message to the user if no right reader is found.

e) Profile Edit & Digital Forms

- i. Signed-in student should be able to edit his/her own profile information via the application anytime without limit as long as there is an Internet connection.
- ii. Information which could be edited include permanent address, mailing address, home number, mobile number, e-mail address and emergency contact details.
- iii. Information which is not editable and could only be viewed include name, race, gender, NRIC No., Student ID No., faculty, course, study session, university e-mail and total credit hours earned.
- iv. Every editable detail should incorporate appropriate input mask. Improper input of details will trigger an error message and the save process will not be allowed.
- v. Before saving new details, a confirmation message will be prompted out first. If save is confirmed, the new details entered by the student will be updated to the database. The same screen will be displayed with newly entered profile information. If save is cancelled, no updates to database will be done and the same screen with existing profile information will be displayed.
- vi. Any other simple registrations or transactions which previously require the use of paper should be made available in the application as digitized form version. Input masks will still be incorporated and completed forms will be sent

directly to the respective database, e-mail or department for processing purposes.

f) Interaction-Enabled Announcements

- i. For each announcement being displayed, there should be a space where both students and staff could leave and view comments as an interactive approach.
- ii. Students and staff who commented will be identified using their own ID.
- iii. A comment posted must not be more than 500 characters.
- iv. Administrators have the right to delete comments than contains harsh or inappropriate use of words or are considered as spams.
- v. The date and time in which a particular comment was posted should be able to be viewed.
- vi. The title of the announcement in which the comment is subjected to will also be listed to avoid confusions.

g) Reward Points System

- i. An announcement of any categories being accessed by a signed-in student will automatically entitle him/her to five reward points.
- ii. Upon receiving the particular reward point, the total reward points accumulated by the student will be recalculated and updated to the database.
- iii. The total reward points the signed-in student has accumulated by far should be displayed in the main menu of the UTARoid application.
- iv. The application should be wise enough to detect which announcement has already been read by the student to avoid any duplication of reward points gained.
- v. A read announcement in the past shall no longer entitle the particular student to gain the five reward points. A short-timed message box will be displayed to the user stating that he/she did not receive any points as the announcement has been read before.
- vi. There should be a different screen which displays the rewards being offered on the particular period. Among the details for each reward which should also be shown include its name, the total points needed to redeem it, remaining stocks, and the individual or party who offered the reward.

- vii. Whenever a student selected a reward to be redeemed, the application will firstly check if he/she has enough reward points.
- viii. If the student has adequate reward points and there are still remaining stocks, a confirmation message will be prompted out asking if the student really wish to exchange his/her earned points with the selected reward. If confirmation is made, student will then be notified the steps to possess the selected reward and his/her total reward points will be deducted accordingly before being updated to the database.
- ix. If the student has not enough points or there is no stock available, an error message will be displayed.

h) Platform to Post or Update Announcement

- i. A website will be designed and developed specifically for administrators or staff to post announcements.
- ii. The first page which will be display upon reaching the website is the user verification page in which individuals are required to enter Staff ID and Password. Invalid verification prohibits user from further accessing the contents of the website.
- iii. As soon as the user successfully login, he/she will be navigated to the main page where options of creating new announcement and view of existing announcements are provided.
- iv. A new announcement should include the name of the individual who created it, announcement category, department which is directly responsible to it and message contents. The name of the individual who created it is automatically filled with the current signed-in staff name and is non-editable.
- v. Before a new announcement is being posted, a confirmation message will be triggered first. If confirmation is made, the announcement will only then be saved in the database and students should now be able to view the new announcement via their own Android phones.
- vi. For each viewable announcement, staff could also view its comments and at the same time leave comment.
- vii. User is allowed to sign out from the website.

(Note that since focus is put more on the UTARoid application, dummy websites are allowed to be used for this project)

Chapter 5: Interface Design

In this chapter, sketches of each and every screens of the UTARoid application will be shown and explained thoroughly. For some screens, the standard sizes and dimensions for included elements such as icons and image buttons will be stated. This chapter can be served as guidelines for future developers of the application besides acting as a storyboard.

5.1 Login Screen

Upon launching the UTARoid application, the Login screen shall appear first where user can enter his/her student ID and password for verification purposes. The following figure shows how the Login screen should look like:-

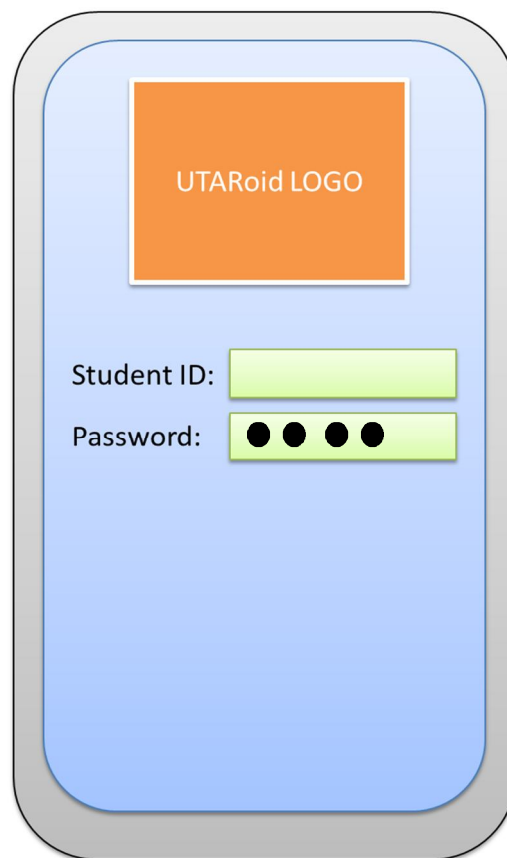


Diagram 5.1.1 Sketch of the Login Screen

The characteristics of the login screen are:-

- UTARoid logo is to be displayed at the top centre of the screen. The dimensions of the logo should be of 165px x 170 px. It is preferable that the logo displayed is same as the application's icon to avoid any confusion.
- The layout of the screen should be simple but attractive.
- Two input fields are to be displayed within the screen; one for the input of Student ID and another for the input of password. The position of the text fields are to be aligned accordingly.
- A label "Student ID" is to be placed next to the input field of Student ID. User should be able to key-in any characters within the field.
- A label "Password" is to be placed next to the input field of Password. User should be able to key-in any characters within the field but all inputted characters will be displayed in bulleted form for security reasons.
- If invalid student ID or password is entered, an error message will be displayed at the centre of the screen as illustrated in the following diagram:-

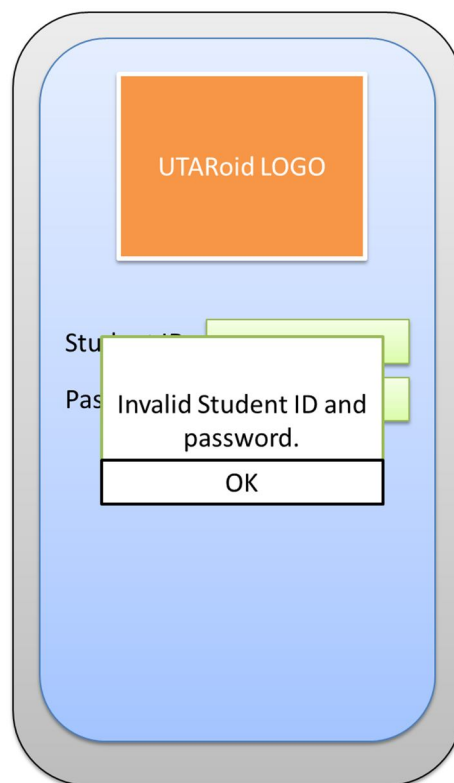


Diagram 5.1.2 Sketch of Login Screen with Error Message Box Displayed

5.2 Home Screen

Upon successful login, student will be navigated into the home screen of the application as illustrated in the figure below:-

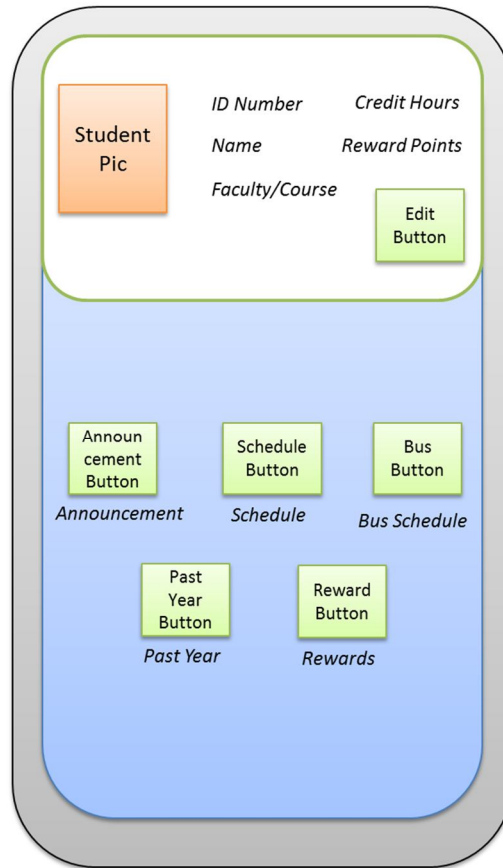


Diagram 5.2.1 Sketch of UTARoid's Home Screen

The characteristics of the home screen are:-

- The screen is divided into two frames namely, the Upper Frame and Lower Frame.
- The Upper Frame consists of the verified Student's ID picture and some general information such as ID Number, Name, Faculty/Course, Credit Hours and Reward Points gained so far. All this information is to be auto-generated upon entering to this screen.
- The font used for the Upper Frame shall be the standard Android font with Small size.
- Extra information can be added by future developers whenever needed.
- There is also an image button in the Upper Frame which leads to the Edit Profile screen upon being clicked.

- Image used for Image Button should be relevant to the theme of “Edit” and shall be in the size of 48x48 pixels.
- The Lower Frame mainly consists of several image buttons; where each of the image button leads to its respective module (as shown in the figure above).
- Image used for Image Button in the Lower Frame should be relevant to its respective theme and shall be in the size of 48x48 pixels.
- A label is to be placed below each image button to serve as guidance to users where the button will lead them to. Font used shall be of Android standard font with Small size.
- If there is any addition of new modules in the future, the image button leading to that module shall always be placed in the Lower Frame.
- The layout of the screen should be simple but attractive. Layouts for both Upper and Lower Frame are highly recommended to be different from each other and shall match accordingly without degrading the attractiveness.

5.3 Edit Profile Screen

As mentioned, students are allowed to edit a few of their personal information via the application itself without the need to fill-in physical forms. Upon tapping on the Edit Profile image button from the Home Screen, a similar screen as shown in the figure on the right should be displayed:-

The diagram shows a mobile application screen for editing a profile. It features a light blue background with rounded corners. The form consists of the following fields:

- Student ID: [Input field]
- Name: [Input field]
- Faculty: [Input field]
- Course: [Input field]
- Year of Study: [Input field]
- Credit Hours: [Input field]
- Reward Points: [Input field]
- E-Mail Address: [Input field]
- Mailing Address: [Input field]

A purple "Save" button is located at the bottom center of the form area.

Diagram 5.3.1 Sketch of the Edit Profile Screen

The characteristics of the edit profile screen are:-

- All general information of the verified student (both editable and non-editable) will be auto-generated and displayed upon entering this screen.
- For non-editable information such as Name, Student ID, Course and Credit Hours, text fields are to be disabled. Another alternative is to only view labels containing the respective data (replacing disabled text field with label)
- For editable information such as e-mail address and mailing address, text fields should be enabled and are allowed to be manipulated by students. Since mailing address, requires a huge text data, multiline text field is suggested to be used.
- Fonts used should be of standard Android font with Medium size (depending on what layout is used and how many data are displayed within a single screen)
- Scrollable Layout is suggested to be used for this screen.
- Background should be simple but attractive and does not interrupt with the student's editing activities. In other words, layout chosen must match with the colours of the foreground and text fields – promoting readability.
- A Save button is to be placed by the bottom end of the screen (centre) where upon finish editing, student can tap on it to save the new edited information.

5.4 UTARoid List Layout

There are many screens/modules within the application where a list of titles is to be displayed to students. The following figure shows an example of how list layout shall be displayed to the user (Announcement Titles):-



Diagram 5.4.1 Sketch of UTARoid List Layout

The characteristics of UTARoid list layout are:-

- The font used for list shall be the standard Android font face and to be adjusted in Medium size for readability purpose.
- Cell-padding for each item should be at least 1px to avoid the list from looking too congested and to enhance readability.
- The background should be simple but yet attractive and does not inhibits readability (matching the foreground colour used).
- Sorting of items depends on the requirements of each module.
- Depending on the functionality/module, each item is either clickable or non-clickable. For clickable items, a specified process such as navigating to another screen or downloading clicked document will start upon being clicked by the user.

The screens which will be using UTARoid list layout include Department Category (categorization of announcements in terms of department), Announcement Titles, Bus Schedules List and Past Year Exams List.

5.5 UTARoid Multiline List Layout

There are certain lists in UTARoid where multiple lines will be displayed for a single row of item as opposed in Section 5.4. The following figure illustrates an example of how UTARoid multiline list layout would look like (List of Classes Registered by Student on Monday):-

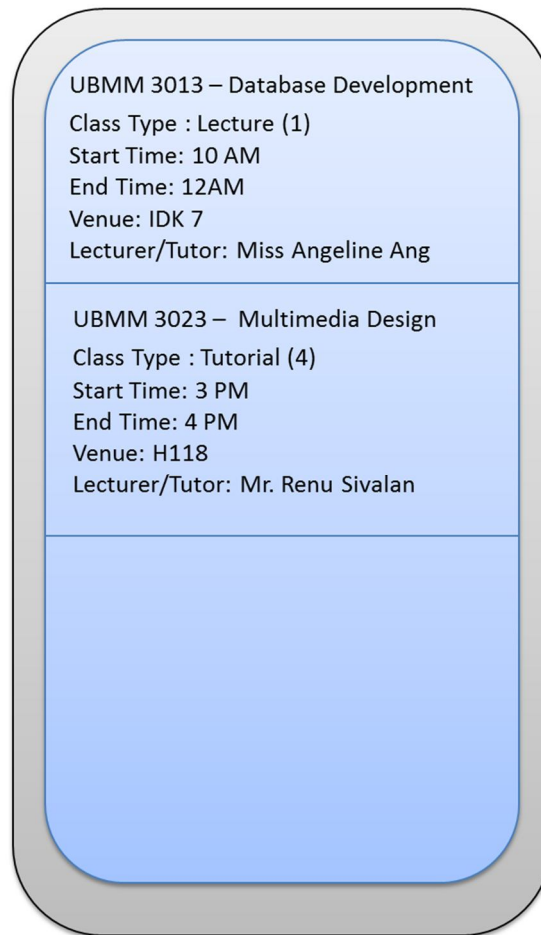


Diagram 5.5.1 Sketch of UTARoid Multiline List Layout

The characteristics of UTARoid multiline list layout are:-

- The font used for list shall be the standard Android font face and to be adjusted in Small size for readability and compact purpose.
- Cell-padding for each item should be at least 1px to avoid the list from looking too congested and to enhance readability.
- The background should be simple but yet attractive and does not inhibits readability (matching the foreground colour used).

- Sorting of items depends on the requirements of each module.
- Depending on the functionality/module, each item is either clickable or non-clickable. For clickable items, a specified process such as navigating to another screen or downloading clicked document will start upon being clicked by the user.

The screens which will be using UTARoid multiline list layout include viewing of Comment List for each announcement, Time Table List, Event List and Reward List. The data to be displayed for each item of each of this list varies according to the requirements of the module.

(Note that for the Comment List, each row of item will also have a tiny “Add” button situated at the right side. Upon tapping on the “Add” button, user will be navigated to the Post Comment screen)

5.6 View Announcement Screen

Upon tapping on an announcement item from the list of announcements, student will be navigated to a screen where the contents of the announcement can be viewed. The design of the screen should be as similar as the figure below:-



Diagram 5.6.1 Sketch of View Announcement Screen

The characteristics of View Announcement screen are:-

- The title of the announcement should be displayed at the top of the screen to prevent confusion.
- The contents of the announcement should be displayed at the centre of the screen.
- The additional details of the announcement such as the author and date posted should be displayed right after the contents end.
- The font which shall be used is standard Android font face with the size adjusted to small. However, the title could be bolded or changed to a larger size.
- The background shall be simple but attractive and does not inhibit readability (matching with the font colour)
- A View Comment button is to be placed at the bottom of the screen to enable student to view comments related to the announcement.

(Note that announcement for specific lecture, tutorial and practical classes will also use similar design as above.)

5.7 Post Comment Screen

As mentioned earlier, to promote interactivity within UTARoid, students are allowed to post comment for any announcement. The Post Comment screen is displayed whenever the “Add” button of the Comment List screen is tapped. The following figure illustrates how the Post Comment screen should be designed:-

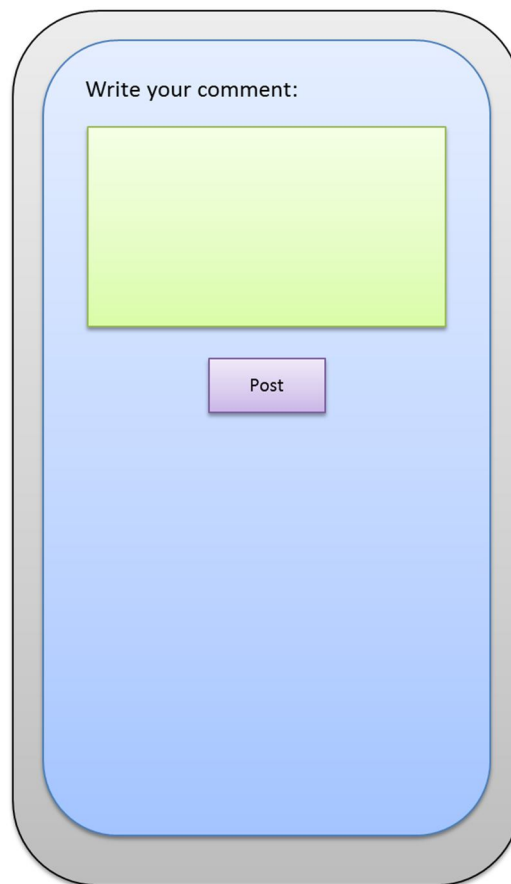


Diagram 5.7.1 Sketch of Post Comment Screen

The characteristics of Post Comment screen are:-

- A label giving instruction on what student should do is displayed at the top of the multiline text box.
- A multiline text box is used for comment input from student.
- A Post button is set directly at the bottom of the multiline text box where upon being clicked, student's inputted comment will be posted to the database and should be viewable under the Comment List of the respective announcement.
- Background shall be simple and attractive but does not mismatch with the foreground.

Chapter 6: System Design

In this chapter, use cases and activity diagrams for the whole UTARoid application will be illustrated.

6.1 Use Case of UTARoid

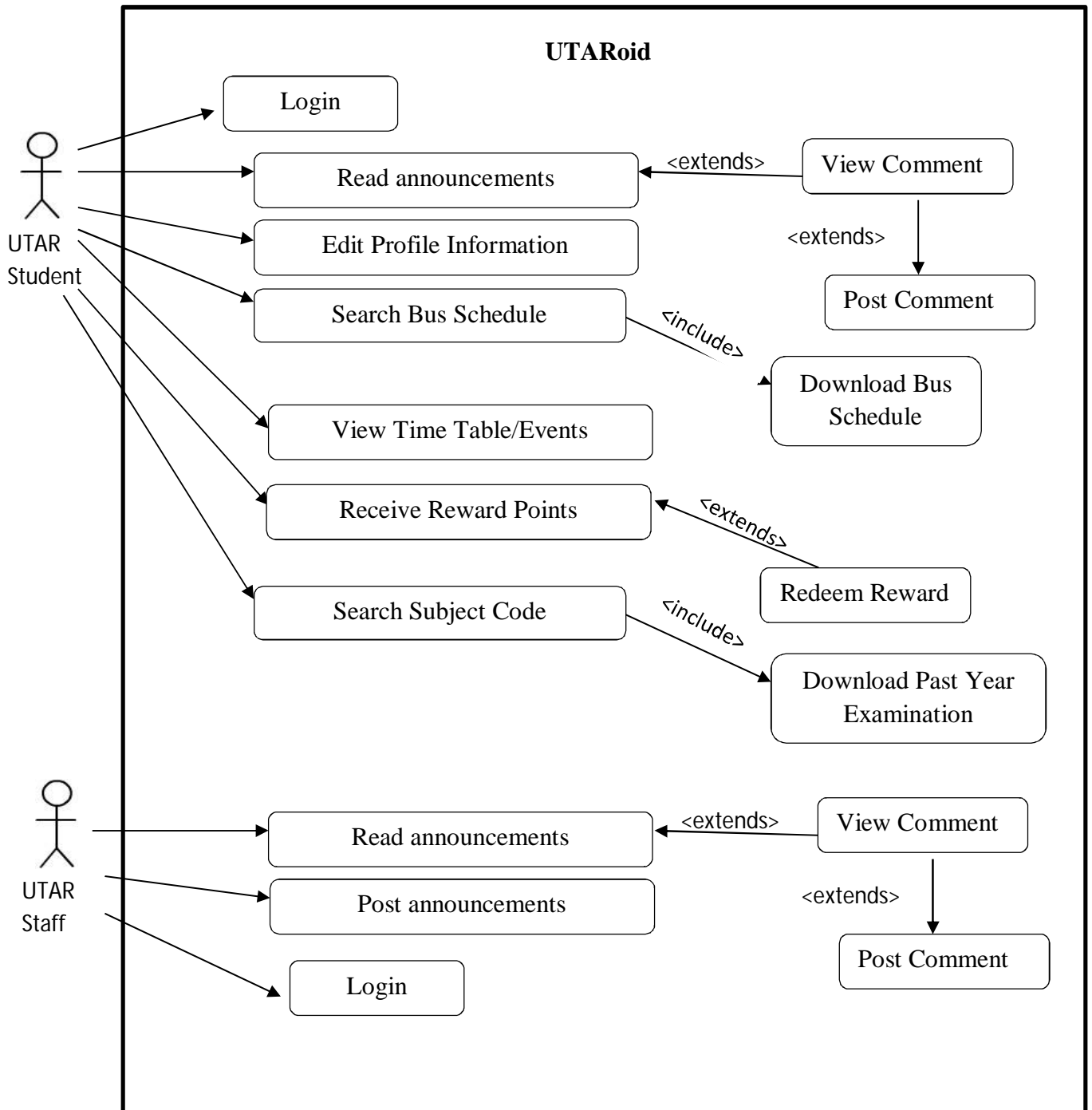


Diagram 6.1.1 Use Case of UTARoid

6.2 Activity Diagrams of UTARoid

6.2.1 Edit Profile Activity Diagram (Student Perspective)

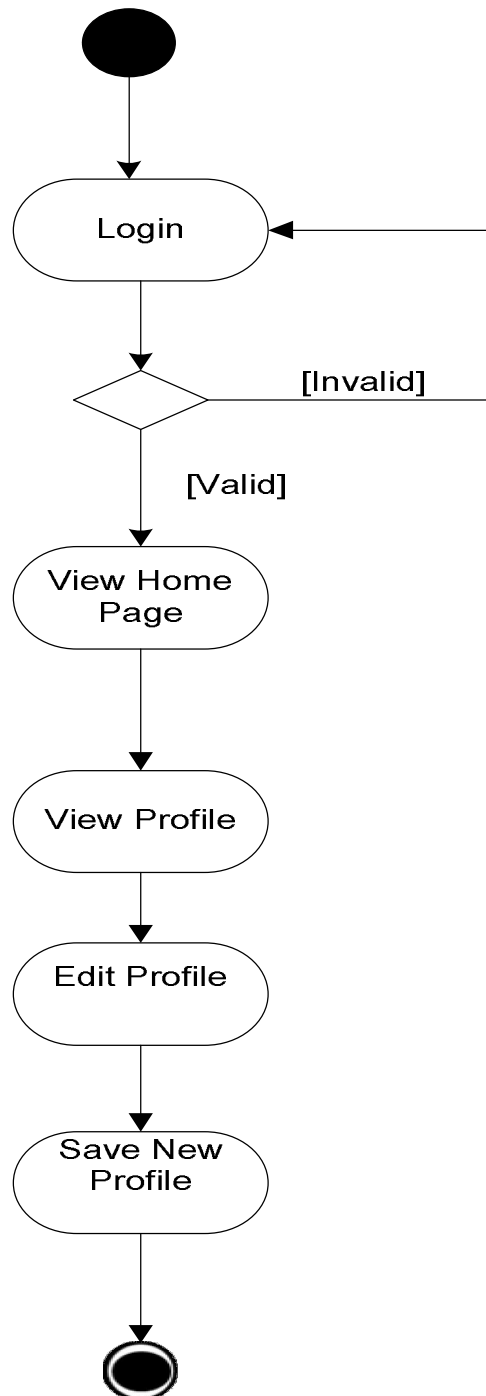


Diagram 6.2.1 Edit Profile Activity Diagram of UTARoid

6.2.2 Announcement & Comment Activity Diagram

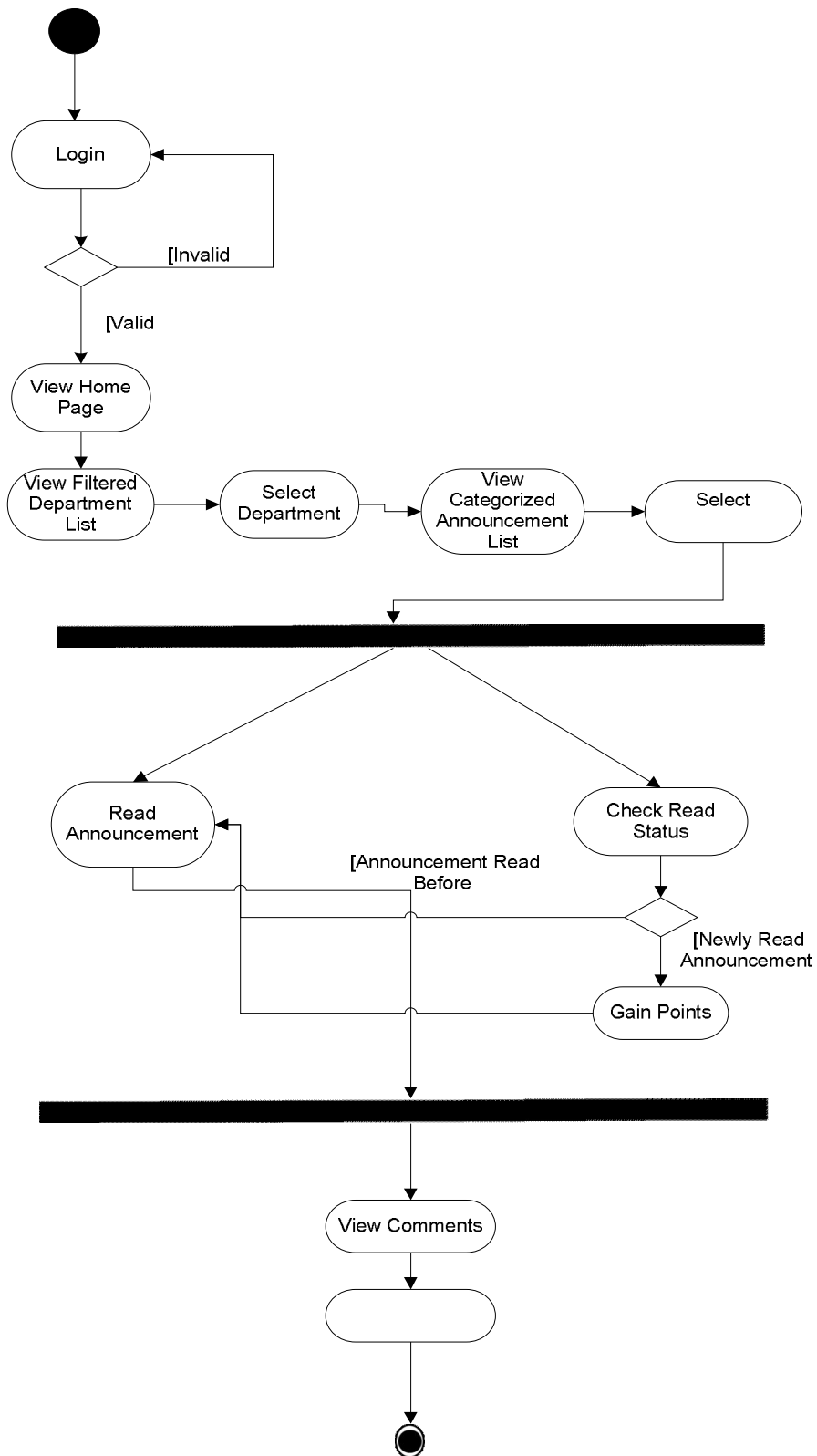


Diagram 6.2.2 Announcement & Comment Activity Diagram of UTARoid

6.2.3 Time Table & Event Activity Diagram

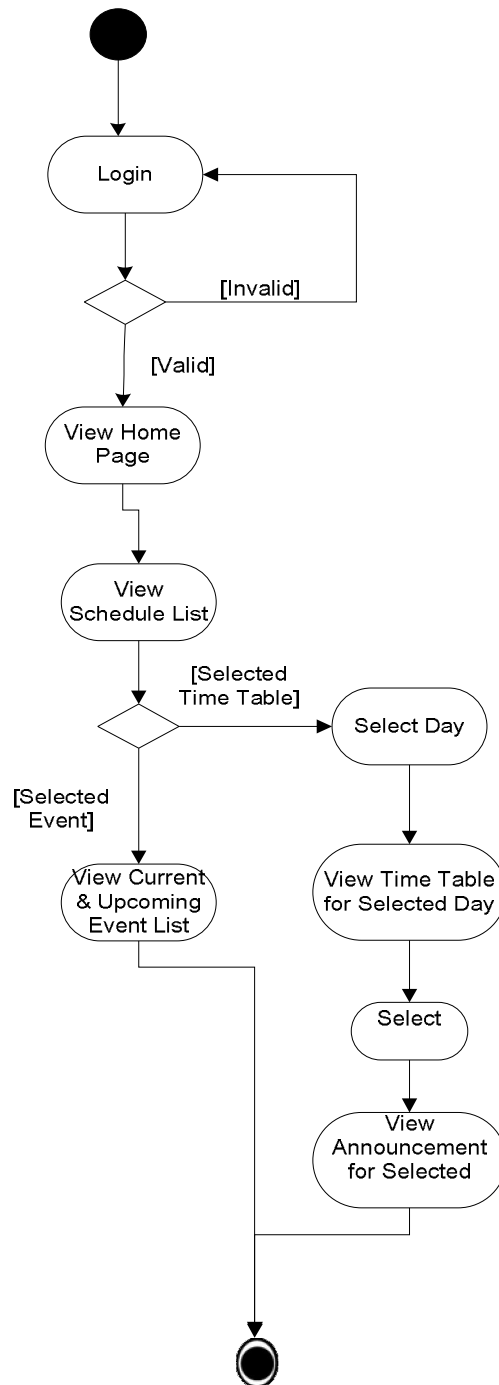


Diagram 6.2.3 Time Table & Event Activity Diagram of UTARoid

6.2.4 Bus Schedule Activity Diagram

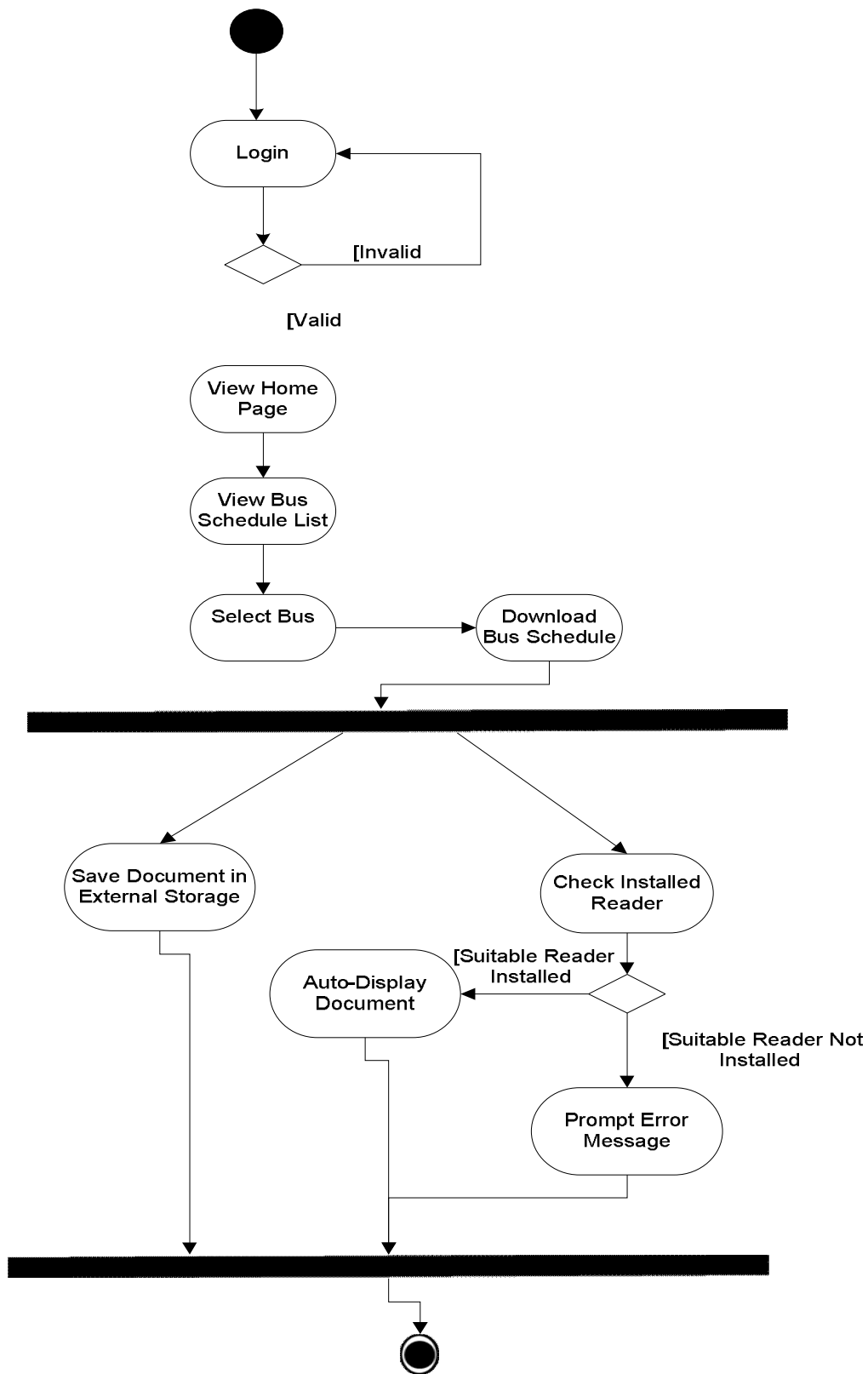


Diagram 6.2.4 Bus Schedule Activity Diagram of UTARoid

6.2.5 Past Year Examination Activity Diagram

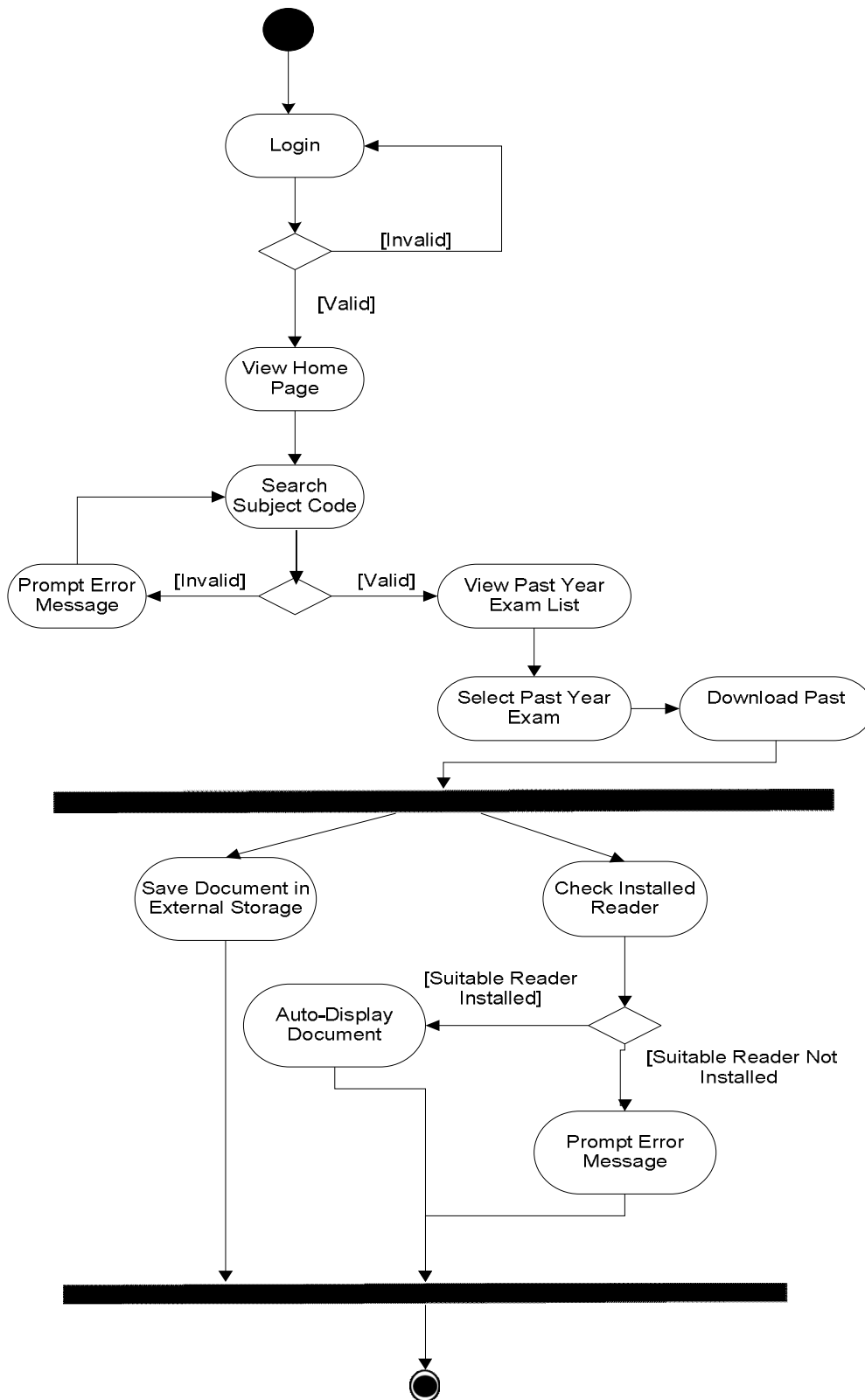


Diagram 6.2.5 Past Year Examination Activity Diagram of UTARoid

6.2.6 Reward Redemption Activity Diagram

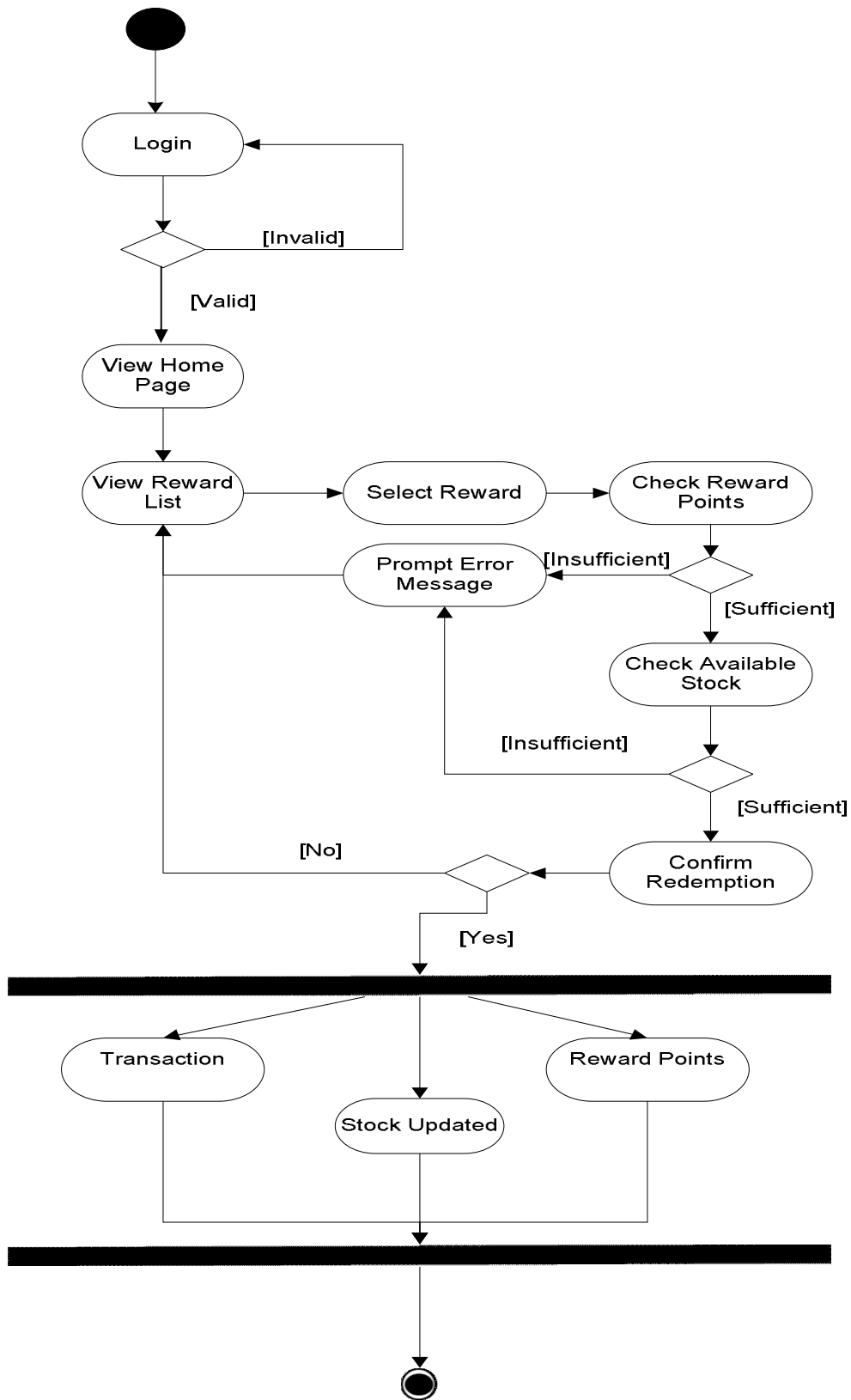


Diagram 6.2.6 Reward Redemption Activity Diagram of UTARoid

Chapter 7: Database Design

In this chapter, an Entity-Relationship Diagram (ERD) will be drawn to show the necessary database tables which shall be created and used for the application to run smoothly. Then, data dictionary table will also be plotted to explain the structure and usage of each column for each table.

7.1 ERD Diagram

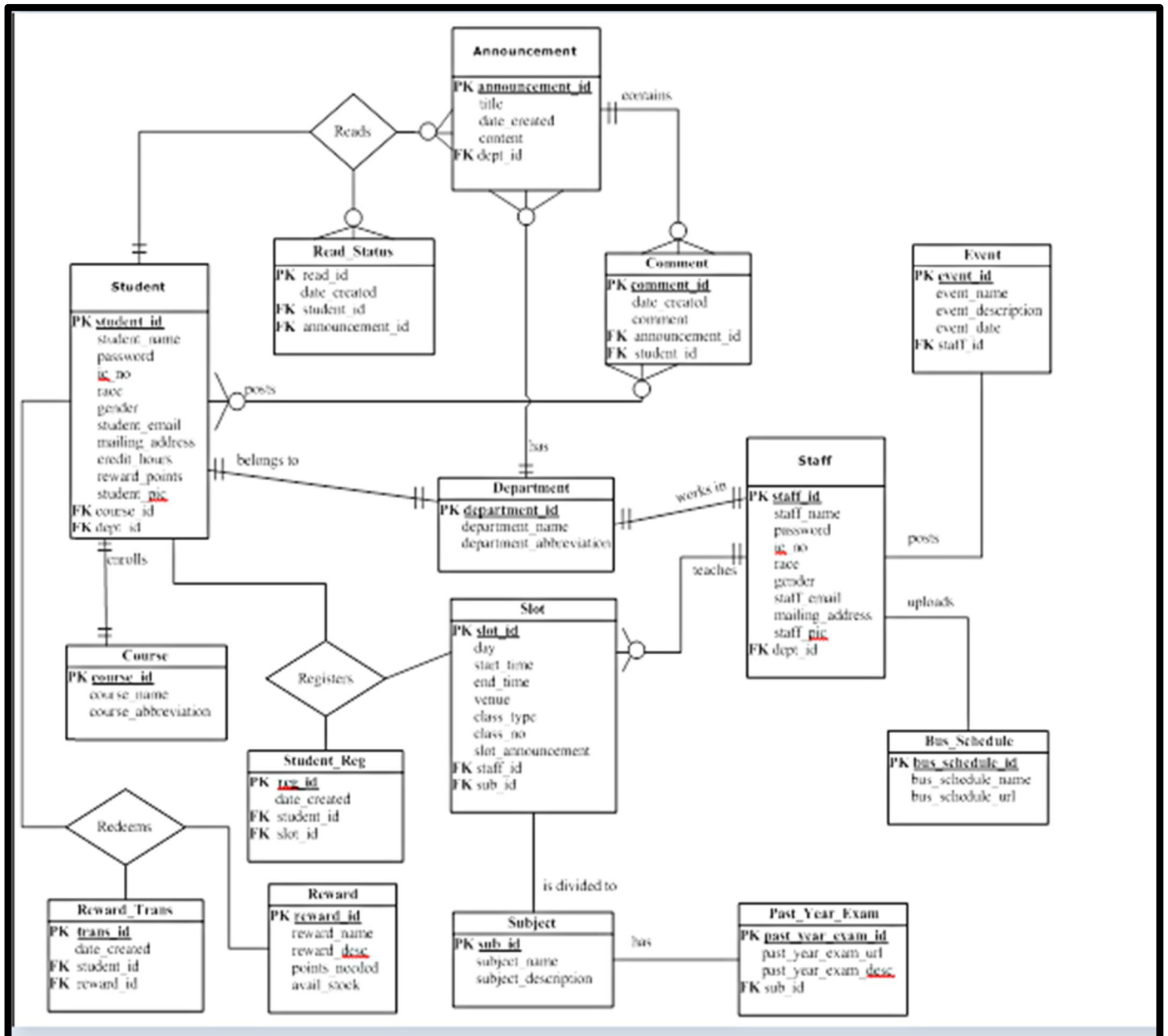


Diagram 7.1.1 ERD Diagram of UTARoid

7.2 Data Dictionary

Table Name: Student

Function: Table that stores profile information of students.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
student_id	Integer	Yes	Yes	-	Attribute that uniquely identify the Student table. Stores the ID number of a student. Required for login.
password	Varchar(25)	Yes	-	-	Stores the password specified by student. Required for login.
student_name	Varchar(100)	Yes	-	-	Stores the full name of student.
ic_no	Char(14)	-	-	-	Stores the Identification Number of a student.
race	Varchar(15)	-	-	-	Stores the race of the student.
gender	Char(6)	-	-	-	Identify the gender of the student.
student_email	Varchar(50)	-	-	-	Stores the student's email address.
mailing_address	Varchar(100)	-	-	-	Stores the mailing address of student.
credit_hours_earned	Integer(3)	-	-	-	Stores credit hours earned by students so far.
reward_points	Integer(5)	-	-	-	Stores reward points earned by students so far.
student_pic	Varchar(255)	-	-	-	Stores the URL of student's profile picture.
dept_id	Char(4)	Yes	-	Yes	Stores the department ID that the student belongs to. References department table.
course_id	Char(4)	Yes	-	Yes	Stores the course ID that the student enrolls in. References course table.

Table 7.2.1 Data Dictionary for Student Table

Table Name: Announcement

Function: Table that stores announcement records (posted by staff and categorized according to the department the staff belongs to)

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
announcement_id	Integer(11)	Yes	Yes	-	Attribute that uniquely identify the Announcement table. Stores the ID number of an announcement.
title	Varchar(255)	Yes	-	-	The title of announcement.
date_created	Date/Time	-	-	-	The date the announcement is posted.
content	Varchar(255)	Yes	-	-	The contents of the announcement.
dept_id	Char(4)	Yes	-	Yes	The department the announcement belongs to. Used for categorization purposes. References department table.

Table 7.2.2 Data Dictionary for Announcement Table

Table Name: Read_Status

Function: Table that stores records of announcements that had been read by students. Used to identify if a particular announcement read by a student has already been read in the past (record found in this table) or is newly read. Newly read announcement entitle the particular student to gain reward points.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
read_id	Integer(9)	Yes	Yes	-	Attribute that uniquely identify the Read_Status table.
date_created	Date/Time	-	-	-	The date the student read the announcement for the first time.
student_id	Integer(7)	Yes	-	Yes	The student ID belonged to the student who has read the announcement. References student table.
announcement_id	Integer(11)	Yes	-	Yes	The announcement ID that refers to the announcement that has been read by the student. References announcement table.

Table 7.2.3 Data Dictionary for Read_Status Table

Table Name: Comment

Function: Table that stores comment records for each announcement (posted by either students or staffs).

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
comment_id	Integer(11)	Yes	Yes	-	Attribute that uniquely identify the Comment table.
comment	Varchar(255)	-	-	-	The contents of the comment posted by either student or staff.
date_created	Date/Time	-	-	-	The date and time the comment is posted.
student_id	Integer(7)	Yes	-	Yes	Student ID that belongs to the student who posted the comment. References student table.
announcement_id	Integer(11)	Yes	-	Yes	Announcement ID that belongs to the announcement where the comment is posted to. References announcement table.

Table 7.2.4 Data Dictionary for Comment Table

Table Name: Department

Function: Table that stores department/faculty records.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
department_id	Char(4)	Yes	Yes	-	Attribute that uniquely identify the Department table.
department_name	Varchar(100)	-	-	-	The full name of the department.
department_abbreviation	Char(8)	-	-	-	The abbreviation used for the department. Abbreviation is used for easy-references/space saving.

Table 7.2.5 Data Dictionary for Department Table

Table Name: Course

Function: Table that stores course records.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
course_id	Char(4)	Yes	Yes	-	Attribute that uniquely identify the Course table.
course_name	Varchar(100)	-	-	-	The full name of the course.
course_abbreviation	Char(8)	-	-	-	The abbreviation used for the course. Abbreviation is used for easy-references/space saving.

Table 7.2.6 Data Dictionary for Course Table

Table Name: Staff

Function: Table that stores profile information of staffs.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
staff_id	Integer(7)	Yes	Yes	-	Attribute that uniquely identify the Staff table. Stores the ID number of a staff. Required for login.
password	Varchar(25)	Yes	-	-	Stores the password specified by staff. Required for login.
staff_name	Varchar(100)	Yes	-	-	Stores the full name of staff.
ic_no	Char(14)	-	-	-	Stores the Identification Number of a staff.
race	Varchar(15)	-	-	-	Stores the race of the staff.
gender	Char(6)	-	-	-	Identify the gender of the staff.
staff_email	Varchar(50)	-	-	-	Stores the staff's email address.
mailing_address	Varchar(100)	-	-	-	Stores the mailing address of staff.
staff_pic	Varchar(255)	-	-	-	Stores the URL of staff's profile picture.
dept_id	Char(4)	Yes	-	Yes	Stores the department ID that the staff belongs to. References department table.

Table 7.2.7 Data Dictionary for Staff Table

Table Name: Event

Function: Table that stores event records (As posted by staff)

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
event_id	Integer(10)	Yes	Yes	-	Attribute that uniquely identify the Event table.
event_name	Varchar(80)	Yes	-	-	Stores the name of the event.
event_description	Varchar(255)	Yes	-	-	The description of the event.
event_date	Date/Time	Yes	-	-	The date of the event.
staff_id	Integer(7)	-	-	Yes	The staff ID that belongs to the staff who posted the event information. References staff table.

Table 7.2.8 Data Dictionary for Event Table

Table Name: Bus_Schedule

Function: Table that stores downloadable bus schedule records.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
bus_schedule_id	Char(4)	Yes	Yes	-	Attribute that uniquely identify the Bus Schedule table.
bus_schedule_name	Varchar(100)	Yes	-	-	The name of the bus schedule which should indicate the campus and semester the schedule relates to.
bus_schedule_url	Varchar(100)	Yes	-	-	The URL of the bus schedule which allows student to directly download the schedule's PDF file in device.

Table 7.2.9 Data Dictionary for Bus Schedule Table

Table Name: Subject

Function: Table that stores subject records.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
sub_id	Char(8)	Yes	Yes	-	Attribute that uniquely identify the Subject table.
sub_name	Varchar(100)	-	-	-	The full name of the subject.
sub_description	Varchar(255)	-	-	-	The description of the subject.

Table 7.2.10 Data Dictionary for Subject Table

Table Name: Slot

Function: Table that stores each subject's available slots and their details.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
slot_id	Integer(11)	Yes	Yes	-	Attribute that uniquely identify the Slot table.
day	Varchar(15)	-	-	-	The day the class falls on.
start_time	Date/Time	-	-	-	The time the class starts.
end_time	Date/Time	-	-	-	The time the class ends.
venue	Char(8)	-	-	-	The venue of the class.
class_type	Char(1)	-	-	-	The type of class. Could be L (Lecture), T (Tutorial) or P (Practical)
class_no	Integer(2)	-	-	-	The class number.
slot_announcement	Varchar(255)	-	-	-	Announcement for the particular class.
staff_id	Integer(7)	Yes	-	Yes	The ID of staff who is the lecturer/tutor of the class. References staff table.
sub_id	Char(4)	Yes	-	Yes	The ID of subject that is taught in the class. References subject table.

Table 7.2.11 Data Dictionary for Slot Table

Table Name: Student_Reg

Function: Table that stores records of subject slots registered by each student.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
reg_id	Integer(11)	Yes	Yes	-	Attribute that uniquely identify the Student_Reg table.
date_created	Date/Time	-	-	-	The date the student registered the subject slot.
student_id	Integer(7)	Yes	-	Yes	The student ID belonged to the student who registered for the subject slot. References student table.
slot_id	Integer(11)	Yes	-	Yes	The slot ID for the class registered by the student. References slot table.

Table 7.2.12 Data Dictionary for Student_Reg Table

Table Name: Past_Year_Exam

Function: Table that stores downloadable past year examination records.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
past_year_id	Varchar(8)	Yes	Yes	-	Attribute that uniquely identify the Past_Year_Exam table.
past_year_desc	Varchar(255)	Yes	-	-	The title/description of the past year exam. Include details such as the subject name and the semester the exam was tested on.
past_year_url	Varchar(255)	Yes	-	-	The URL of the past year exam which allows student to directly download the schedule's PDF file in device.
sub_id	Char(8)	Yes	-	Yes	The ID of subject that is associated with the past year exam. Aids in filtering past year exams based on the subject ID student searches. References subject table.

Table 7.2.13 Data Dictionary for Past_Year_Exam Table

Table Name: Reward

Function: Table that stores redeemable rewards' details.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
reward_id	Char(5)	Yes	Yes	-	Attribute that uniquely identify the Reward table.
reward_name	Varchar(100)	Yes	-	-	The name of the reward.
reward_desc	Varchar(255)	-	-	-	The description of the reward.
points_needed	Integer(5)	Yes	-	-	The points needed to redeem the reward.
avail_stock	Integer(5)	Yes	-	-	The reward's available stock.

Table 7.2.14 Data Dictionary for Reward Table

Table Name: Reward_Trans

Function: Table that stores transactions made by students in the process of redeeming rewards. When a reward is confirmed to be redeemed, this table will automatically be updated with the transaction details.

Field Name	Data Type	No Null	Primary Key	Foreign Key	Description
trans_id	Integer(5)	Yes	Yes	-	Attribute that uniquely identify the Reward_Trans table.
date_created	Date/Time	-	-	-	The date the student redeemed the reward.
student_id	Integer(7)	Yes	-	Yes	The student ID belonged to the student who redeemed the reward. References student table.
reward_id	Char(5)	Yes	-	Yes	The reward ID of the reward redeemed by student. References reward table.

Table 7.2.15 Data Dictionary for Reward_Trans Table

Chapter 8: Implementation and Deployment

This chapter includes descriptions on what software and language are used for the development of UTARoid application. Critical factors that will affect the launching of the application will also be stated out as guidelines and error prevention for future developers. Next, special background features will also be explained with the aid of partial programming codes as references. Lastly, each developed screens for each module will be shown with a list of related PHP files, Java classes, and XML files involved to provide convenience to future developers in the events of updating or manipulating a particular module.

8.1 Language Used

In the development of UTARoid application, the following programming languages are used:-

a) Java Programming Language

As mentioned earlier in Chapter 2: Literature Review, development of Android applications require the foundation, knowledge and skills in using Java as this language is the main programming language supported by Android. Thus, Java concepts such as inheritance, usage of superclass and subclass and abstract and Java syntaxes are applied throughout the development of UTARoid application (using the Eclipse software). The following figure shows a part of the Java environment with basic examples of how main class is initiated and methods are written and called:-

```

package com.example.utaroid;

import java.io.File;

public class busPdf extends Activity {

    TextView tv_loading;
    String dest_file_path = "Bus Schedule.pdf";
    int downloadedSize = 0, totalsize;
    float per = 0;

    private static final String TAG_BUS_SCHEDULE_URL = "bus_schedule_url";

    String download_file_url = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        tv_loading = new TextView(this);
        setContentView(tv_loading);
        tv_loading.setGravity(Gravity.CENTER);
        tv_loading.setTypeface(null, Typeface.BOLD);

        Intent intent = getIntent();
        String pdfurl = intent.getStringExtra(TAG_BUS_SCHEDULE_URL);

        download_file_url = pdfurl;

        downloadAndOpenPDF();
    }

    void downloadAndOpenPDF() {
        new Thread(new Runnable() {
            public void run() {
                Uri path = Uri.fromFile(downloadFile(download_file_url));
                try {
                    Intent intent = new Intent(Intent.ACTION_VIEW);
                    intent.setDataAndType(path, "application/pdf");
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        }).start();
    }
}

```

Diagram 8.1.1: Java Environment in Development Phase

b) Extensible Markup Language (XML)

XML is generally used to manipulate the layout for each screen involved in the UTARoid application. The coding standards used shall comply with the type of Android layouts applied on each screen such as RelativeLayout, AbsoluteLayout or Scrollable Layout (each with their own rules) and also Android specified terms and standards. The following figure shows an partial code example of how the manipulation of a screen layout is done using XML in Eclipse:-

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/crhours"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/homebg2" >

    <TextView
        android:id="@+id/username"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/userid"
        android:layout_below="@+id/userid"
        android:layout_marginTop="17dp"
        android:text="Loading"
        android:textColor="@color/white"
        android:textAppearance="@android:attr/textAppearanceSmall" />

    <TextView
        android:id="@+id/department"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/username"
        android:layout_below="@+id/username"
        android:layout_marginTop="16dp"
        android:text="Loading"
        android:textColor="@color/white"
        android:textAppearance="@android:attr/textAppearanceSmall" />

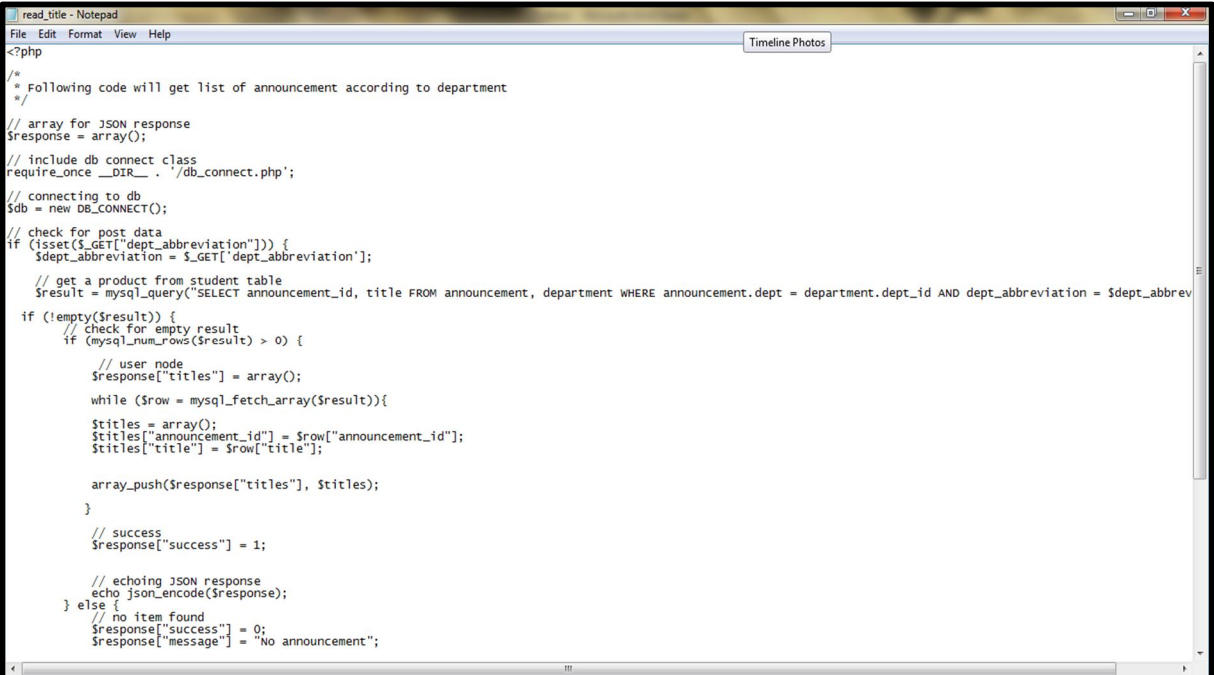
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/userid"
        android:layout_alignBottom="@+id/userid"
        android:layout_alignParentRight="true"
        android:layout_marginRight="48dp"

```

Diagram 8.1.2: Partial XML Coding Example

c) Hypertext Preprocessor (PHP)

As UTARoid also requires a web server to retrieve, update, delete and perform other data manipulation, a scripting language is required. Since PHP is open-source, supports SQL and is easy to be learned especially its coding standards, this scripting language is chosen for certain functionalities of the application. The codes can be written in a simple Notepad as illustrated in the following diagram:-



```
read_title - Notepad
File Edit Format View Help
<?php
/*
 * Following code will get list of announcement according to department
 */
// array for JSON response
$response = array();
// include db connect class
require_once __DIR__ . '/db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// check for post data
if (isset($_GET['dept_abbreviation'])) {
    $dept_abbreviation = $_GET['dept_abbreviation'];
    // get a product from student table
    $result = mysql_query("SELECT announcement_id, title FROM announcement, department WHERE announcement.dept = department.dept_id AND dept_abbreviation = $dept_abbrev");
    if (!empty($result)) {
        // check for empty result
        if (mysql_num_rows($result) > 0) {
            // user node
            $response["titles"] = array();
            while ($row = mysql_fetch_array($result)){
                $titles = array();
                $titles["announcement_id"] = $row["announcement_id"];
                $titles["title"] = $row["title"];
                array_push($response["titles"], $titles);
            }
            // success
            $response["success"] = 1;
            // echoing JSON response
            echo json_encode($response);
        } else {
            // no item found
            $response["success"] = 0;
            $response["message"] = "No announcement";
        }
    }
}
```

Diagram 8.1.3: Example of the Contents of a PHP File Written Using Notepad

d) Structured Query Language (SQL)

For the creation, retrieval, update and deletion of data within the database stored in the local server, SQL is used. Simple query statements were used throughout the development of UTARoid where SELECT statement is used for retrieval, UPDATE statement is used for updating a record and INSERT INTO statement is used to enter new records. Other terms such as ORDER BY are also used depending on the requirement of the module. As all these actions deal with the database and web server, all statements are normally found in the respective PHP files.

8.2 Software Used

Software used for the development, implementation and deployment of UTARoid is listed as follows:-

- a) **Eclipse SDK Version 3.7.2** – Main development platform. Make sure Android SDK is included when being installed for Android application development. Supports Java and XML.
- b) **XAMPP Version 1.7.4** – Package that consist of *MySQL 5.5.8* (performing database operations), *PHP 5.3.5* (for UTARoid dummy website and connection between the application and web server) and *phpMyAdmin* (viewing the database as a whole via browser and manually perform database operations)
- c) **Notepad** – Writing PHP codes

8.3 Critical Factors & Vital Steps in Development

The following are the critical factors or important steps which should be acknowledged by developers during the development of UTARoid. Ignorance of these factors or steps will eventually cause the application to have errors or could not be launched accordingly.

- **Debugging Project Using Physical/Real Device**

Instead of Android Virtual Device (AVD) provided in Eclipse which takes time to launch a project, developers may choose to debug project using own/real smartphone device for better efficiency and look-and-feel. To do so, the following steps are required to be followed:-

- a) Install the appropriate USB driver for the device being used. The links for websites for several original equipment manufacturers (OEMs) can be found in this link <http://developer.android.com/tools/extras/oem-usb.html>.
- b) Note that different smartphone manufacturer requires different type of USB driver being installed in the computer. For an example, in the development of UTARoid, Samsung SIII is used as the device for debugging the project. In order for Samsung devices to be used as a development tool via a computer, Samsung Kies which include the appropriate USB drivers is installed beforehand.

- c) As soon as the appropriate USB driver is installed, connect the device to the computer via a USB hub.
- d) In the Settings option of the device, go to Developer Options and activate USB Debugging. With this option activated, the device will turn into debug mode whenever it is connected to the computer, allowing debugging activities.
- e) Run your project. The application shall be installed and launched in the device.

- **Permission for Application to Access Internet**

UTARoid requires Internet access for functions that requires retrieval and manipulation of data stored in the web server. By default, any Android application will never allow access to the Internet unless given permission to do so. To grant an application to use access the Internet, include this line of code right after the <manifest> tag in the project's manifest XML file (in this case, UTARoid Manifest):-

```
<uses-permission android:name="android.permission.INTERNET" />
```

- **Permission for Application to Save/Overwrite Downloaded Item in Phone's Storage**

Some modules within UTARoid require saving of downloaded documents such as Bus Schedule and Past Year Exam into the phone's memory. By default, Android application will not allow manual saving and overwriting of items in the external storage of a device/phone unless permitted. To grant an application to save or overwrite a downloaded item into the phone, include this line of code right after the <manifest> tag in the project's manifest XML file (in this case, UTARoid Manifest):-

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

- **Java Activities in Manifest**

Even if the Java classes which serve as activities for the application do not contain any errors, application may fail to be launched as the classes have yet to be initiated or permitted in the manifest XML file of the project. To allow an activity written in a Java class to be executed by the application, the <activity> tag must be included in the manifest file as follows (the block of code in this example allows classAnnouncement.java to be executed when called in the application):-

```
<activity
  android:name="com.example.utaroid.classAnnouncement"
  android:label="@string/title_activity_login"
  android:windowSoftInputMode="adjustPan">

</activity>
```

Within the <activity> tag, developers can add in other properties such as label and windowSoftInputMode as appropriate depending on the module requirements.

- **Placement of PHP Files and Other Server Items**

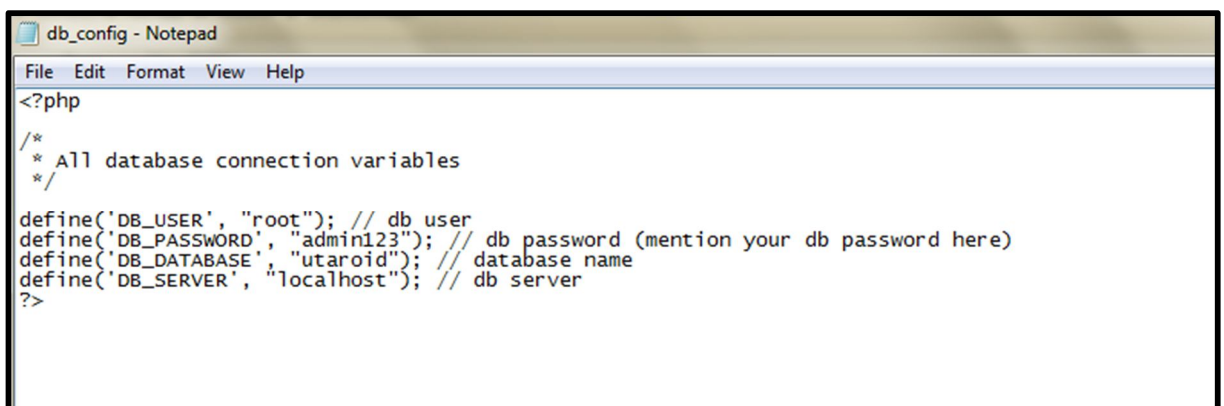
Upon successful installation of XAMPP package into the computer, a folder named 'htdocs' could be found within the xampp folder which is created in C:// by default. All PHP files and items such as images and documents which are to be accessed by the mobile application shall be stored in this 'htdocs' folder.

Given a scenario where a folder named 'Schedule' which contains a PDF file named "Bus2013" is created in the 'htdocs' folder. To access to this file, the proper URL which should be used by the developer is localhost/Schedule/Bus2013.pdf. *(Note that localhost should be changed to other IP address depending on the network the system is connecting to or the position of the web server)*

- **Connection to SQL Database**

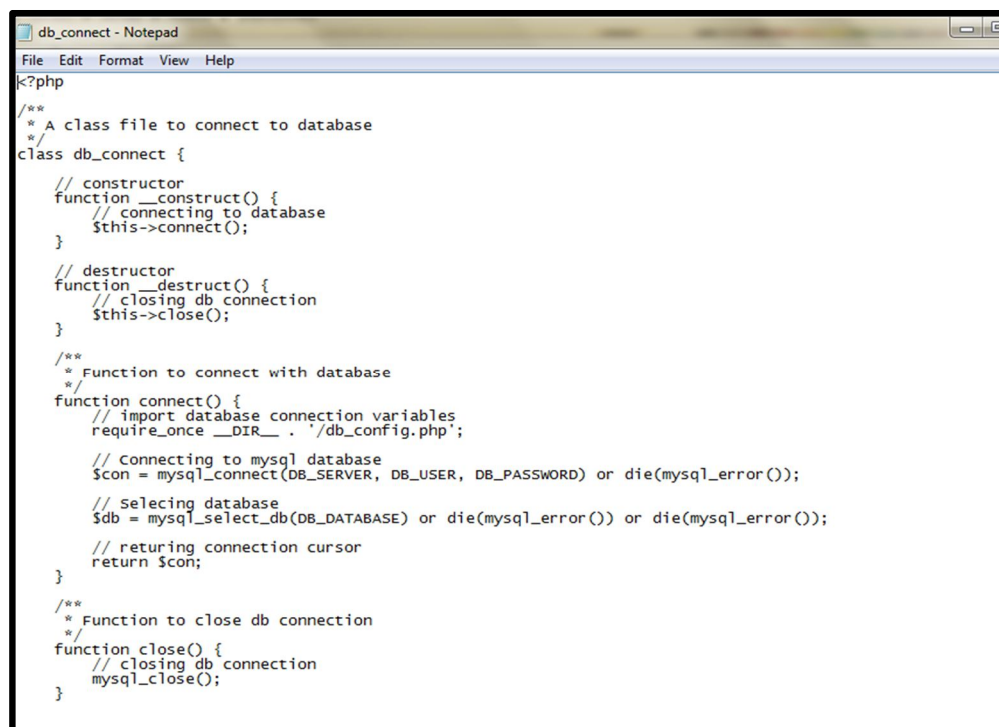
Before being able to retrieve or manipulate data in the specified SQL database, a connection must be made first which requires entry of username and password. A PHP file should be created in which the codes within allows connection from various sources especially UTARoid to the specified web server when accessed. The script should contain the right username and password for successful operations. In the development of UTARoid, two PHP files are created instead where one file (db_config.php) stores the variables (database name, username and password) whereas the other(db_connect.php) makes connection to the database based on the variables written.

db_config.php



```
db_config - Notepad
File Edit Format View Help
<?php
/*
 * All database connection variables
 */
define('DB_USER', "root"); // db user
define('DB_PASSWORD', "admin123"); // db password (mention your db password here)
define('DB_DATABASE', "utaroid"); // database name
define('DB_SERVER', "localhost"); // db server
?>
```

db_connect.php



```
db_connect - Notepad
File Edit Format View Help
<?php
/**
 * A class file to connect to database
 */
class db_connect {
    // constructor
    function __construct() {
        // connecting to database
        $this->connect();
    }
    // destructor
    function __destruct() {
        // closing db connection
        $this->close();
    }
    /**
     * Function to connect with database
     */
    function connect() {
        // import database connection variables
        require_once __DIR__ . '/db_config.php';
        // connecting to mysql database
        $con = mysql_connect(DB_SERVER, DB_USER, DB_PASSWORD) or die(mysql_error());
        // selecting database
        $db = mysql_select_db(DB_DATABASE) or die(mysql_error()) or die(mysql_error());
        // returning connection cursor
        return $con;
    }
    /**
     * Function to close db connection
     */
    function close() {
        // closing db connection
        mysql_close();
    }
}
```

- **Links to Access PHP Files or Other Items in Java Classes**

Most of the Java classes in UTARoid require the execution of processes specified in PHP files especially for retrieval and manipulation of data. The URL stated in the Java class must be accurate or else errors will pop up.

```
// url to get reward list
private static String url_reward = "http://192.168.43.71/afterconnect/read_reward.php";

// url to update points and transaction table upon successful redeem
private static String url_update_points = "http://192.168.43.71/afterconnect/update_points.php";
private static String url_update_trans = "http://192.168.43.71/afterconnect/update_trans.php";
```

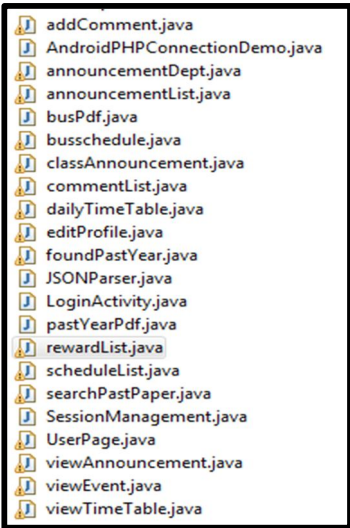
In the development of UTARoid, each URL accessing to a single file will be stored as a String type variable. Whenever there is a need to access the file, developer would just need to code the variable name. Once accessed, the operations within the file will be executed.

(Note that the IP address as in circled in red shall be changed as required depending on the network the computer is connected to or the existing server the activity wish to connect. During the development of UTARoid, the IP address of the device tethering hotspot is used (192.168.43.71)).

- **Naming Convention**

The naming convention used for variables and file names should be of camel casing as shown below:-

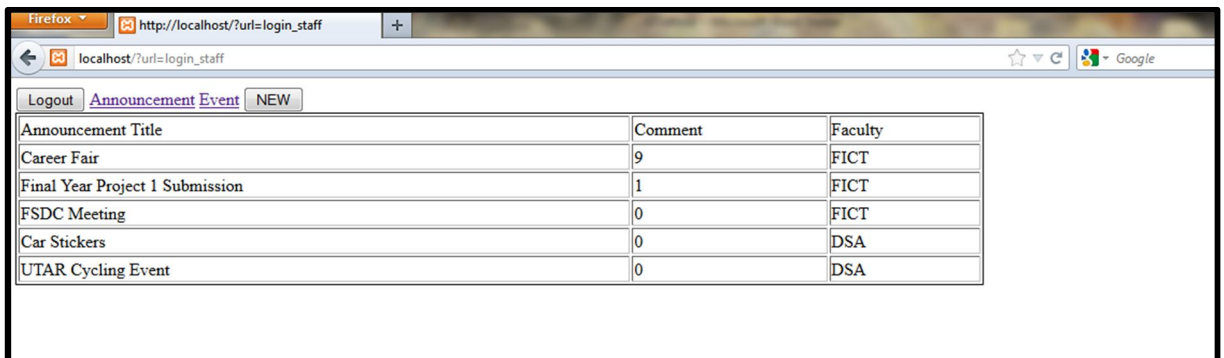
```
String rwdPoints, student_id, rewardId;
int rewardPoints, neededPoints;
```



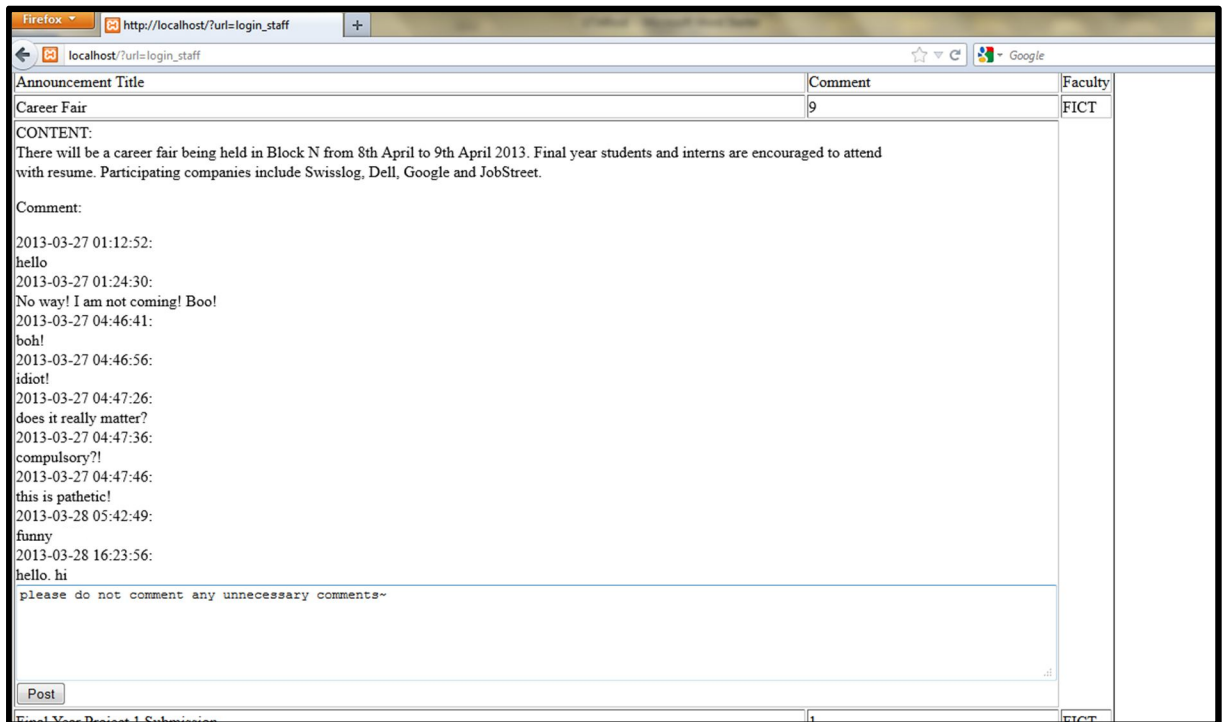
- **Posting New Announcement/Comment Using Dummy Website**

As stated earlier, a dummy website will be created as a means for staff to post announcements which later could be viewable via the UTARoid application. The following are the steps for posting new announcement or comment using the dummy website:-

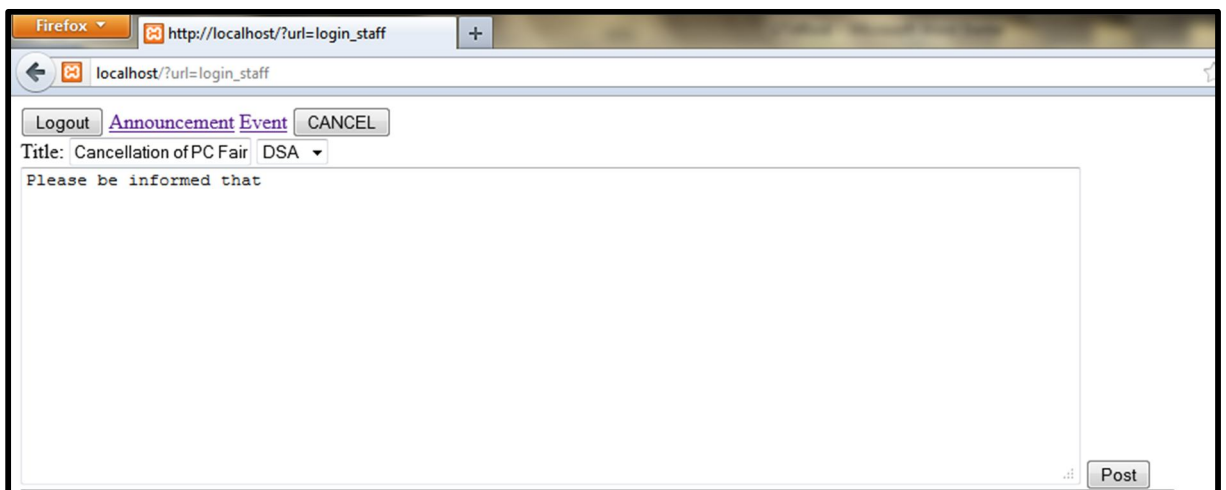
- i. After downloading XAMPP package, make sure the contents of the htdocs folder in the newly installed xampp folder is exactly the same as the contents of the htdocs folder given in the softcopy. Missing of any file could cause the dummy website not to work accordingly.
- ii. Open browser and type the following URL: http://localhost/?url=login_staff
- iii. Enter username and password. The combination must match with a single record from the 'staff' table in the database.
- iv. Upon successful login, a web page similar to the figure below will be displayed:-



- v. When a user clicks on an announcement title, the contents and the list of comments for the selected announcement will be displayed as shown in the figure below:-



- vi. Upon viewing the content and comments of the selected announcement, user can choose to post new comment. After typing new comments on the displayed multi line field, user would just need to click the Post button.
- vii. For posting a new announcement, user would need to click on the New button displayed at the top of the screen. A screen similar to the figure below shall be displayed:-



- viii. Fill in the required fields such as the title and the contents of the announcement.
- ix. Select the department that you belong to or that relates to the announcement before clicking on the “Post” button.
- x. As validation is successfully done, the newly posted announcement will be saved in the database.
- xi. As a student logs into the UTARoid application, the newly posted announcement shall be viewable and new comments can be added in.
- xii. The same procedure applies to posting of new Events. On the Home Screen of the dummy website, user would just need to click on the Event link.

8.4 Special Features

In the development of UTARoid, the following features/techniques/methods are used in order for the program to run as expected:-

i. JSON

JSON as mentioned in Chapter 2: Literature Review, is a method to access information in an organized, effective and efficient way. In the development of UTARoid, JSON had been used in almost every Java classes as a way to retrieve records generated by SQL statement executed in a PHP file. Besides retrieving, the processes of creating and updating data are also aided by JSON concept.

The following steps show how JSON is executed in an attempt to retrieve data from the web server and displayed via the mobile application (this example is taken in the case of retrieving a verified student’s profile information and displayed in the home page of the application):-

- a) Firstly, there should be a block of codes where parameters which are needed for the execution of script in the PHP file are stored in pairs format (as shown below). Then, a HTTPRequest will be made to connect to the respective PHP file via a network before sending in the parameters. The HTTPRequest made should also specify if the “GET” or “POST” method shall be used to send the parameters.

```
List<NameValuePair> params = new ArrayList<NameValuePair>(1);
params.add(new BasicNameValuePair("student_id",name));
JSONObject json = jparser.makeHttpRequest(url_student, "GET", params);
```

The List<NameValuePair> *params* store a list of parameters in pair format of property:name, property:value. As shown above, since only student ID is required as a parameter, the *params* list is only added with a single parameter where “student_id” represents the property:name and the variable “name” acts as the property:value. (Note that the variable “name” has been initiated at the beginning of the program storing a value of “0902861”.)

A JSONObject is then initiated where its task is to parse the data stored in *params* before making a HTTPRequest using the GET method to the specified URL (the String variable, *url_student* has been initiated in the beginning of the program with the value of http://192.168.43.71/afterconnect/read_details.php). As connection is successfully made, the parsed parameter will be sent to the PHP file for further process.

- b) As the required parameter is received by the accessed PHP file, the execution of the remaining scripts continues.



```
read_details - Notepad
File Edit Format View Help
<?php
/*
 * Following code will get student details
 * A student is identified by student id (student_id)
 */
// array for JSON response
$response = array();
// include db connect class
require_once __DIR__ . '/db_connect.php';
// connecting to db
$db = new DB_CONNECT();
// check for post data
if (isset($_GET["student_id"])) {
    $student_id = $_GET["student_id"];
    // get a product from student table
    $result = mysql_query("SELECT student_id, student_name, credit_hours_earned, reward_points, dept_abbreviation, course_abbreviation, student_pic FROM student, depart");
    if (!empty($result)) {
        // check for empty result
        if (mysql_num_rows($result) > 0) {
            $result = mysql_fetch_array($result);
            $student = array();
            $student["student_id"] = $result["student_id"];
            $student["student_name"] = $result["student_name"];
            $student["credit_hours_earned"] = $result["credit_hours_earned"];
            $student["reward_points"] = $result["reward_points"];
            $student["dept_abbreviation"] = $result["dept_abbreviation"];
            $student["course_abbreviation"] = $result["course_abbreviation"];
            $student["student_pic"] = $result["student_pic"];
            // success
            $response["success"] = 1;
            // user node
            $response["student"] = array();
            array_push($response["student"], $student);
            // echoing JSON response
            echo json_encode($response);
        } else {
            // no student found
            $response["success"] = 0;
        }
    }
}
```

In this example, after getting the student ID parameter, a SQL statement of retrieving the required information that matches the parameter given is executed. If record(s) is found in the database, the data for each column of the found record will be stored in an array and the whole array storing the results will be echoed back to the Java class

that had accessed the PHP file. If no record was found or there is an error, a Failure response will be echoed back instead. (*Refer read_details.php in htdocs folder for viewing of full code*)

- c) The array of results echoed back from the PHP file is then stored in the JSONArray in the exact format. The value for each property is then retrieved and stored in a JSONObject, separated by property name (with the format similar to Diagram 2.6.1).

```
try{
int success = json.getInt(TAG_SUCCESS);

if (success == 1) {
// student found
// Getting Array of Products
student_array = json.getJSONArray(TAG_STUDENT);

JSONObject c = student_array.getJSONObject(0);

// Storing each json item in variable
id = c.getString(TAG_STUDENT_ID);
namer = c.getString(TAG_STUDENT_NAME);
crhours = c.getString(TAG_CREDIT_HOURS_EARNED);
rwdpts = c.getString(TAG_REWARD_POINTS);
department = c.getString(TAG_DEPT_ID) + "-" + c.getString(TAG_COURSE_ID);
picurl = c.getString(TAG_STUDENT_PIC);

}}
```

Each value is then retrieved and stored in the Java class defined variable by using the getString() method. The parameter used in the getString() method must be a String variable with the name starting with "TAG_" followed by the respective property name as a technique to match which type of property retrieved that the user is referring to.

- d) Upon storing the retrieved values in the defined variables, the values could now be used throughout the program for various purposes such as displaying the retrieved values to the user or to use the values for further mathematical operations.

ii. Downloading File Mechanism

The following block of codes show how a file is being downloaded from a specified URL before being saved into the phone external storage/specified directory:-

```
File downloadFile(String download_file_path) {
    File file = null;
    try {

        URL url = new URL(download_file_path);
        HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();
        urlConnection.setRequestMethod("GET");
        urlConnection.setDoOutput(true);
        urlConnection.connect();

        File SDCardRoot = Environment.getExternalStorageDirectory();

        file = new File(SDCardRoot, dest_file_path);
        FileOutputStream fileOutput = new FileOutputStream(file);

        InputStream inputStream = urlConnection.getInputStream();

        totalsize = urlConnection.getContentLength();
        setText("Starting PDF download...");
        // create a buffer...
        byte[] buffer = new byte[1024 * 1024];
        int bufferLength = 0;

        while ((bufferLength = inputStream.read(buffer)) != -1) {
            fileOutput.write(buffer, 0, bufferLength);
            downloadedSize += bufferLength;
            per = ((float) downloadedSize / totalsize) * 100;
            setText("Total PDF File size : "
                + (totalsize / 1024)
                + " KB\n\nDownloading PDF " + (int) per
                + "% complete");
        }

        fileOutput.close();
        inputStream.close();
        setText("Download Complete. Open PDF Application installed in the device.");
    } catch (final MalformedURLException e) {
```

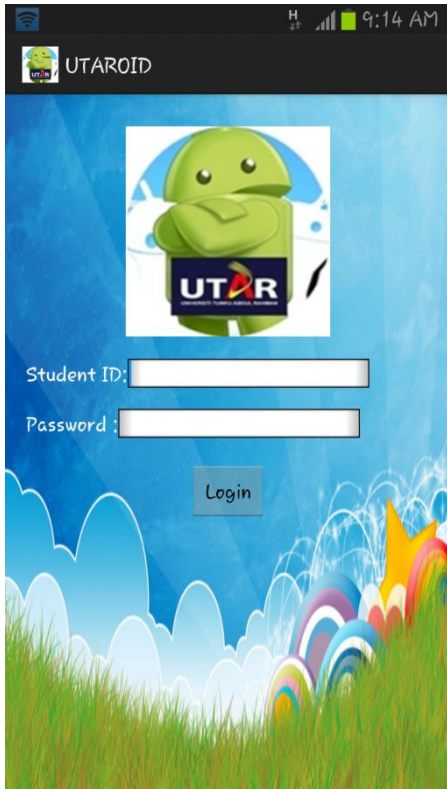
As soon as the download is complete, the program will then check if the device used to launch the application is installed with a suitable reader application which could read the downloaded file. If no suitable reader is to be found, an error message will be displayed telling the particular user to install an appropriate reader in order to view the file. The block of codes that execute this function is as follows:-

```
void downloadAndOpenPDF() {
    new Thread(new Runnable() {
        public void run() {
            Uri path = Uri.fromFile(downloadFile(download_file_url));
            try {
                Intent intent = new Intent(Intent.ACTION_VIEW);
                intent.setDataAndType(path, "application/pdf");
                intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                startActivity(intent);
                finish();
            } catch (ActivityNotFoundException e) {
                tv_loading
                    .setError("PDF Reader application is not installed in your device");
            }
        }
    }).start();
}
```

8.5 Screens and Files Involved

The following list shows the screens of UTARoid which are implemented and deployed together with the PHP files, Java classes and XML files involved for each of the screen. This subchapter serves as a guideline or directory to future developers whenever there is a need to make amendments to the screens or the functionalities behind the screen.

i. Login Screen



Functionality:

Provides fields to user for input of Student ID and Password. Verification of login credentials will be performed here.

PHP Files Involved:

- check.php

Java Files Involved:

- LoginActivity.java
- SessionManagement.java

XML File:

- activity_login.xml

ii. Home Screen



Functionality:

Screen displayed upon successful login. Displays basic information of student in the Upper Frame and selectable options in the Lower Frame.

PHP Files Involved:

- read_details.php

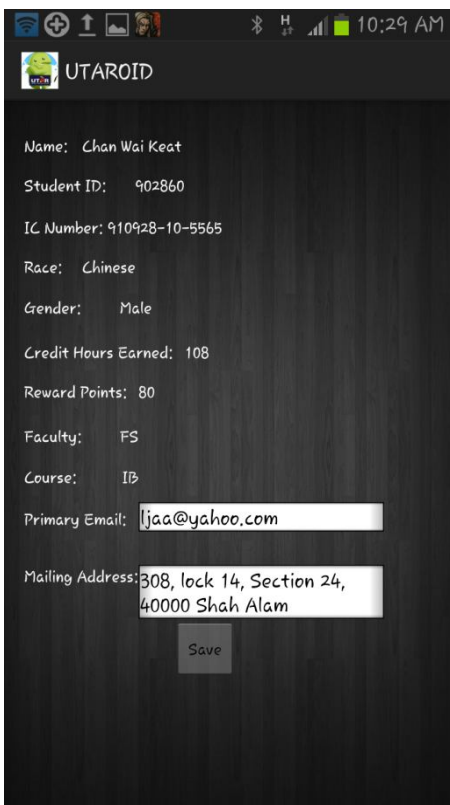
Java File Involved:

- UserPage.java
- SessionManagement.java

XML File:

- homepage.xml
-

iii. Edit Profile Screen



Functionality:

View all information of a student and allows profile editing.

PHP Files Involved:

- read_details2.php
- update_details.php

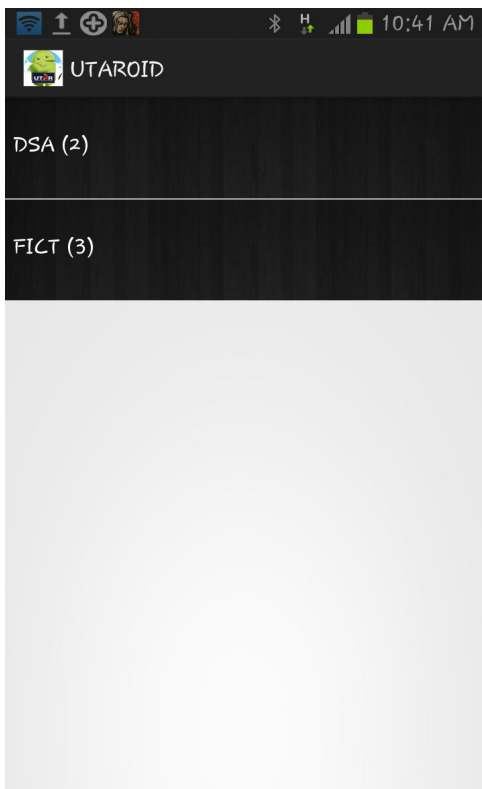
Java File Involved:

- editProfile.java

XML File:

- editprofile.xml

iv. Department List Screen



Functionality:

Listing of departments which have posted announcements. Number in column indicates total announcement by that department. (Auto-categorization and auto-filter functions)

PHP Files Involved:

- read_department.php

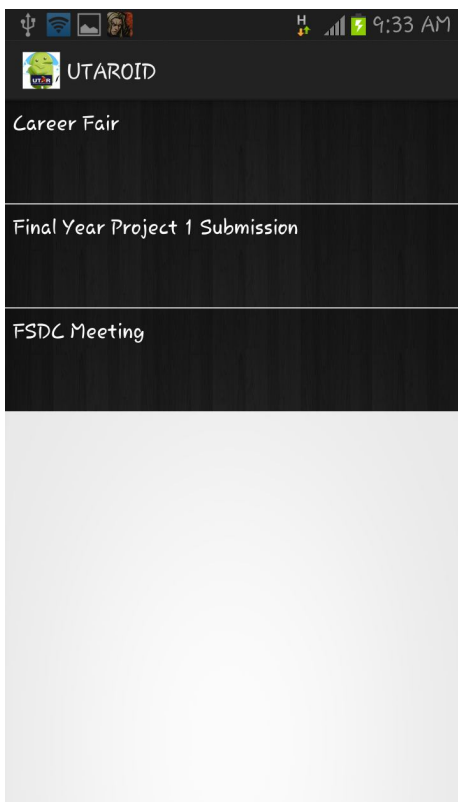
Java File Involved:

- announcementDept.java

XML File:

- departmentitem.xml

v. Announcement List Screen



Functionality:

Listing of announcement titles based on the selected department from Department List Screen.

PHP Files Involved:

- read_title.php

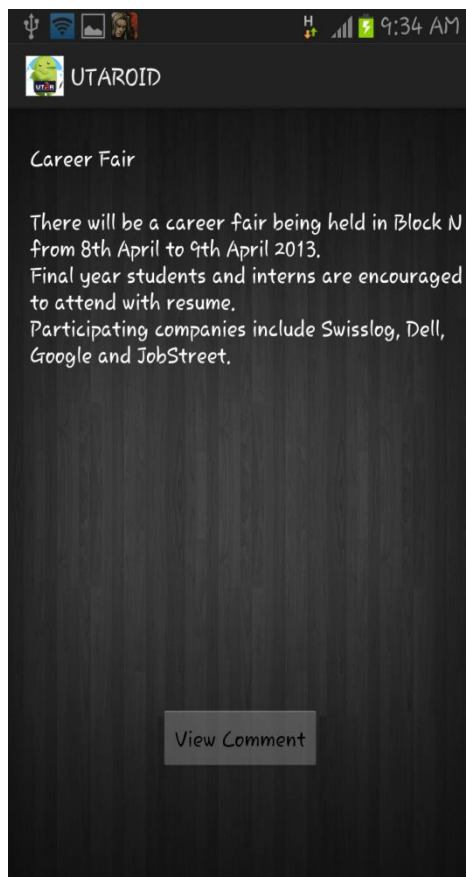
Java File Involved:

- announcementList.java

XML File:

- announcementitem.xml

vi. Announcement Content Screen



Functionality:

View the content of the announcement based on the selected announcement title from Announcement List Screen. Allow rewards point gain for newly read announcement.

PHP Files Involved:

- read_announcement.php
- read_status.php
- gain_points.php
- update_status.php

Java File Involved:

- viewAnnouncement.java

XML File:

- view_announcement.xml

vii. Comment List Screen



Functionality:

List the comments of the announcement viewed. Displayed when the View Comment button from Announcement Content screen is tapped.

PHP Files Involved:

- read_comment.php

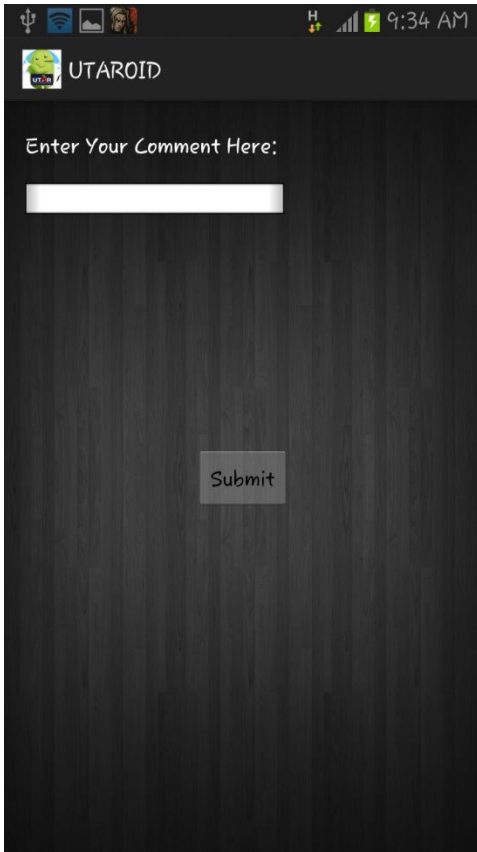
Java File Involved:

- commentList.java

XML File:

- commentitem.xml

viii. Post Comment Screen



Functionality:

Allows user to post comment for selected announcement. Displayed when Add button from Comment List screen is tapped.

PHP Files Involved:

- update_comment.php

Java File Involved:

- addComment.java

XML File:

- commentspace.xml

ix. Class List Screen (Displayed upon selecting a day from range of Monday – Friday)



Functionality:

Allows user to view list of registered class for the selected day. Each item in list comes with all necessary details of the class.

PHP Files Involved:

- read_time_table.php

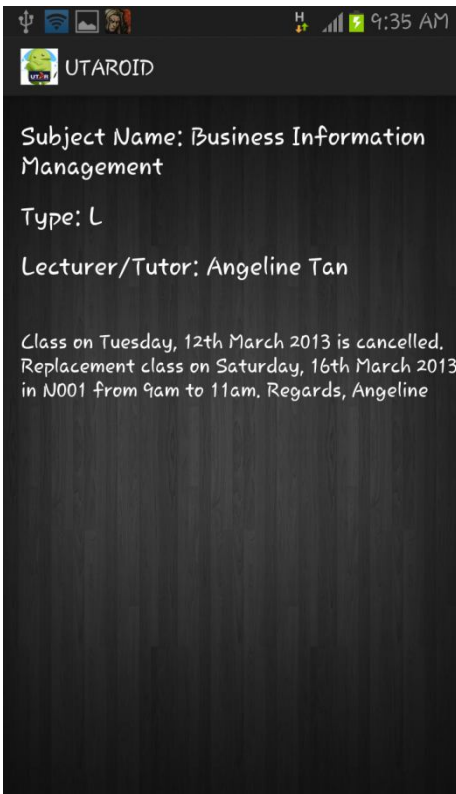
Java File Involved:

- viewTimeTable.java

XML File:

- timetableitem.xml

x. Class Announcement Screen



Functionality:

Display class announcement posted by the respective class's lecturer/tutor. Displayed to user when an item from Class List screen is tapped.

PHP Files Involved:

- read_class_announcement.php

Java File Involved:

- classAnnouncement.java

XML File:

- class_announcement.xml

xi. Event List Screen



Functionality:

Display list of upcoming and currently held events to students. Each item in list comes with the necessary details of the event.

PHP Files Involved:

- read_event.php

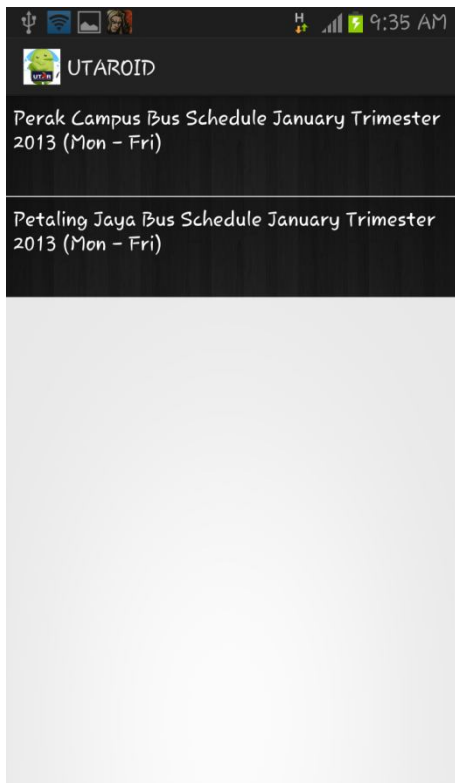
Java File Involved:

- viewEvent.java

XML File:

- eventitem.xml

xii. Bus Schedule List Screen



Functionality:

Display list of available bus schedule documents to be downloaded.

PHP Files Involved:

- read_bus_schedule.php

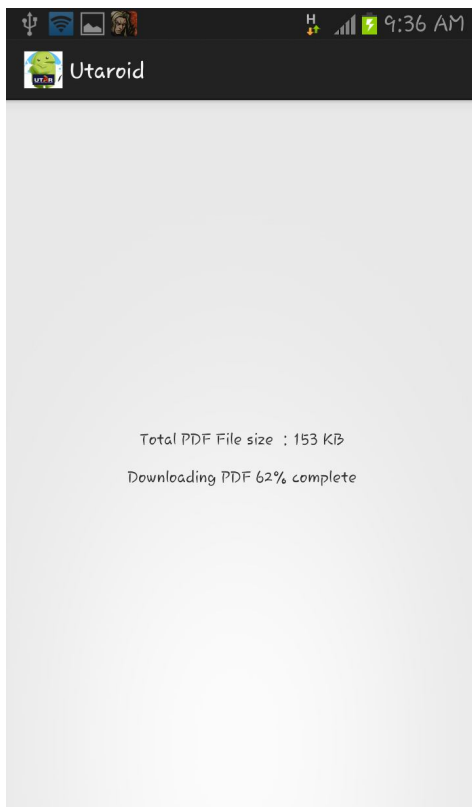
Java File Involved:

- busschedule.java

XML File:

- busitem.xml

xiii. Bus Schedule Downloading Screen



Functionality:

Display the downloading progress of the selected bus schedule item from the Bus Schedule List screen

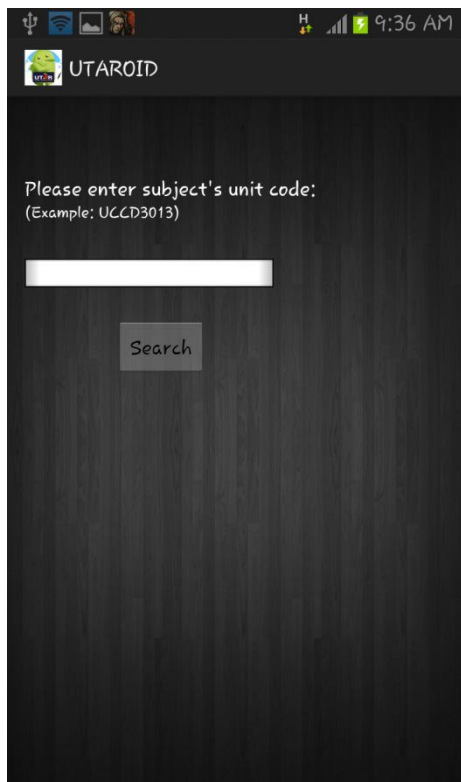
PHP Files Involved: -

Java File Involved:

- busPDF.java

XML File:-

xiv. Search Past Year Paper Screen



Functionality:

Allows user to input subject code before program lists out downloadable past year papers related to the subject code.

PHP Files Involved:

- read_past_year.php

Java File Involved:

- searchPastPaper.java

XML File:

- searchpastpaper.xml

xv. Past Year Paper List Screen



Functionality:

List all downloadable past year papers related to subject code inputted in the Search Past Year Paper screen

PHP Files Involved:

- read_past_year.php

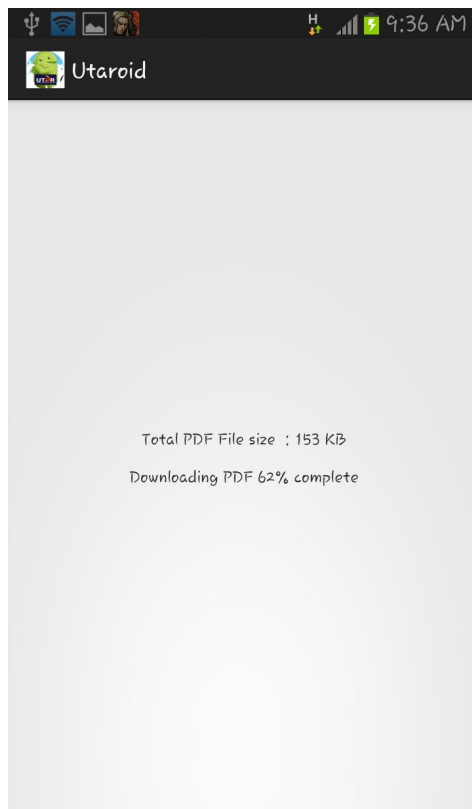
Java File Involved:

- foundPastYear.java

XML File:

- pastyearitem.xml

xvi. Past Year Paper Downloading Screen



Functionality:

Display the downloading progress of the selected past year paper from the Past Year Paper List screen

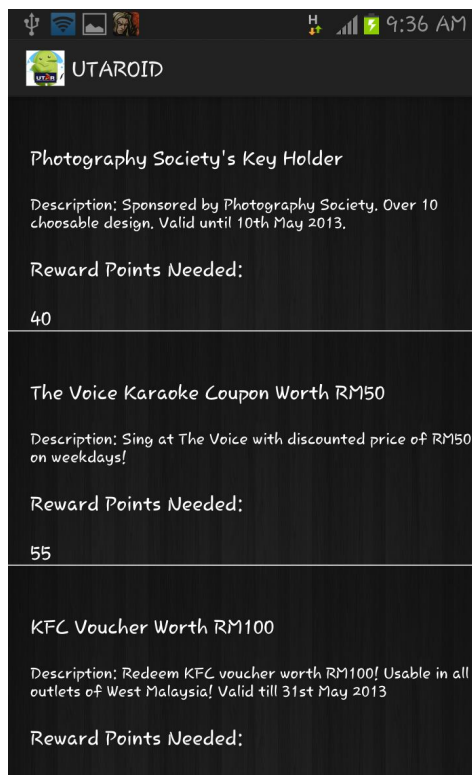
PHP Files Involved: -

Java File Involved:

- pastYearPdf.java

XML File:-

xvii. Reward List Screen



Functionality:

List redeemable rewards for a particular period. Allows immediate redeem upon tapping on an item and confirmation.

PHP Files Involved:

- read_reward.php
- update_points.php
- update_trans.php

Java File Involved:

- rewardList.java

XML File:

- rewarditem.xml

Chapter 9: Testing

In the testing phase, a few evaluators are chosen to test the UTARoid application for errors upon being deployed. The following table shows the Test Plan which includes the expected results and actual results:-

Evaluators Involved: Chin Wei Kit, Chong Jia Zheng, Tong Woei Chyuan

Test No.	Target Screen	Test Name	Purpose of Test	Expected Result	Actual Result	Outcome and Actions Required
1	activity_login.xml	Successful Login	Test that users are allowed to input the correct username and password.	Username and Password textboxes are displayed. Upon entering correct username and password, user will be navigated to the Home Screen.	As expected.	Screen rendered as expected and user navigated to Home Screen when correct username and password entered. No actions required.
2	activity_login.xml	Login Failure	Test what will happen of wrong username and password pair is entered.	Username and Password textboxes are displayed. Upon entering incorrect username and password, an error message box will be displayed at the same screen telling user the cause of login failure.	As expected.	Error message prompted out when incorrect username and password pair is entered. No actions required.

3	homepage.xml	Home Page View	Test what will be displayed on the home page upon navigating to the screen.	<p>On the Upper Frame, picture, student ID, student name, credit hours, faculty and reward points will be displayed based on the verified credentials given in the Login Page.</p> <p>Image buttons leading to other modules/functions of the application will be displayed at the Bottom Frame.</p>	As expected.	<p>The upper frame of the screen displayed the accurate picture and details of the current user. User can navigate across screens as they click on any of the image buttons placed in the lower frame.</p> <p>No actions are required.</p>
4	editprofile.xml	Update Profile	Test if edited data are updated to the database after the Save button is clicked.	<p>Screen displays all information of the student.</p> <p>Editable data includes E-Mail address and mailing address where user is allowed to manipulate/change data.</p> <p>When Save button is tapped, the new data should be updated to the database according to the user's student Id.</p>	As expected.	<p>Student new details successfully updated to database. New details are displayed when screen is refreshed.</p> <p>No actions are required.</p>

5	departmentitem.xml	Categorization & Filtration	Test the auto-categorization and filtration of announcements.	<p>Screen displays list of departments which has posted announcements. Departments with no announcements won't be displayed.</p> <p>Announcements are categorized according to department and for each item displayed in the list, a number in bracket is also displayed indicating the total number of announcements the particular department has.</p>	As expected.	<p>Departments are listed in alphabetical order with number in bracket also viewable in each row of the list indicating total number of announcements the department has. Item when tapped navigates user to list of announcement for selected department.</p> <p>No actions required.</p>
6	view_announcement.xml	View Announcement	Test if the contents of the announcement match with the selected announcement title from the announcement list.	When an item in the announcement list is tapped, user will be navigated to a screen where the full details of the selected announcement could be viewed.	As expected.	Announcement content viewed matches with the selected announcement title from the listing screen. No actions required.
7	commentitem.xml	View Comments	Test if the comments for the selected announcement could be viewed.	<p>When the View Comments button is tapped from the View Announcement screen, user will be navigated to a screen showing a list of comments posted for the selected announcement sorted based on post date.</p> <p>Each row of item should have the comment, author, post date and the announcement title.</p>	As expected.	Comments related to the selected announcement are viewable in list format with all extra details such as author and post date for each comment displayed. No actions required.

8	commentospace.xml	Adding Comment	Test if user can directly post a comment for the selected announcement	<p>In the Post Comment screen (displayed when the Add button from Comment List screen is tapped), user should be able to input comment in the available multiline text box.</p> <p>When the Post button is tapped and confirmation is made, user's comment will be saved in the database and his/her comment shall be viewable in the Comment List screen now.</p>	As expected.	New comment posted for the selected announcement is viewable in the Comment List screen at the bottom of the list. No actions are required.
9	timetableitem.xml	View Time Table based on Selected Day	Test if user could view registered classes listed in ascending order of time for a selected day.	<p>When a user selects a day ranging from Monday to Friday in the Day Selection screen triggered when user wish to view time table, list of classes registered by the student on the particular day shall be listed in ascending order of time. Other details such as venue and lecturer's name shall also be displayed in each item.</p> <p>When no classes were registered for that day, a Toast will be prompted out telling user that there is no class on the particular day.</p>	As expected.	<p>List of classes for the selected day registered by verified user is displayed. Other information for each class such as venue and staff name are also displayed accordingly.</p> <p>When no classes are found, a Toast is prompted out notifying user that he/she has no classes on that day.</p> <p>No actions are required.</p>

10	class_announcement.xml	View Class Announcement	Test if user could view announcement related to the selected class	<p>When a user taps on a class item from the list of classes he/she registered on the selected day, he/she will be navigated to a screen where announcement related to the particular class will be displayed.</p> <p>Extra information such as post date and author will also be displayed.</p>	As expected.	<p>The announcement related to the selected class is displayed with other details such as post date and author.</p> <p>No actions required.</p>
11	eventitem.xml	View Events	Test if user could view list of upcoming and current events happening in UTAR	<p>When user taps the Event button under the Schedule module, he/she will be navigated to a screen listing all the upcoming and current events happening in UTAR.</p> <p>Each item shall have the Event Name, Event Description and Event Date.</p> <p>Nothing happens when any of the item in the list is tapped.</p>	As expected.	<p>The list of upcoming and current events happening in UTAR together with each of the item's details is displayed in ascending order of Event Date.</p> <p>No actions required.</p>
12	busitem.xml	View Bus Schedule List	Test if user could view the list of downloadable bus schedules.	When user tap on the Bus image button at the Home Screen, he/she shall be navigated to a screen listing all updated downloadable bus schedules.	As expected	<p>All downloadable bus schedules are listed with title specifying the Campus and Semester.</p> <p>No actions required.</p>

13	busPDF.java	Download and View Bus Schedule	Test if user could download bus schedule into external storage and directly view the document	<p>When an item is selected form the list of bus schedule, application will directly download the selected file via a specified URL.</p> <p>A progress bar will be shown notifying the downloading progress to the user.</p> <p>When download is complete, the file will be saved in the device's external storage.</p> <p>If the device is installed with a suitable document reader, the downloaded file will automatically be opened and displayed to the user.</p>	As expected	<p>Selected bus schedule file is successfully downloaded, saved into external storage for offline reading purpose and automatically opened and displayed when suitable document reader is installed in the device.</p> <p>No actions required.</p>
----	-------------	--------------------------------	---	--	-------------	--

14	busPDF.java	Download Bus Schedule but Failure to View	Test if user could download bus schedule into external storage and what will happen if no suitable reader is installed	<p>When an item is selected form the list of bus schedule, application will directly download the selected file via a specified URL.</p> <p>A progress bar will be shown notifying the downloading progress to the user.</p> <p>When download is complete, the file will be saved in the device's external storage.</p> <p>When a suitable file reader is not installed within the device, an error message will be prompted out notifying the user that he/she needs to install a suitable reader to view the document.</p>	As expected	<p>Error message prompted out. File is saved in the external memory but could not be viewed as no suitable reader is installed within the device.</p> <p>No actions required.</p>
15	searchpastpaper.xml	Search Past Year Exam Successful	Test if user could view list of past year papers when entered subject code is valid	<p>Users are allowed to enter subject code in the only text field displayed either in upper-case or lower-case characters.</p> <p>When the search button is tapped, application begins the search process of downloadable past year exam papers based on the entered subject code.</p> <p>Documents are found and listed.</p>	As expected.	<p>All related downloadable past year examination are found and listed in the next screen.</p> <p>No actions required.</p>

16	searchpastpaper.xml	Search Year Failure	Past Exam	<p>Test if error message is prompted out if invalid subject code is entered/no document found.</p>	<p>Users are allowed to enter subject code in the only text field displayed either in upper-case or lower-case characters.</p> <p>When the search button is tapped, application begins the search process of downloadable past year exam papers based on the entered subject code.</p> <p>Documents are not found and error message is prompted.</p>	As expected	<p>No downloadable documents related to the entered subject code are found. Error message prompted out and user stays on the same screen.</p> <p>No actions required.</p>
17	pastYearPdf.java	Download and View Past Year Examination		<p>Test if user could download selected past year examination document into external storage and directly view the document</p>	<p>When an item is selected from the list of past year exam, application will directly download the selected file via a specified URL.</p> <p>A progress bar will be shown notifying the downloading progress to the user.</p> <p>When download is complete, the file will be saved in the device's external storage.</p> <p>If the device is installed with a suitable document reader, the downloaded file will automatically be opened and displayed to the user.</p>	As expected	<p>Selected past year exam file is successfully downloaded, saved into external storage for offline reading purpose and automatically opened and displayed when suitable document reader is installed in the device.</p> <p>No actions required.</p>

18	pastYearPdf.java	Download Past Year Examination but Failure to View	Test if user could download selected past year examination file into external storage and what will happen if no suitable reader is installed	<p>When an item is selected form the list of past year exam, application will directly download the selected file via a specified URL.</p> <p>A progress bar will be shown notifying the downloading progress to the user.</p> <p>When download is complete, the file will be saved in the device's external storage.</p> <p>When a suitable file reader is not installed within the device, an error message will be prompted out notifying the user that he/she needs to install a suitable reader to view the document.</p>	As expected	<p>Error message prompted out. File is saved in the external memory but could not be viewed as no suitable reader is installed within the device.</p> <p>No actions required.</p>
19	rewarditem.xml	View Rewards	Test if user could view list of redeemable rewards.	<p>When user taps the Reward button in the Home Screen, he/she will be navigated to a screen listing all redeemable rewards.</p> <p>Each item shall have the Reward Name, Description, Points Needed and Total Stock Available.</p> <p>Redeem process engaged when any of the item is tapped.</p>	As expected.	<p>The list of redeemable rewards with each of the item's details is displayed in ascending order of Points Needed.</p> <p>No actions required.</p>

20	rewarditem.xml	Successful Reward Redemption	Test if user with sufficient reward points could redeem selected reward.	<p>When the user taps on an item in the Reward List, application will compare user's total reward points and the reward points needed for the selected reward redemption.</p> <p>When user's points are sufficient, application then checks total available stock of the selected item.</p> <p>When stock is sufficient, a confirmation message will be prompted out.</p> <p>When confirmation is made, information message will be prompted out stating selected reward is successfully redeemed and user's points will be deducted accordingly.</p>	As expected.	<p>User successfully redeemed the selected reward and points are deducted accurately. The stock of the selected reward is also deducted by 1.</p> <p>No actions required.</p>
21	rewarditem.xml	Failure in Reward Redemption due to Insufficient Points	Test if error message will be prompted out when user has insufficient points when redeeming reward.	<p>When the user taps on an item in the Reward List, application will compare user's total reward points and the reward points needed for the selected reward redemption.</p> <p>When user's points are insufficient, error message prompted out.</p>	As expected.	<p>Error message prompted out and no changes made to user's reward points. User stays on the same screen.</p> <p>No actions required.</p>

22	rewarditem.xml	Failure in Reward Redemption due to Insufficient Stock	Test if error message will be prompted out when selected reward does not have any stock left.	<p>When the user taps on an item in the Reward List, application will compare user's total reward points and the reward points needed for the selected reward redemption.</p> <p>When user's points are sufficient, application then checks total available stock of the selected item.</p> <p>Stock is checked to be insufficient and error message is prompted out.</p> <p>Redemption failed.</p>	As expected.	<p>Error message prompted out stating that selected reward is in insufficient state.</p> <p>User's points are not deducted and user stays on the same screen.</p> <p>No actions required.</p>
23	announcement.php (Available in htdocs as this test is related to PHP dummy website.)	Post Announcement using Dummy Website	Test if staff could successfully post announcement using the PHP dummy website.	<p>After successful login, user is allowed to click on the Post New Announcement button found on the top of the screen.</p> <p>User then can fill in the required fields such as contents and select the department he/she belongs to.</p> <p>As the Submit button is clicked, a validation will be performed to check if all required fields are filled in the correct format.</p> <p>New announcement posted and saved in database.</p>	As expected.	<p>New announcement posted successfully using the verified staff ID.</p> <p>In the UTARoid application, newly posted announcement is viewable by student.</p> <p>No actions required.</p>

24	announcement.php (Available in htdocs as this test is related to PHP dummy website.)	View and Post Comment using Dummy Website	Test if staff could view comments for selected announcement and successfully post a new comment using the PHP dummy website.	<p>After successful login, user is shall be able to view all announcements in the dummy website.</p> <p>As user select on an announcement, the contents of the announcement and all its comments shall be displayed.</p> <p>User is allowed to post new comment by filling the Add Comment text field.</p> <p>As the Submit button is clicked, a validation will be performed to check if the Add Comment field is filled in the correct format.</p> <p>New comment posted and saved in database. New comment is also now viewable both in the website and the UTARoid application.</p>	As expected.	<p>New comment posted in the selected announcement.</p> <p>In the UTARoid application, student is able to view the newly posted comment.</p> <p>No actions required.</p>
----	---	---	--	---	--------------	--

Table 9.1.1: Test Plan

Chapter 10: Future Enhancements

This chapter suggests future enhancements which could be implemented into UTARoid to maintain the loyalty of its users besides making it more sophisticated, usable and contributive. The following are the enhancement where developers shall consider:

10.1 Trigger of Notifications

Whenever there is a new related announcement being posted or an update of any existing announcements, notifications should be triggered to alert the students. The trigger of notifications should be activated in a timely and efficient manner especially when the student just login into the application. When a new announcement is posted or an existing one is updated, signed-in student must be notified automatically even if he/she is currently not accessing the application (performing other activities).

Notifications are compulsory to be triggered whenever there are new or updated announcements detected in the six favourite announcement categories chosen by the student. Notifications for other categories which are not selected as favourite are optional. Again, this functionality should be easy to be used via implementation of checkboxes.

The notification should clearly state which category the new or updated announcement belongs to and who posted it. The following diagram shows the basic flow on how a notification for UTARoid in Android platform can be triggered:-

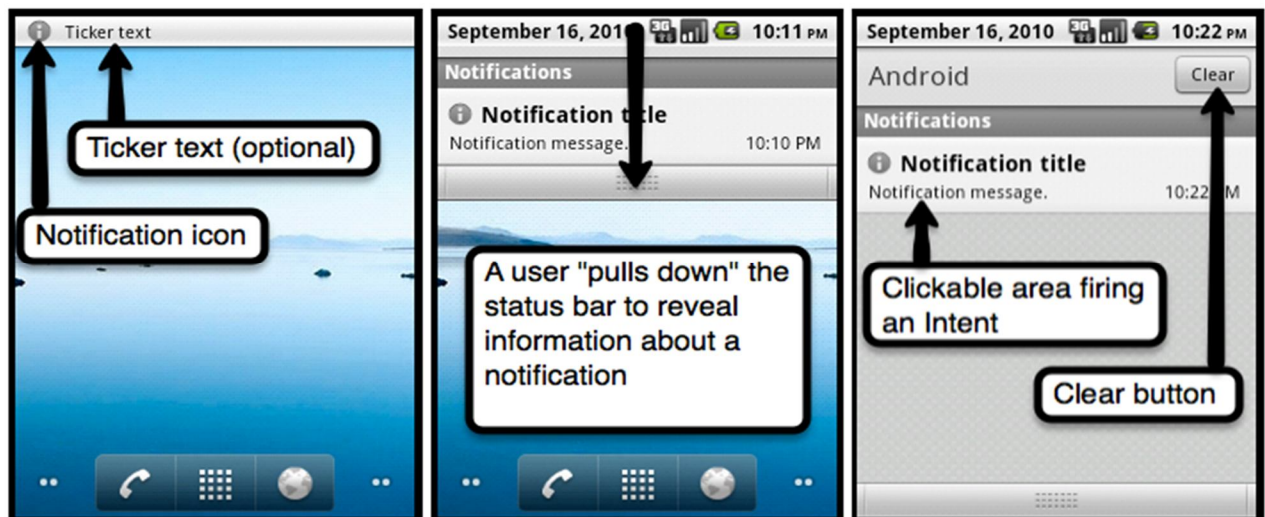


Figure 10.1.1 – Notification in Android shown to user via Status Bar (Mobilepearls, n.d.)

10.2 Automatic Sign-In and Update

Whenever the Android phone is connected to any WiFi services especially the one which is provided in the university compound or Internet, the application should be able to automatically sign-in the student's account using the verified username and password of the latest login. This functionality should be configurable in a way that the student can choose to either allow automatic sign-in or disallow it. For any successful login, either manually or automatically, students should be able to see all the updated announcements or unread bulletins for every category including their favourite ones besides retrieving notifications if there are any important messages that are related to them.

In other words, if the function is allowed and Internet is connected via any means, the application will automatically sign-in a student's account using the verified username and password combination of the previous successful login.

There should be a design in such a way that the signed-in student will be able to acknowledge which category contains new or updated announcements which are unread. In the sample design below, the icon will be able to tell its user that there is 1 new Twitter post yet to be read.

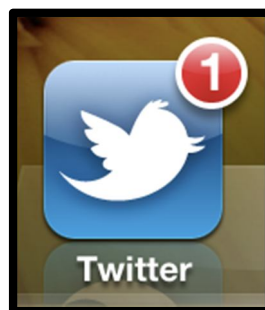


Figure 10.2.1 – A sample diagram on how to acknowledge user new, updated or unread posts by using Twitter application example (Tech Tips & Tricks, 2012)

10.3 Add-On of Other New Modules

As can be seen in the initial version of UTARoid, only five main modules are included i.e Announcement, Time Table/Event, Bus Schedule, Past Year Examination and Rewards (New module that has never existed in other announcement channels).

Thus, when the application becomes stable and an upgrade of version is needed, developers should consider in adding on new modules into the application based on the priorities, popularity and requirements of the UTAR students (as can be seen in the Survey Results in Appendix C).

Example of modules which could be added include Billing, Soft Skills, Academic Advisor, Course Structure and mini-WBLE where notes and exercises could be downloaded within few single clicks anytime anywhere. The addition of these modules from time to time will eventually add in value and usefulness to the application itself; gaining more new users and maintaining existing ones.

10.4 Sharing Announcement to Social Networking Websites

Verified users of the application are allowed to connect to social networking websites such as Facebook and Twitter using their own account. Then, upon reading any announcement, they have the option of sharing the announcement via their connected social network accounts by just a simple click on the “Share” button.

With this functionality also known as viral marketing in business point of view, announcements posted by staff will be more easily to be spread to the public and thus, serving the goal of this project which is to effectively and efficiently communicate announcement to UTAR students.

10.5 Filter of Harsh Words in Posted Comments

To avoid conflicts besides maintaining the image of the university, comments which is about to be posted by students shall firstly be checked by the system if they contain any vulgarity or inappropriate words. If such words are found, the comment should automatically be filtered

and not be allowed to be posted/saved to the database; preventing any readers from being able to read the particular comment.

Chapter 11: Conclusion

As Android-smart phone users among UTAR students will never stop to increase, an announcement portal which is to be implemented in Android platform is relevant enough to be developed. With the availability of this application named UTARoid, problem faced by university staffs in conveying messages to students could be solved in a more effective and efficient way when compared to other existing platforms. Plus, students will not have difficulties to retrieve new and updated announcements anytime anywhere as long as their Android phone is capable of retrieving Internet connection, thus, ruling out chances for them to miss out bulletins or alerts related to them or which serves as their golden opportunities for future career or achievements. The sophisticated features that Android platform have could also result in a UTAR announcement portal which is more attractive, user-friendly, interactive and notification-effective.

As the developed application is targeted to students and their collaboration in the testing phases is highly essential, the agile method is selected. Plus, agile methodology supports iterative process where developers could return to previous stages of the life cycles and make improvements or enhancements based on the requirements of the students.

Meanwhile, two techniques are used to gather requirements for the project, namely Survey method and Interview Sessions. Survey method is highly effective to gather a huge amount of students' information regarding popularity of Android, opinions of existing announcement platforms and expectations of UTARoid in a short time. Interview session with one of the staff of the top management in another case, could aid in getting a clearer view on what problem which is faced by the organization in conveying messages to students.

In the implementation process, Java programming language will be used since Android development supports the mentioned language. As every Android application recommends never-ending upgrade of versions to remain competitive in the market or to maintain loyalty of current users, UTARoid should also consider further enhancements in terms of notifications, automatic sign-in and update and add-on of other new modules as requested by students themselves.

References

Association of Modern Technologies Professionals (2012) *Software Development Methodologies*. [online] Available at: <http://www.itinfo.am/eng/software-development-methodologies/> [Accessed: 20 July 2012].

Brady, A. (2012) *3 Ways Mobile Apps Can Reward Customer Loyalty*. [online] Available at: <http://therealtime.com/2012/09/03/3-ways-mobile-apps-can-reward-customer-loyalty/> [Accessed: 2 Mar 2013].

Clay, K. (2012) *Will 2013 Be The Year of Loyalty Programs?*. [online] Available at: <http://www.forbes.com/sites/kellyclay/2012/12/15/will-2013-be-the-year-of-loyalty-programs/> [Accessed: 2 Mar 2013].

Dempsey, J. (2012) *Android loses to Apple when it comes to security*. [online] Available at: <http://www.cvbizjournal.com/columns/james-dempsey/android-loses-to-apple-when-it-comes-to-security.html> [Accessed: 7 July 2012].

Developer.android.com (2012) *Android 4.1 for Developers | Android Developers*. [online] Available at: <http://developer.android.com/about/versions/jelly-bean.html> [Accessed: 7 July 2012].

Egham. (2012) *Gartner Says Worldwide Smartphone Sales Soared in Fourth Quarter of 2011 With 47 Percent Growth*. [online] Available at: <http://www.gartner.com/it/page.jsp?id=1924314> [Accessed: 10 July 2012].

Escallier, P. (2010) *Maximum PC | 10 Things Android Does Better Than iPhone OS - Page 1*. [online] Available at: http://www.maximumpc.com/article/features/10_things_android_does_better_iphone?page=0_0 [Accessed: 7 July 2012].

Gatherspace.com (n.d.) *Agile Software Development*. [online] Available at: http://www.gatherspace.com/static/agile_software_development.html [Accessed: 20 July 2012].

Hassell, J. (2010) *Developing for the iPhone and Android: The pros and cons - Computerworld*. [online] Available at:

http://www.computerworld.com/s/article/9178684/Developing_for_the_iPhone_and_Android_The_pros_and_cons?taxonomyId=18&pageNumber=1 [Accessed: 7 July 2012].

Haughey, D. (2009) *Waterfall v Agile: How Should I Approach My Software Development Project?* . [online] Available at: <http://www.projectsart.com/articles/waterfall-v-agile-how-should-i-approach-my-software-development-project.php> [Accessed: 20 July 2012].

Jones, M. (2011) *Agile Project Requirements Gathering: Techniques to Identify User Stories*. [online] Available at: <http://www.brighthubpm.com/agile/123103-assembling-a-toolbox-for-agile-requirements/> [Accessed: 17 Aug 2012].

Lengstorf, J. (2009) *JSON: What It Is, How It Works, & How to Use It*. [online] Available at: <http://www.copterlabs.com/blog/json-what-it-is-how-it-works-how-to-use-it/> [Accessed: 2 Mar 2013].

Limkokwing University of Creative Technology (n.d.) *Android | Limkokwing University of Creative Technology*. [online] Available at: <http://www.limkokwing.net/android/> [Accessed: 12 July 2012].

Maji, A. K., Kangli, H., Sultana, S. & Bagchi, S. (2010) Characterizing Failures in Mobile OSes: A Case Study with Android and Symbian. [Online]. Available at: https://engineering.purdue.edu/dcs/publications/papers/2010/android_issre10_submit.pdf [Accessed: 7 July 2012]

Mobilepearls (n.d.) *Android Notifications*. [online] Available at: <http://mobilepearls.com/labs/android-notifications/> [Accessed: 17 Aug 2012].

Mochal, T. (2001) *Gathering business requirement*. [online] Available at: <http://www.techrepublic.com/article/gathering-business-requirements/1045549> [Accessed: 17 Aug 2012].

Mochal, T. (2008) *10 techniques for gathering requirements*. [online] Available at: <http://www.techrepublic.com/blog/10things/10-techniques-for-gathering-requirements/287> [Accessed: 17 Aug 2012].

Nitschinger, M. (2010) *Handling JSON like a boss in PHP*. [online] Available at: <http://nitschinger.at/Handling-JSON-like-a-boss-in-PHP> [Accessed: 3 Mar 2013].

Priya Viswanathan (n.d.) *Android OS Vs. Apple iOS – Which is Better for Developers?*. [online] Available at: <http://mobiledevices.about.com/od/kindattentiondevelopers/tp/Android-Of-Vs-Apple-Ios-Which-Is-Better-For-Developers.htm> [Accessed: 7 July 2012].

Rodriguez, B. (2012) *Android vs. iOS: the Pros and Cons of Two App Platforms*. [online] Available at: <http://www.lunawebs.com/blog/2012/03/18/android-vs-ios-the-pros-and-cons-of-two-app-platforms-825/> [Accessed: 7 July 2012].

Samuel90. (2010) *Requirements Gathering Techniques*. [online] Available at: <http://www.slideshare.net/Samuel90/requirements-gathering-techniques> [Accessed: 17 Aug 2012].

Spataru, A., C. (2010) *Agile Development Methods for Mobile Applications*. [Online]. Available at: <http://www.inf.ed.ac.uk/publications/thesis/online/IM100767.pdf> [Accessed: 20 July 2012]

Tamada, R. (2012) *How to connect Android with PHP, MySQL*. [online] Available at: <http://www.androidhive.info/2012/05/how-to-connect-android-with-php-mysql/> [Accessed: 3 Mar 2013].

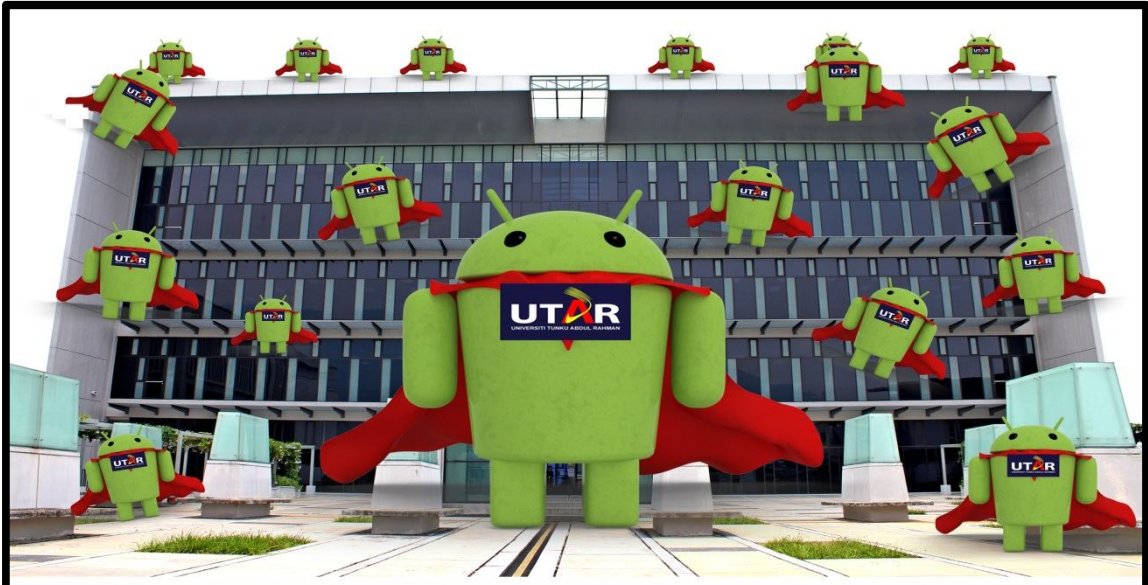
Tech Tips & Tricks (2012) *How To Add A Notification Alert Badge In An Android ImageView*. [online] Available at: <http://tech.chitgoks.com/2012/07/06/how-to-add-a-notification-alert-badge-in-an-android-imageview/> [Accessed: 17 Aug 2012].

WillowTree Apps, Inc (n.d.) *University of Virginia iPhone and Android Apps*. [online] Available at: <http://www.willowtreeapps.com/apps/education/university-of-virginia/> [Accessed: 8 July 2012].

Wiras Adi (2008) *System Development Life Cycle (SDLC) Methodologies*. [online] Available at: <http://www.slepi.net/blog/system-development/system-development-life-cycle-sdlc-methodologies.html> [Accessed: 20 July 2012].

Yap, W. J. (2012) *Auto WiFi MMU - Android Apps on Google Play*. [online] Available at: <https://play.google.com/store/apps/details?id=my.edu.mmu.wifi&hl=en> [Accessed: 20 July 2012].

APPENDIX A: POSTER



UTAROID

ANDROID-BASED ANNOUNCEMENT PORTAL FOR STUDENTS

UTAR Latest Announcements and Alerts in Your Hands Anytime Anywhere!



Events, Competitions, Time Table, Bus Schedule, Examination, Cancellation of Classes and many others in one single application!



A FYR PROJECT BY CHIN WEI KIT

APPENDIX B: SURVEY SHEET SAMPLE

Student Survey for UTARoid : Android-based Announcement Portal

Faculty: _____ Course: _____ Trimester: _____

Gender: M/F _____ Date: _____

My name is Chin Wei Kit and I am currently pursuing Bachelor of Information Systems (Hons) Business Information Systems in Universiti Tunku Abdul Rahman (UTAR). As a Year Three student, I am required to complete my Final Year Project (FYP) based on the approved project title namely UTARoid.

UTARoid is generally an application specially designed for Android smart phone users which is aimed to facilitate communications of announcements between UTAR faculty staffs and the students. Besides, UTARoid will also be designed in a way that will encourage students to frequently access the announcement portal anytime anywhere, thus ruling out chances for them to miss any bulletins or alerts especially last minute ones.

Among the objectives which are expected to be accomplished with this survey are

- g) To calculate the percentage of students who possess at least one Android smart phone
- h) To determine how frequent the students access to the existing announcement platforms
- i) To investigate which categories of announcements students view the most
- j) To determine the problems of the existing announcement platforms that could not communicate announcements to students effectively and efficiently
- k) To determine the expectations or requirements of the newly developed UTARoid application
- l) To retrieve ideas and suggestions from students on how they would increase their loyalty towards the newly developed announcement portal

Hence, your participation is truly appreciated.

Section A – Android Smart Phone Users Among UTAR Students

1. Do you at least possess an Android smart phone?
 Yes No

2. If Yes, what is the brand of your Android smart phone you are currently using?
 Sony Ericsson Samsung HTC
 Huawei Others (please specify) _____

3. If No, assuming that you have adequate amount of money, will you purchase one?
 Yes No

4. If No, why?

Section B – Study on Existing Platforms

1. Which of these existing announcement platforms that you usually read updates, bulletins or alerts from?
 UTAR Main Website WBLE UTAR Portal
 UTAR Student E-Mail Facebook Twitter

2. What other sources do you normally get the announcements from? (*Optional*)
 Verbally from Friends Verbally from Lecturers/Tutors
 Others (Please Specify) _____

3. Generally, how frequent do you access the existing announcement platforms? (*Choose one only*)
 Daily Once a week Once in a fortnight
 Once a month Only when asked/forced to do so
 Others (Please Specify) _____

4. Are the existing platforms well-organized in terms of arrangement of information?
 Yes No

5. Are the existing platforms well-categorized (easily find preferred bulletins, alerts etc.)?
 Yes No

6. Are the existing platforms attractive enough to encourage you to access them frequently?
 Yes No

7. Are the existing platforms interactive enough allowing students to ask questions/comment/suggest and get immediate replies for a particular announcement?
 Yes No

8. Are you notified effectively and efficiently for updates made especially last minute ones in the existing platforms?
 Yes No

9. Which of the existing platforms do you favour the most in terms of organization, categorization, interactivity and attractiveness?
 UTAR Main Website WBLE UTAR Portal
 UTAR Student E-Mail Facebook Twitter
 None of the above

Section C – UTARoid Expectations & Requirements

1. Please rank the following features of the application; 1 being the most important element in effectively and efficiently communicating announcements to students and 8 being the least important.

- Attractiveness (Usage of more attractive and simple texts, audios, animations, graphics or other video elements)
- Categorization (Announcements are categorized accordingly based on types or departments to allow easy search of preferred group of bulletins, alerts etc.)
- Personalization (Students are allowed to filter favourite/frequently access announcement categories on their main menu)
- Notifications (Notifications being triggered in a timely manner whenever there is a new announcement being posted or updates especially last minute ones are performed)
- Downloadable Contents (Downloadable announcements or information to allow offline readable contents especially for time table, bus schedule etc.)
- Interactivity (Students to be able to ask questions, comment or suggest for a particular announcement and get immediate replies either from mates or the person who posted the announcement)
- Reward Points System (Students to gain reward points for each announcement read and with accumulated points could be exchanged with a reward such as soft skill incentives, cash etc.)
- User-Friendliness (Students to be able to access a particular section with just few clicks and no complications were experienced while exploring the application)

APPENDIX C: SURVEY RESULTS

As one of the techniques to have a better understanding on the requirements needed for the project, to study student's opinions towards existing announcement platforms and to determine the popularity of Android smart phones, a survey has been conducted. There were up to 30 respondents for this survey which comprises of student from Faculty of Information Communication and Technology (FICT) and FS (Faculty of Science).

Under this appendix, a tabulation of the survey results is made and inferences are given based on the tabulated data.

Survey Results of Section A – Android Smart Phone Users Among UTAR Students

In the first section of the survey, questions which determine the popularity of Android smart phone users among UTAR students were asked. The following pie chart shows the amount of students who own at least an Android smart phone and those who do not:-

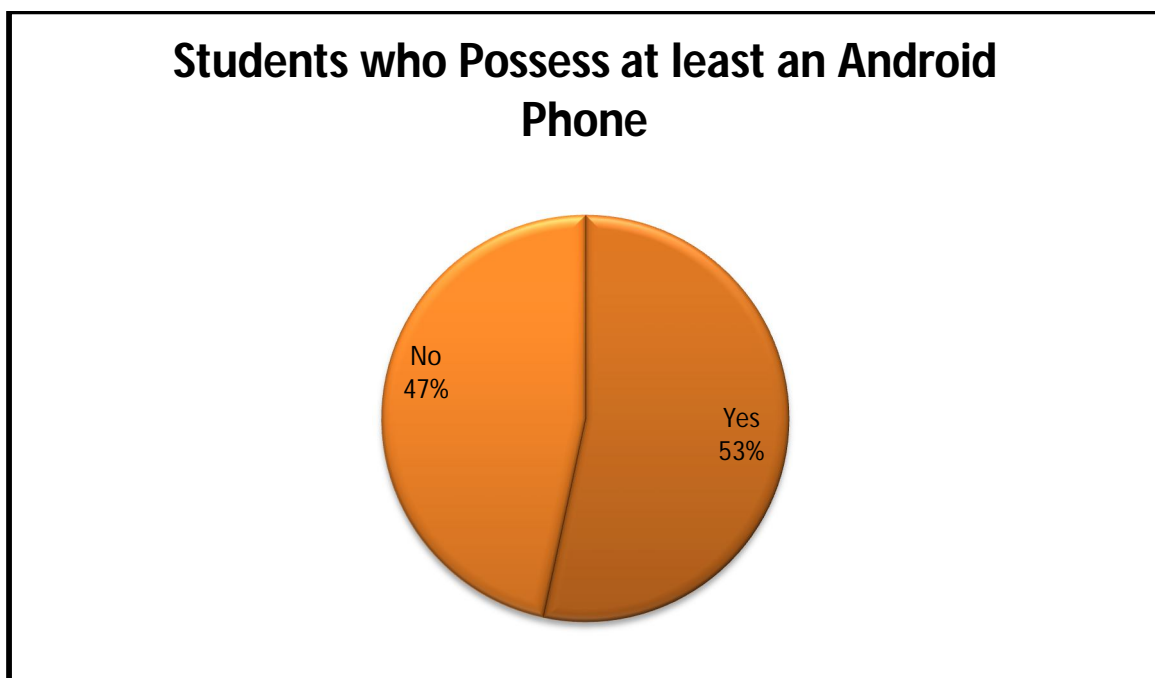


Figure A1 – Pie Chart on Amount of Students who Possess at least an Android Smart Phone

From the pie chart above, we can conclude that 53% of the 30 respondents (up to 16 students) at least own an Android smart phone. Extra information and inferences which are obtained from the first section of the survey include:-

- 8 out of 14 students who do not possess an Android smart phone will buy one assuming that they have enough amount of money.

- The remaining 6 will not buy an Android smart phone because most of them are fans of Apple's iOS, another smart phone operating system.
- From the results tabulated, it can be foreseen that the use of smart phones either of Androids or iOS is popular among students in UTAR and will further grow as time passes. Thus, developing UTARoid as an approach for Android-based announcement portal is a rational move.

Survey Results of Section B – Study on Existing Platforms

In the second section of the survey, questions were asked to gain information on respondents' frequency of visit to existing announcement platforms, platforms which they visit and favour most and their opinions on these platforms.

The following table shows the tabulated results on which existing platform that the respondents usually visit to read updates, bulletins and alerts. (For this question, respondents are allowed to choose more than one platform)

Existing Platform	Number of Respondents
UTAR Main Website	3
WBLE	25
UTAR Portal	19
UTAR Student E-Mail	0
UTAR Facebook Fan Page	17
UTAR Twitter Account	1

Table A1 – Results on which Existing Platform that Respondents Usually Visit for Announcements

From the above table, it can be concluded that WBLE is the most visited platform by the respondents whereas not a single person check their own UTAR e-mail for announcements.

Besides the above existing platforms, it was also recorded that

- 100% of the participants also get notified on announcement verbally from friends.
- 73.3% of the participants usually get notified on announcement also verbally from their lecturers or tutors.
- Other sources of announcements include banners, posters and notice boards all around the campus. However, only two respondents were usually notified via these resources.

The following bar chart is tabulated to study on the respondents' frequency of visit to these existing announcement platforms.

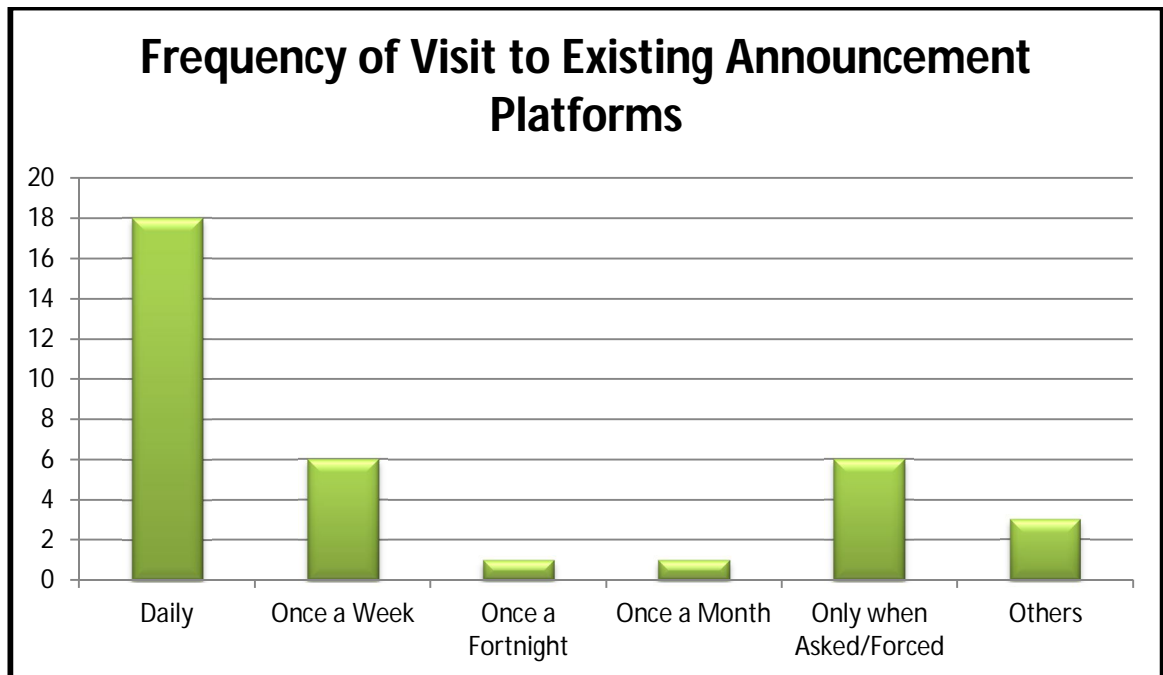


Figure A2 – Bar Graph showing Frequency of Visit of Respondents to Existing Announcement Platforms

From the bar graph above, a few conclusions and inferences can be made:-

- Majority of the students (18 of them) visit the existing platforms daily in general. However, problem of conveying announcements to students still exist and this matter need to be investigated.
- A worrying factor is that there are still some students (almost 12 of them when summed up) who only visit these platforms less frequent or only when asked or forced to do so. Thus, chances of them missing any important alerts are high.
- Other answers include 'Once in a few days', '2-3 times per week' and 'Very Rare'.

The following bar graph shows the opinions of the respondents towards the existing platforms based on characteristics of organization, categorization, attractive, interactive and effectiveness and efficiency of notification.

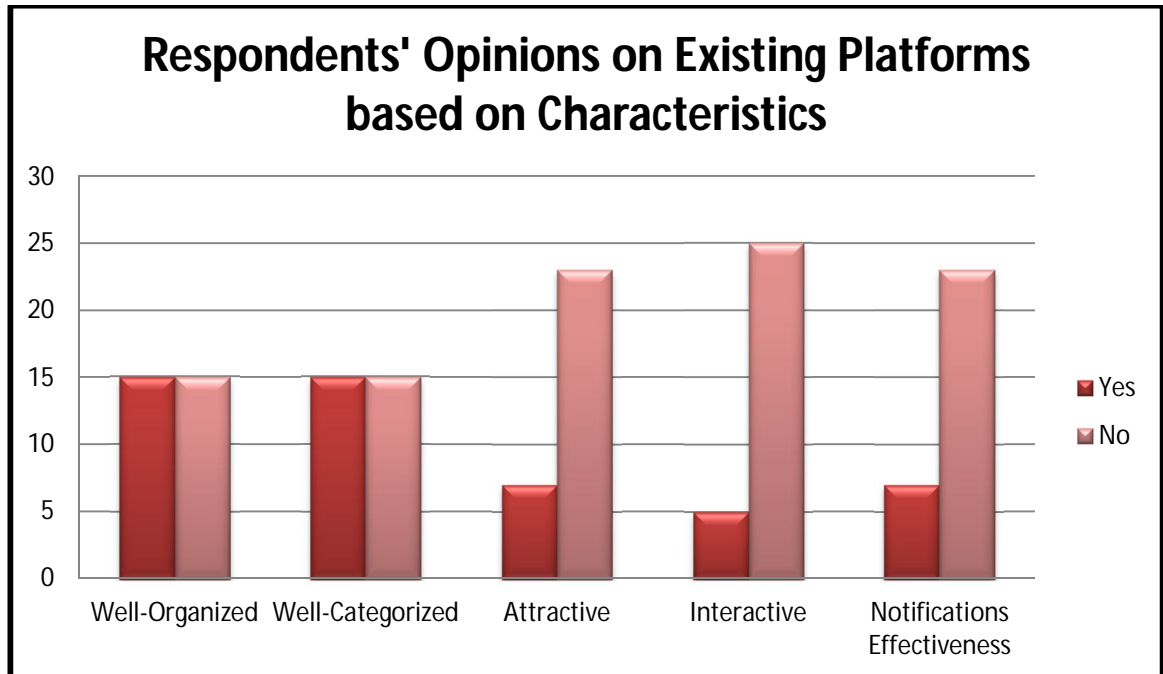


Figure A3 – Bar Chart of Respondents’ Opinions on Existing Platforms based on Characteristics

From the above bar chart, conclusion and inferences which can be made are:-

- Half of the students think that the existing platforms are well-organized and well-categorized whereas the other half do not.
- 23 respondents think that the existing platforms are not attractive enough. This may be the major cause of decreasing of student’s loyalty to visit the existing platforms frequently.
- 27 respondents think that the existing platforms are non-interactive, thus, having difficulties in asking questions, commenting, suggesting or getting immediate replies for a particular announcement.
- 23 respondents chose the answer “No” when they are asked if they are notified effectively and efficiently for updates made especially last minute ones.
- Hence, from this section, it can be seen that existing platforms majorly lack of attractiveness, interactivity and effectiveness and efficiency in sending notifications of announcement updates.

Respondents were also asked for their favourite existing platform in terms of organization, categorization, interactivity and attractiveness in this section. The following pie chart signifies their response:-

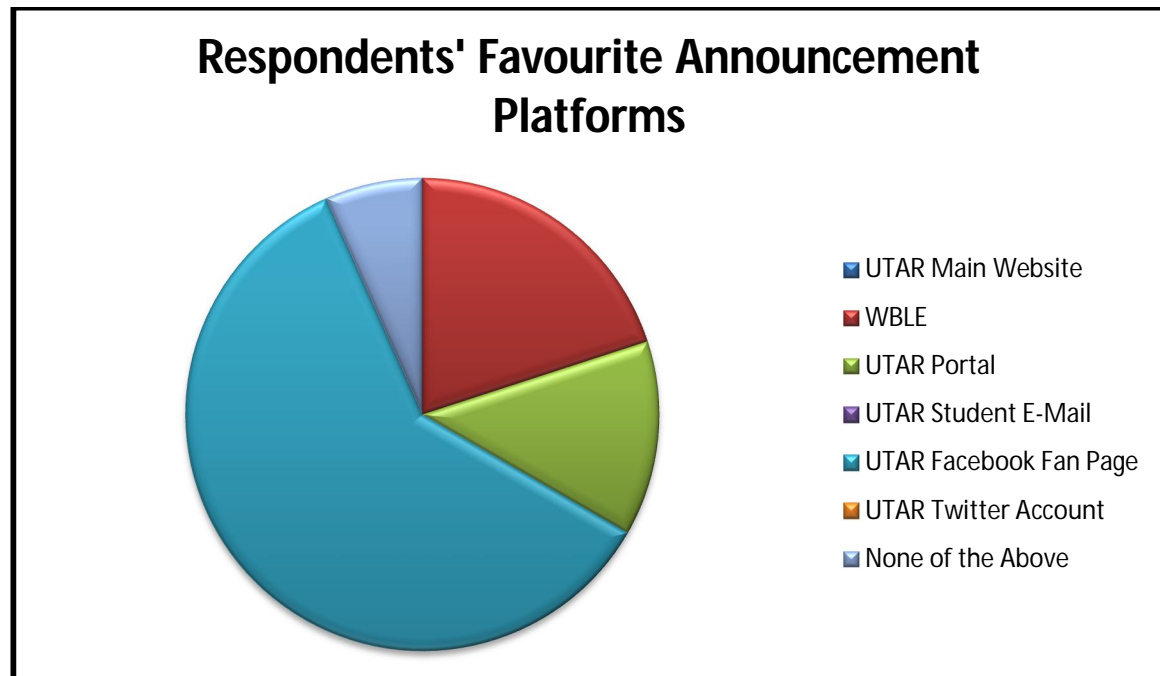


Figure A4 – Pie Chart of Respondents' Favourite Announcement Platforms

From the pie chart above, conclusions and inferences can be made as follows:-

- The most favourite announcement platform which the respondents voted for is UTAR Facebook Fan Page (18 of them).
- The announcement platform which is self-developed by UTAR officials and has garnered the most votes as a favourite is UTAR WBLE (6 of them).
- UTAR Main Website, UTAR Student E-Mail and UTAR Twitter Account are the least favourite as each of them did not have any votes.
- Three respondents felt that none of the platforms are their favourites.
- Since Facebook is the most accepted platform by the respondents, its design and architecture can be used as a reference for the development of UTARoid. Designs ideas of UTAR WBLE and UTAR Portal are not to be ignored too.

Survey Results of Section C – UTARoid Expectations and Requirements

In the final section of the survey, respondents were queried on what are their expectations and requirements from the development of UTARoid application. The results from this section are essential for the understanding of what student really need from new announcement platform which uses Android operating system.

In the first part of Section C, participants were asked to rank the features which they think is important for UTARoid (1 being the most important and 8 being the least important). From the ranking made by these students, the following table of results is formulated:-

Features	Times Ranked as Most Important	Times Ranked as Least Important
Attractiveness	4	3
Categorization	2	1
Personalization	0	5
Notifications	11	0
Downloadable Contents	3	1
Interactivity	3	2
Reward Points System	1	16
User Friendliness	6	1

Table A2 – Ranking of Most and Least Important UTARoid Features Expected by Respondents

From the results tabulated above, the following conclusions and inferences can be made:-

- The most important element in which students really need is Notifications where they are automatically notified in a timely manner whenever there is a new announcement posted or updates are made especially last minute ones.
- The least important element voted by most of the students is the Reward Points System. It can be assumed that what students really need is to just acknowledge announcements in a timely manner and obtaining rewards by reading announcements is totally unnecessary. However, this system should not be totally ignored as from the results obtained from one of the questions where students are asked if this system could enhance their loyalty, 13 students answered ‘Yes’ whereas 12 answered ‘Maybe’.

- As there are mixed rankings from the respondents, all other elements will also be put into consideration while developing the application. However, only the features which generally receive top rankings will be put more focuses on while features that are not preferred by most respondents will be less prioritized.

The next question asked in this section is what announcement categories which respondents wish to be included in the UTARoid application. From the results gathered, it can be seen that most students wish to have the following categories to be accessed via their smart phones:-



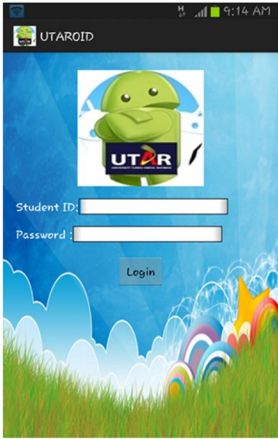
- General Announcements
- Academic Advisor
- Bus Schedule
- Cancellation of Class
- Course Structures
- Events
- Examinations
- Fees
- Final Year Project (FYP)
- Time Table

Finally, here is the list of selected recommendations, suggestions or comments which are received when the survey is conducted:-

- Application not limited to Android users only. Developers should develop one for Apple iOS's users. (Top suggestion)
- Ensure staffs especially lecturers have the knowledge to update information to UTARoid and fully utilize this platform. A worrying factor is that some staffs do not even possess an Android phone, thus, having no knowledge on how students will retrieve the announcement.
- All information from Portal and WBLE could be synchronized into one application, thus avoiding the need to go from one platform to another.
- Notification of coursework marks uploaded by lecturers should be included.
- Integration with UTAR OPAC System to search for reference books and journals.
- Avoid frequent system crash scenarios and make sure application is usable in all Android phones.

APPENDIX D: USER MANUAL

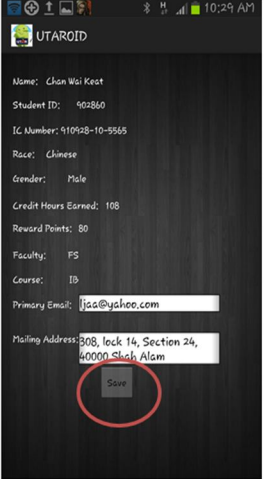
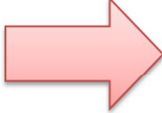



a) Login Procedure



1) Login screen will be displayed whenever UTARoid application is launched. Enter Student ID and Password before clicking on the Login button.

2) If entered Student Id and Password matches / is found, user will be navigated to the Home Screen with all his/her basic information and profile pic displayed in the upper frame.

b) Edit Profile Information Procedure

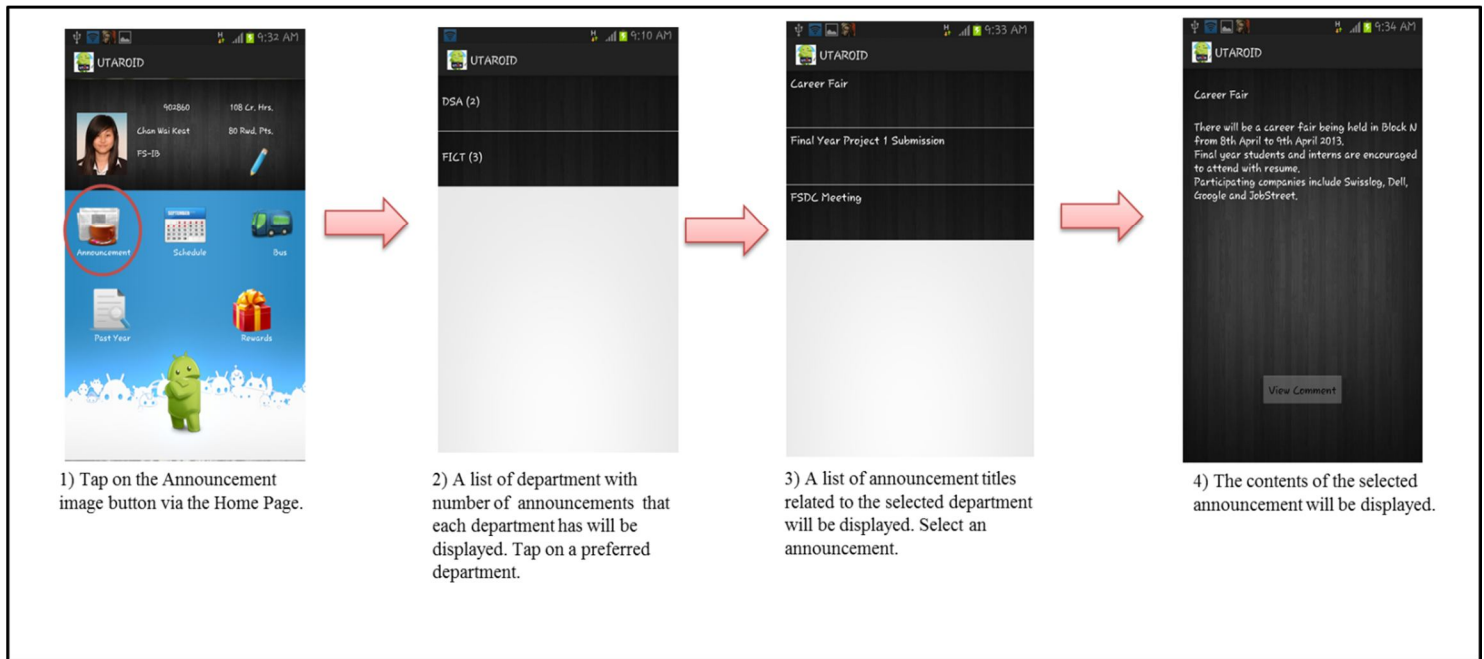


1) Tap on the Edit Profile button located at the upper frame.

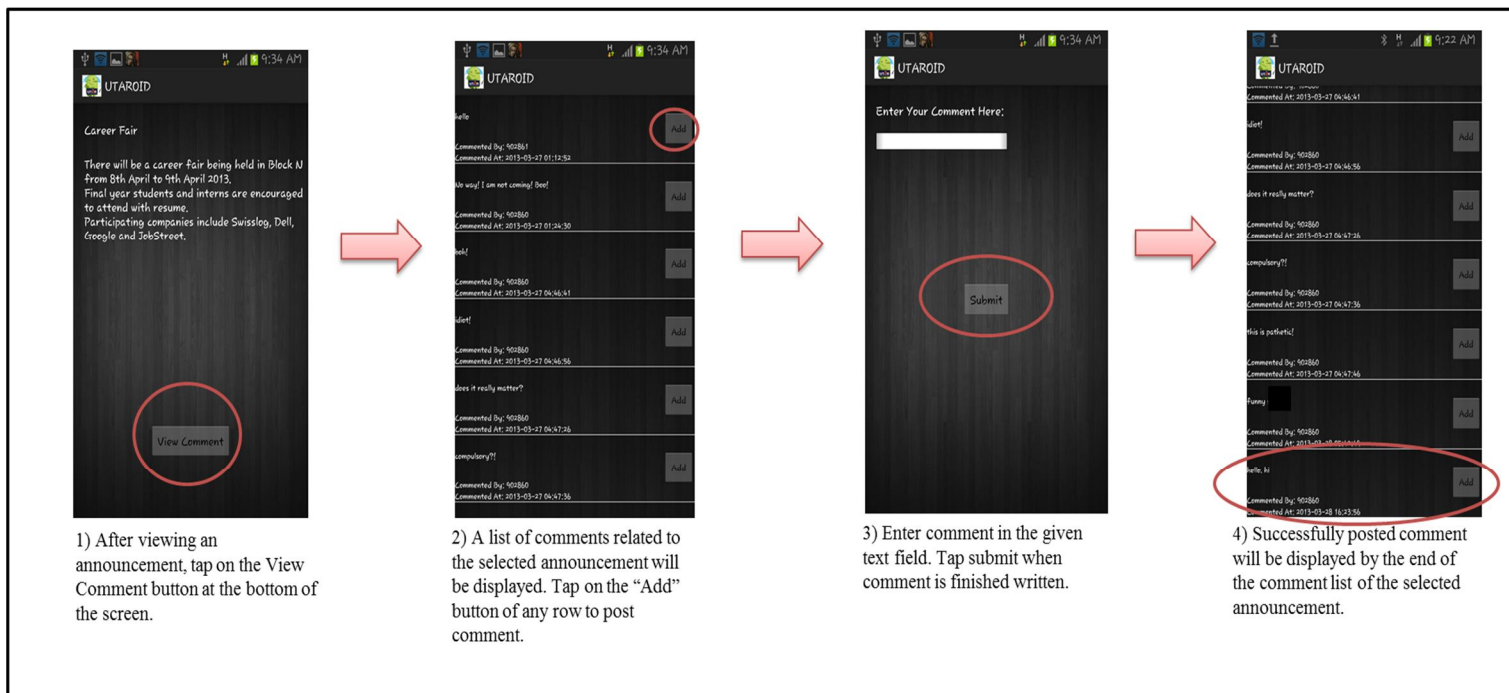
2) Profile information displayed. You are only allowed to edit certain information such as E-Mail Address and Mailing Address.

3) Click the Save button as changes are done. A message box will be displayed informing user if changes is successfully saved or otherwise.

c) View Announcement Procedure



d) View & Post Comment Procedure



e) View Event Procedure

1) Tap on the Schedule image button via the Home Page.

2) In the Schedule Selection screen, tap on University Events.

3) A list of current and upcoming events will be displayed. Each row contains the name of the event, description and the date of the event.

f) View Daily Time Table Procedure


1) Tap on the Schedule image button via the Home Page.

2) In the Schedule Selection screen, tap on Semester Time Table.

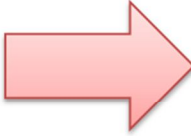
3) Select a day from the list.

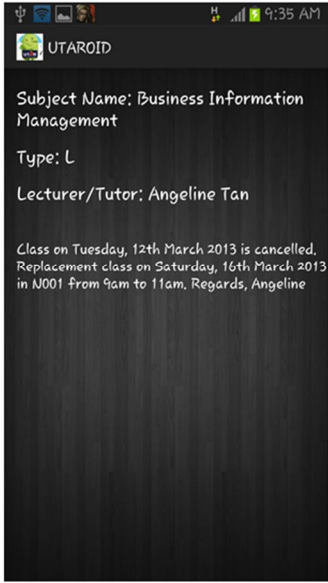
4) A list of registered classes for the selected day will be displayed in order of time. Contents include subject name, type of class, start and end time, venue and name of lecturer/tutor.

g) View Class Announcement Procedure




1) In the list of registered classes for the selected day, tap on the class which you wish to read announcement related to that class.



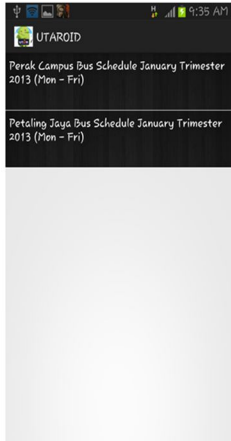


2) The announcement related to the selected class will be displayed.

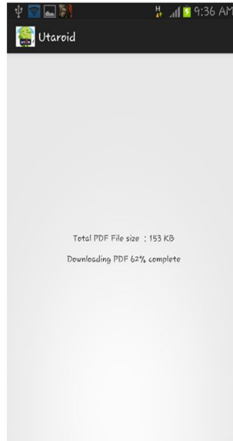
h) Download and View Bus Schedule Procedure



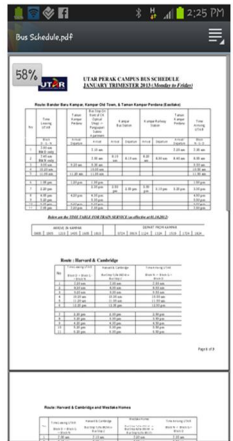
1) Tap on the Bus image button via the Home Page.



2) Select a bus schedule file that you wish to download from the list.

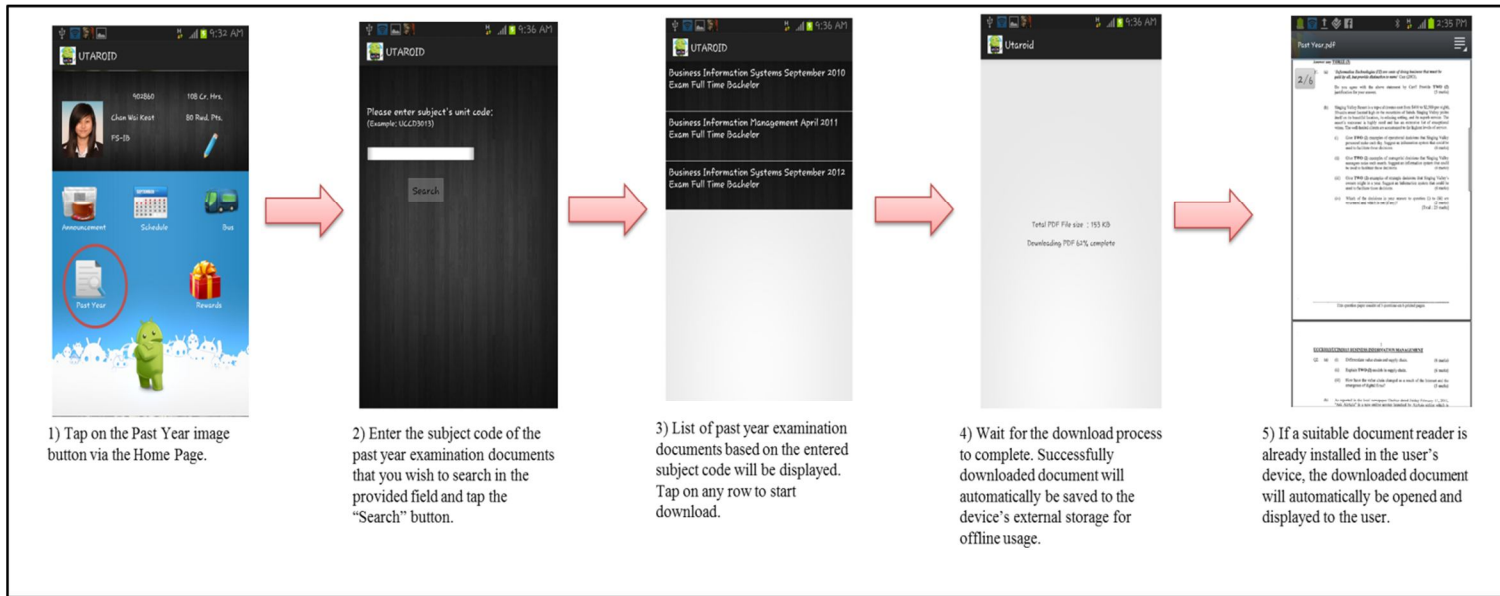


3) Wait for the selected file to be fully downloaded. Complete download of a file will automatically be saved in device's external memory.

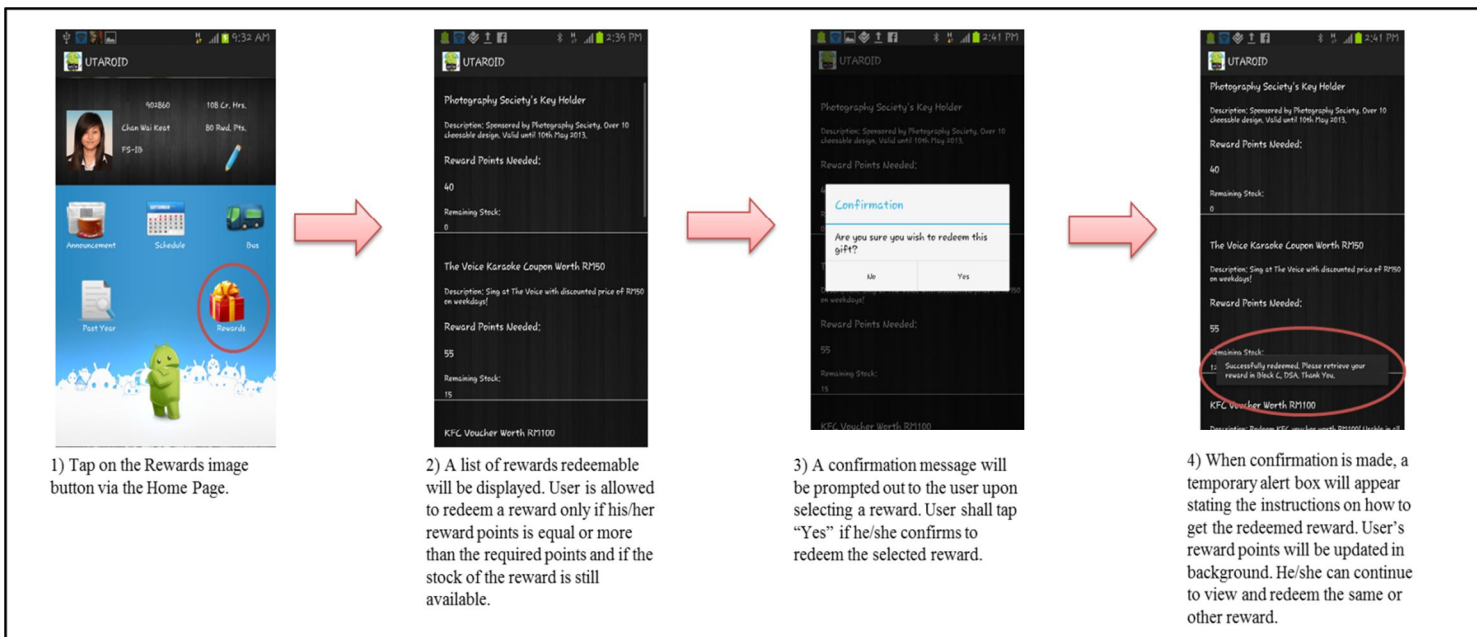


4) If user's device is installed with a suitable document reader, the downloaded file will automatically be opened and displayed to the user.

i) Search, Download and View Past Year Examination Procedure



j) Reward Redemption Procedure



APPENDIX E: TURNITIN REPORT