

PERPUSTAKAAN UMP

SM

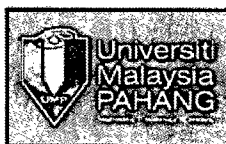


0000103266

NURSAIFULLAH BIN MAT NOR @ ABD RAHIM

103266

UNIVERSITI MALAYSIA PAHANG



UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS

JUDUL: SMART COMPLAINS FOR ANDROID

SESI PENGAJIAN:

SAYA NURSAH FULLAH BIN MAT NOR (HURUF BESAR)

Mengaku membenarkan tesis/laporan PSM ini disimpan di Perpustakaan Universiti Malaysia Pahang dengan syarat-syarat kegunaan seperti berikut:

1. Tesis/Laporan adalah hakmilik Universiti Malaysia Pahang.
2. Perpustakaan Universiti Malaysia Pahang dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institut pengajian tinggi.
4. **Sila tandakan (✓)

SULIT

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) *

TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan) *

TIDAK TERHAD

ROMLI

Alamat tetap: 72 KAMPUNG RAHMAT
21450, SETIU,
TERENGGANU

Tarikh: 2/01/2015

Disahkan Oleh

Penyelia
RAHIWAN NAZAR BIN ROMLI
LECTURER
FACULTY OF COMPUTER SYSTEMS & SOFTWARE
ENGINEERING
UNIVERSITI MALAYSIA PAHANG
LEBUHRAYA TUN RAZAK, 26300 GAMBANG,
KUANTAN, PAHANG

TEL: 09-549 2134 FAX: 09-549 2144
Tarikh: 2/1/15

*Sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis/laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

SMART COMPLAINS FOR ANDROID

NURSAIFULLAH BIN MAT NOR @ ABD RAHIM

THESIS SUBMITTED IN FILFILMENT OF THE DEGREE OF COMPUTER SCIENCE
(COMPUTER SYSTEMS AND NETWORKING)

FACULTY OF COMPUTER SYSTEMS AND SOFTWARE ENGINEERING

2014

DECLARATION

I declare that this thesis entitled "Knowledge Repository for Ministry of Education" is the result of my own research expert as cited in reference. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

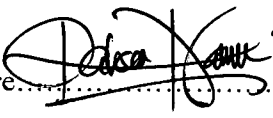
Signature : nrmm

Name : NURSAIFULLAH BIN MAT NOR @ ABD RAHIM

Date : 22 DECEMBER 2014

SUPERVISOR DECLARATION

I hereby declared that I have read this thesis and in my opinion this thesis/report is sufficient in term of score and quality for the award of the degree of Bachelor of Computer Science (Computer Systems and Networking)

Signature.....

Supervisor Name.....

Date.....2/1/15.....

RAHIWAN NAZAR BIN ROMLI
LECTURER
FACULTY OF COMPUTER SYSTEMS & SOFTWARE
ENGINEERING
UNIVERSITI MALAYSIA PAHANG
LEBUHRAYA TUN RAZAK, 26300 GAMBANG,
KUANTAN, PAHANG
TEL : 09-549 2134 FAX : 09-549 2144

ACKNOWLEDGEMENTS

First and foremost I praise be to the Almighty Allah for all His blessings for giving granting me with patience, guidance and good health throughout the duration of this Bachelor research.

I am very fortunate and thankful to have Mr. Rahiwan Nazar bin Romli as my research supervisor for all of his kind advices and thoughts. Without him I would not be able to carry this research. Also, I would like to express my deepest and highest appreciation to all lecturers, staffs and friends who have been giving me guidance and motivation throughout the completion of this project. Last but not least, I am so blessed to have my family who has been showering me with love and endless support.

ABSTRACT

This project is to design an Android application that will act as a medium for the students to lodge reports related to any dissatisfaction on the services in the campus and so forth. For example, a student found that the meal he bought from the cafeteria is molded or one of the equipment in the gym has broken, so he can simply just take a photo of it and upload it into the system. This project will be developed using Eclipse. This project is chosen because I find it as a very interesting project. Furthermore, it is also very practically feasible to construct, practically applicable and useful and has the potential to be marketed. In addition, it is different from other existing similar systems in the market. Java programming is the language used in developing Smart Complains. It was chosen because it gives more structured approach rather than the conventional way. The reports which have been uploaded will be managed in a web application designed for the administrators. The final product of this project is a fully functional application that achieves the initial objectives of this project and at the same time illustrates the practical feasibility of our idea. However, since the main objective is only to illustrate the feasibility of our idea, therefore the specification and quality of certain components have been compromised to reduce the project cost. But, in the real life situation, the specifications of certain components have to be changed for optimum and satisfactory performance.

ABSTRAK

Projek ini adalah untuk merangka satu aplikasi Android yang akan bertindak sebagai pengantara untuk para pelajar membuat aduan atas ketidakpuasan terhadap perkhidmatan-perkhidmatan yang telah disediakan di kampus dan sebagainya. Sebagai contoh, seorang pelajar menemui bahawa makanan yang dibeli dari kafetaria berkulat dan tidak boleh dimakan lagi, atau salah satu peralatan di gimnasium rosak, dia boleh melaporkan hal itu dengan mengambil gambar dan memuatnaik ke dalam sistem. Projek ini akan dibangun menggunakan Eclipse. Projek ini dipilih kerana saya mendapati ianya sangat menarik. Tambahan lagi, ia juga sangat praktikal untuk dilaksanakan, boleh dipraktikkan dan berpotensi untuk dipasarkan. Di samping itu, ianya berbeza daripada sistem-sistem yang sama yang telah ada. Laporan-laporan yang telah dimuat naik akan diuruskan di dalam aplikasi laman sesawang yang telah dibina untuk pentadbir. Produk terakhir projek ini ialah satu aplikasi yang berfungsi sepenuhnya dan mencapai semua objektif dan tujuan yang telah ditetapkan. Bagaimanapun, memandangkan objektif utama adalah hanya untuk mengilustrasi idea secara praktikal, spesifikasi dan beberapa komponen tertentu tidak digunakan untuk mengurangkan kos projek. Tetapi, dalam situasi kehidupan yang sebenar, spesifikasi sesetengah komponen harus ditukar untuk prestasi yang optimum dan menepati citarasa pengguna.

TABLE OF CONTENTS

		PAGE
	DECLARATION	IV
	SUPERVISOR DECLARATION	V
	ACKNOWLEDGEMENTS	VI
	ABSTRACT	VII
	ABSTRAK	VIII
	CONTENTS	IX
	LIST OF TABLES	XII
	LIST OF FIGURES	XIII
CHAPTER I	INTRODUCTION	
1.1	INTRODUCTION AND MOTIVATION	1
1.2	PROBLEM STATEMENT	2
1.3	OBJECTIVE	3
1.4	SCOPE	3
1.5	THESIS ORGANIZATION	3
CHAPTER II	LITERATURE REVIEW	
2.1	INTRODUCTION	5
2.2	OVERVIEW OF SMART COMPLAINS	6
	2.2.1 WHAT IS SMART COMPLAINS	6
2.3	BENEFITS OF AUTOMATED HOUSE	7
2.4	EXISTING SYSTEMS	7
	2.4.1 MANUAL REPORT LODGING	7
	2.4.2 ANDROID SOFTWARE DEVELOPMENT	7
	2.4.3 WAMP SERVER	8
	2.4.4 E-ADUAN MAJLIS PERBANDARAN PULAU PINANG	8

2.5	COMPARISON	10
2.6	CONCLUSION	11
CHAPTER III METHODOLOGY		
3.1	INTRODUCTION	12
3.2	PROJECT INITIATION AND PLANNING	14
3.3	ANALYSIS	14
3.4	SYSTEM DESIGN	15
3.5	TESTING AND DEBUGGING	15
3.6	SOFTWARE AND HARDWARE REQUIRED	16
	3.6.1 SOFTWARE	16
	3.6.2 HARDWARE	16
3.7	CONCLUSION	17
CHAPTER IV DESIGN AND IMPLEMENTATION		
4.1	IMPLEMENTATION	18
4.2	PROCESS FLOW	18
4.3	SYSTEM ARCHITECTURE	20
4.4	CODING	21
	4.4.1 ANDROID APPLICATION	21
	4.4.2 WEB APPLICATION	25
4.5	TESTING	25
4.6	CONCLUSION	25

CHAPTER V	RESULTS AND DISCUSSION	
5.1	INTRODUCTION	28
5.2	EXPECTED RESULTS	28
5.3	RESULTS OF THE SYSTEM	29
5.4	SYSTEM LIMITATION	34
	5.4.1 PRIVATE OFFLINE NETWORK	34
	5.4.2 NO ACTUAL SERVER	34
5.5	ADVANTAGES OF SMART COMPLAINS	34
5.6	FUTURE ENHANCEMENT	34
CHAPTER VI CONCLUSION		
6.1	INTRODUCTION	35
6.2	SUMMARY OF LITERATURE REVIEW	36
6.3	SUMMARY OF METHODOLOGY	36
REFERENCES		37
APPENDIX		39

LIST OF TABLES

TABLE NUMBER		PAGE
3.1	SOFTWARE REQUIREMENTS AND PURPOSES	16
3.2	HARDWARE REQUIREMENTS AND PURPOSES	16

LIST OF FIGURES

FIGURE NUMBER		PAGE
2.4.4.1	E-ADUAN MAJLIS PERBANDARAN P. PINANG	9
2.4.4.2	MPPP'S ADUAN BARU	10
3.1	RAD CYCLE	13
4.1	PROCESS FLOW	19
4.2	SYSTEM ARCHITECTURE	20
4.3	REGISTER SERVICE	21
4.4	LOGIN SERVICE	22
4.5	SEND REPORT SERVICE 1	23
4.6	SEND REPORT SERVICE 2	24
4.7	LOGIN SERVICE WEB	27
4.8	REGISTER SERVICE WEB	26
4.9	REPORT SERVICE WEB	26
4.10	UPLOAD SERVICE WEB	27
5.3.1	ANDROID APPS SPLASH SCREEN	29
5.3.2	ANDROID APPS REGISTER PAGE	30
5.3.3	ANDROID APPS LOGIN PAGE	31
5.3.4	WEB APP LOGIN PAGE (ADMIN)	32
5.3.5	WEB APP REPORT LIST	33
5.3.6	WEB APP REPORT DETAILS	33

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION AND MOTIVATION

University Malaysia Pahang's Majlis Perwakilan Pelajar (UMP-MPP) is a committee built from the students' representatives from all faculties. MPP acts as the main entity to every clubs and students organization in University Malaysia Pahang. A total of 23 student representatives have been elected from each faculty to lead the student's leadership.

The vision of MPP's establishment is to espouse the university in producing graduates who are competent in accordance with UMP's Strategic Plan. There are four missions stated which are; raising the leadership with integrity towards management practices and excellent work culture, empowering the humanity values covering all aspects of students development in producing skilled graduates, creating intelligent students in facing the challenges of the competitive world. And the last one would be to sustain a culture of national integration in campus life.

Furthermore, establishment of the MPP is to act as the voice of the students. In other words, students will be able to voice out their reports regarding any unsatisfied with the services

provided in UMP. For example, one of the equipments in the gymnasium is broken, and then the students can make a report to the MPP so that further actions can be taken.

According to the current manual system, a student will have to go to the MPP office and fill out a report form before submitting the report. So, it will require some effort.

Smart Complains for Android is a new approach to be introduced to ease the student in making reports to the student council. It is an easy-to-use platform where a student can just simply make a report to the MPP by just uploading a photo related to the report along with details of the report. Then the report will be stored in a database and there will be a web system where to act as the medium for the council members to manage the reports.

1.2 PROBLEM STATEMENT

Manually, students whom want to make a report will have to go to the MPP's office. Then they will have to fill out forms regarding the report.

Three main problems that the students are currently facing with this traditional manual report lodging system are:

1. Students will have to make the effort to go to the office.
2. Reports cannot be done during weekends.
3. Possibility to lose the report forms during processing.

1.3 OBJECTIVE

This project was conducted to meet four objectives. The objectives of this research are:

- To develop an android application.
- To upload photos onto the MPP's database.
- To manage the database.
- To test the system developed based on case study.

1.4 SCOPE

The scopes of Smart Complains for Android Smartphone are as listed;

1. Using Android operating system as the main platform for developing Smart Complains.
2. Register and login functions on the android application.
3. Upload and store reports into the database.
4. Manage the database (view, find, delete, update reports).
5. Print reports through printers.

1.5 THESIS ORGANIZATION

This thesis will have a total of six chapters ranging from chapter one until six. In chapter one, discussion is more on to introduction to this project which includes problem statement, objectives and scopes of this project.

In chapter 2, literature review will be discussed. Discussion will be made about current system that exist, their advantages and disadvantages, and problems with current system

In chapter 3, methodology chosen for this project will be discussed. The requirements for hardware and software also will be identified and discussed.

Chapter 4 is a design and implementation phase discussion. In this chapter, the project implementation will be shown including system flow; coding and developing process will be explained in more details.

Result and discussion will be shown in chapter 5. Expected results and actual results will be in this chapter together with future enhancement to the project.

Chapter 6 is the final conclusion for this project. Conclusion will include methodology and literature review.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter briefly reviews some future works and researches related to the proposed project. Smart Complain for Android is a android mobile application that is developed to help users which are the University Malaysia Pahang's students to lodge reports and make complains regarding any dissatisfaction on the services provided in UMP or broken/stolen equipments or facilities.

A mobile application is a computer program designed to run on smart phones, tablet computers and many more mobile devices. Apps are usually available through application distribution platforms, which began appearing in 2008 and are typically operated by the owner of the mobile operating system, such as the Apple App Store, Google Play, Windows Phone Store and Blackberry App World.

Mobile apps were originally offered for general productivity and information retrieval, including email, calendar, contacts and stock market and weather information. However, public demands and the availability of developer tools drove rapid expansion into other categories, such as mobile games, factory automation, GPS and so forth.

2.2 OVERVIEW OF SMART COMPLAINS

2.2.1 WHAT IS SMART COMPLAINS?

Smart Complain is an android application which has been developed to provide a new medium/platform to the students of University Malaysia Pahang in lodging their reports/complains for any damaged facilities in the campus or unsatisfied services. For example, a student finds that the meal that he bought from the cafeteria is rotten. So, he wants to report about it to the student council for actions to be taken. What he has to do is just log into Smart Complain and then take a photo of the rotten meal to be uploaded to the database along with some description of that case. One thing that has to be sure is that, the student must first register to the system. This can be done easily by them.

Once the photo has been uploaded onto the server, the administrators will be able to review the report from a web application that will also be developed in this project. From the webpage, administrators can also see the status of the reports whether the actions have been taken or not.

2.3 BENEFIT OF SMART COMPLAINS

The advantages of implementing Smart Complains can be illustrated as follows:

1. Users can simply register themselves using their Student Identification Number. But only one user can be registered under one Identification Number.
2. Simple User Interface eases users to use the application.
3. Photo upload feature makes the report more truthful.
4. Web system for the administrators to manage reports.

2.4 EXISTING SYSTEM

Currently, there are some applications and systems in the market that have been implemented with the same kind of concept. Many studies and researches have been done to enhance the current system with adapting the most recent and latest technology.

2.4.1 MANUAL REPORT LODGING

The manual report lodging is the old-fashioned way of making a report. Basically, a person or in this case, a student would have to make some effort. Say, he wants to report about missing equipment from the gymnasium, he would then have to go the student council's office during the office hours. After that, he will have to fill out forms describing his complain. Finally, wait for the further actions to be taken. However, there is chance that complains would be dropped as from lost.

2.4.2 ANDROID SOFTWARE DEVELOPMENT

Android software development is the process by which new applications are created for the Android operating system. Applications are usually developed in the Java programming language using the Android Software Development Kit, but other development tools are also available.

As of July 2013, more than one million applications have been developed for Android. With over 25 billion downloads. A June 2011 research indicated that 67% of mobile developers used the platform, at the time of publication. In Q2 2012, around 105 million units of Android smartphones were shipped which acquires a total share of 68% in overall smartphones sale.

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computer running Linux, Mac

OSX 10.5.8 or later and Windows XP or later, for the moment one can also develop Android software on Android itself using the AIDE-Android IDE – Java – C++ app and the Java editor app.

The officially supported integrated development environment (IDE) is Eclipse using the Android Development Tools (ADT) Plugin.

2.4.3 WAMP SERVER

WAMP server is a Windows web development environment. It allows you to create web applications with Apache, PHP and the MySQL database. It also comes with PHPMyAdmin to easily manage your databases.

WAMP server installs automatically (installer), and its usage is very intuitive. You will be able to tune your server without even touching the setting files. It is the only packaged solution that will allow you to reproduce server. Once WAMP server is installed, you have the possibility to add as many Apache, MySQL and PHP releases as you want. It also comes with tray icon to manage your server and its settings.

2.4.4 E-ADUAN MAJLIS PERBANDARAN PULAU PINANG (MPPP)

This portal was established so that it would help people to be in touch with the MPPP electronically, without boundaries and limits. Here, the people can lodge any feedback, reports and complaints, questions and also their suggestions.

For the first users of the portal, they have to register as Rakan e-Aduan before they can proceed. Then they can continue and choose the services provided in the portal.

Figure 2.3.4.1 below shows a screenshot of MPPP's e-Aduan.

LAMAN UTAMA
 LOGIN MPPP
 AKTIVITI

MAJLIS PERBANDARAN PULAU PINANG

e-Aduan MPPP

Cepat • Mudah • Efektif

Rakan e-Aduan
 Aduan Baru
 Semak Aduan
 Tatacara SMS

BM | EN

Selamat Datang ke Portal e-Aduan Majlis Perbandaran Pulau Pinang (MPPP)

Portal e-Aduan MPPP diwujudkan untuk memudahkan orang ramai berinteraksi secara elektronik dengan MPPP, tanpa had dan tanpa sempadan waktu dan jarak. Laporkan maklumbalas anda sama ada aduan, pertanyaan, cadangan atau penghargaan di sini.

Sekiranya anda pertama kali melawat portal ini, anda boleh memilih untuk mendaftar sebagai Rakan e-Aduan dengan [KLIK DISINI](#). Selepas itu, sila [LOGIN](#) ke Portal e-Aduan dan memilih perkhidmatan yang telah disediakan.

Sekiranya tidak berminat, anda boleh terus laporkan [Aduan Baru](#) anda atau [Semak Aduan](#) anda dengan klik butang-butang menu di atas. Bagi semakan aduan yang telah dilaporkan sebelum 31 Disember 2010, sila semak [DI SINI](#).

Terima kasih kerana sudi melawat Portal e-Aduan MPPP.

Statistik Aduan Awam Aduan mengikut KATEGORI [tahun ini]

ADUAN MENGIKUT KATEGORI [2014]

Hubungi Kami

Majlis Perbandaran Pulau Pinang
Paras 4, KOMTAR
Jalan Penang
10675 Pulau Pinang

Figure 2.4.4.1 MPPP's e-Aduan

Aduan Baru

Laporkan semua permasalahan dan pengaduan anda yang berkaitan dengan MPPP dengan mengisi maklumat di bawah. Setiap pengaduan yang dibuat akan diberikan perhatian dan tindakan yang sewajarnya.

Aduan Baru

Panduan Ringkas

Untuk mendaftar aduan baru, sila masukkan butiran diri anda serta maklumat sokongan yang lain seperti yang disyaratkan.

Untuk semakan status aduan melalui SMS, daftarkan nombor telefon bimbit anda.

Adalah medan mandatori.

Untuk semakan status aduan menerusi portal web ini, daftarkan samada No. Kad Pengenalan diri atau alamat e-mel atau nombor telefon bimbit anda.

Nama*

Alamat*

No. Kad Pengenalan* (cth: 801122034455)

Alamat E-mel*

Mobile (SMS)* (cth: 0193200160)

Telefon

Fax

Keturunan: Lain-lain

Jantina: Lelaki Perempuan

Kategori Pelanggan: Lain-lain

Seterusnya Batal

Figure 2.4.4.2 MPPP's Aduan Baru page

Figure 2.3.4.2 show a screenshot of MPPP's page where users will use to lodge reports and complaints, questions, feedbacks and recommendations. From here we can only see a web application to do this process. This process can be eased up if there was also a mobile application.

2.5 COMPARISON

MPPP's e-Aduan and Smart Complains both share a similar goal; to provide a convenience solution to report lodging. However, I must say that, Smart Complains has the better approach since it involves mobile computing. As far as we know, mobile computing has become very practical these days.

2.6 CONCLUSION

Based on my observation, Smart Complains for Android phones is a good project to be conducted.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

In this chapter, the report will be discussing about the methodology that will be use throughout this project development. Rapid Application Development (RAD) has been chosen and it consists of four phases; project initiation and planning, analysis, system design and testing and debugging. It is very simple to be understood and used. In RAD, during development process, customer will give some input and feedback that can be used to increase the quality of the project.

This project is about developing an Android mobile application and it will be using Eclipse as the main development platform.

Below are several advantages that come with this model:

- Easy to be used and easily understandable.
- Reduce overall project risk.
- Suitable for small and large project.
- Increase quality of project.

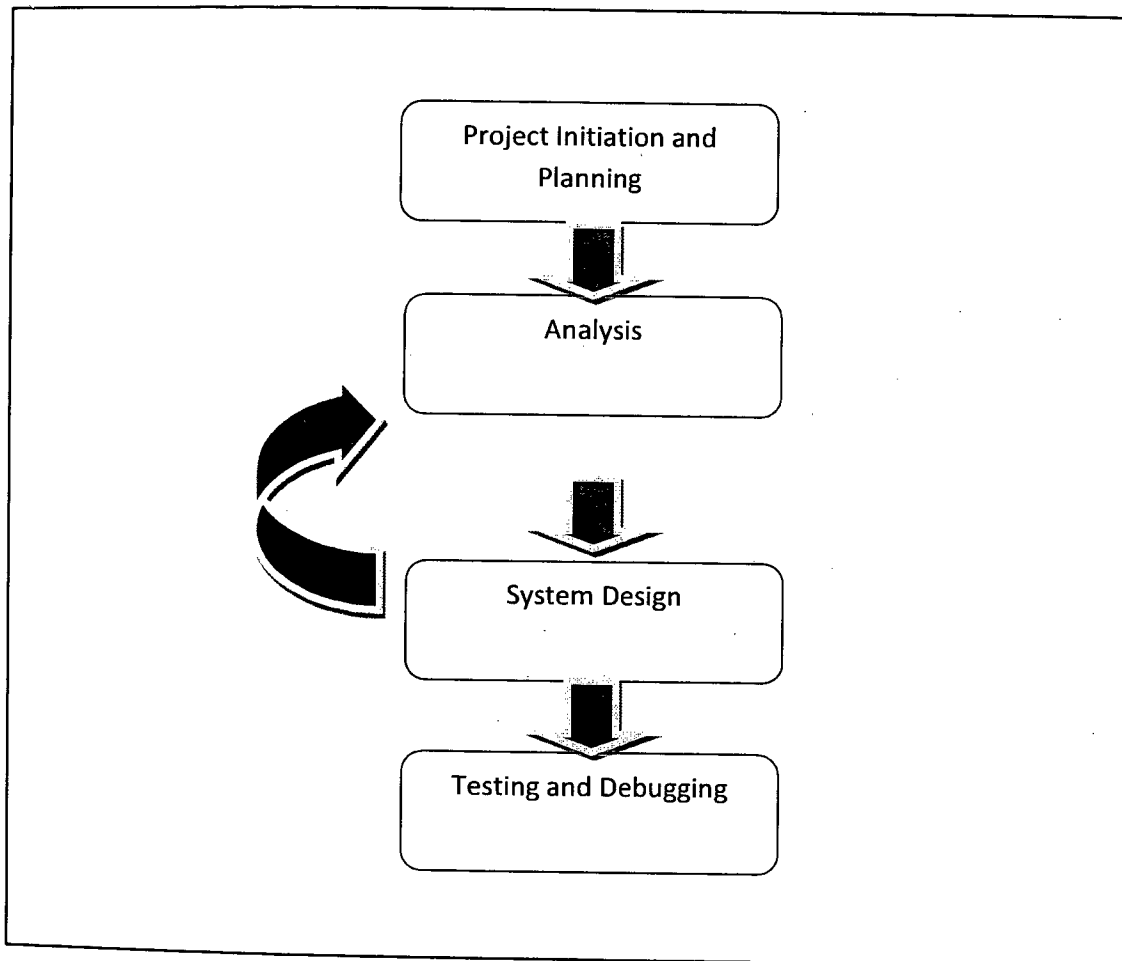


Figure 3.1: RAD Cycle

3.2 PROJECT INITIATION AND PLANNING

In this stage, there will be a meeting and discussion with potential customer and project developer, in order to gather information related to the project's requirement and the idea on how to design the system. This discussion needs to be carried very thoroughly so that there will be no misunderstanding and there will be no mistakes during project development. During this stage, system developers will give the whole explanation to the potential customer.

3.3 ANALYSIS

This analysis phase will gather and define all of the project requirements in a requirement specification document. It should be clear on what to develop by this time and the basic understanding about this project must be clearly understood. Smart Complains was chosen because recently everything needs an easier solution to do it. However, Smart Complains is not going to fully eliminate the traditional method but this project is to improve the existing process.

The objectives of this project need to be fixed in order to make sure that this project will achieve the desired goals or else it would not be able to be considered as a success. There are still a lot more challenges we have to face for this system to be fully implemented. It would take time. Hence, the objective of this project is to explore the concept and the benefits of using Android platform in developing Smart Complains.

Because of some constraints that are faced during the project development, project scope must be set. This system is targeting the students as the users.

3.4 SYSTEM DESIGN

This design phase will define all the functional and technical for this project. All of the hardware and software that will be used must be set on this phase. So, all of the requirement set will be made sure to satisfy.

When starting this phase, in order to get the initial overview, we will start using the software. For developing Smart Complains, Eclipse Luna (version) will be installed to design the mobile application for users. Smart Complains will be coded using Java programming language.

Once designing the mobile application has done, we will proceed designing a web based application for the administrators to manage the reports. This program will be needed a database to store the reports.

3.5 TESTING AND DEBUGGING

In this phase, all components must be ensured that they are error-free and meet all of the requirements stated in the first step. There is one team called quality assurance team that will do several test cases in order to evaluate whether the product/project follows the requirement needed. Sometimes, this phase will also called post implementation phase.

3.6 SOFTWARE AND HARDWARE REQUIRED

3.6.1 SOFTWARE

SOFTWARE	PURPOSE
Eclipse Juno	Android development tool to be used to code Smart Complains
Sublime Text	Used to build the web application for administrators
Wamp Server 64bit	To provide the sever environment and also database
Microsoft Office	Documentation
Adobe Reader	To study project related articles
Web browser	To access the internet

Table 3.1 Software requirements and purposes

3.6.2 HARDWARE

HARDWARE	PURPOSE
Acer Aspire V5	To do all the code, testing, debugging and also to act as the server
Samsung Galaxy S4	Test unit for Smart Complains
TP-Link modem	To create a network for the interconnectivity
Printer	To print reports

Table 3.2 Hardware requirement and purposes

3.7 CONCLUSION

In this chapter, it is already explained on methodology and software and hardware requirement. RAD is easy to be understood and developer will find it adaptable and flexible to change. RAD is also suitable for small and large project.

As for software and hardware, all of the requirement has been listed here. This is important in order to make sure this system will be completed on time. Requirements are also important as project goes on, everything needs to be ready to smoothen the processes. We also need to allocate the budget for this based on the requirements.

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 IMPLEMENTATION

In this chapter, the whole process of this project will be showed. In RAD, implementation will be done after development process and implementation phase including the code for the system, system debugging and documentation. The main function of coding in this system is to connect the Smart Complains android application with the server and also the web application.

4.2 PROCESS FLOW

Process flow is the important part where it will visualize the system to the future user before the system can be applied and widely used. It will show step by step processes in system so that anybody who are not familiar with the system, can know more through this process flow. Developer also may know the expected result by learning the process flow. Process flow will also determine whether the objectives of this project have been achieved or not.

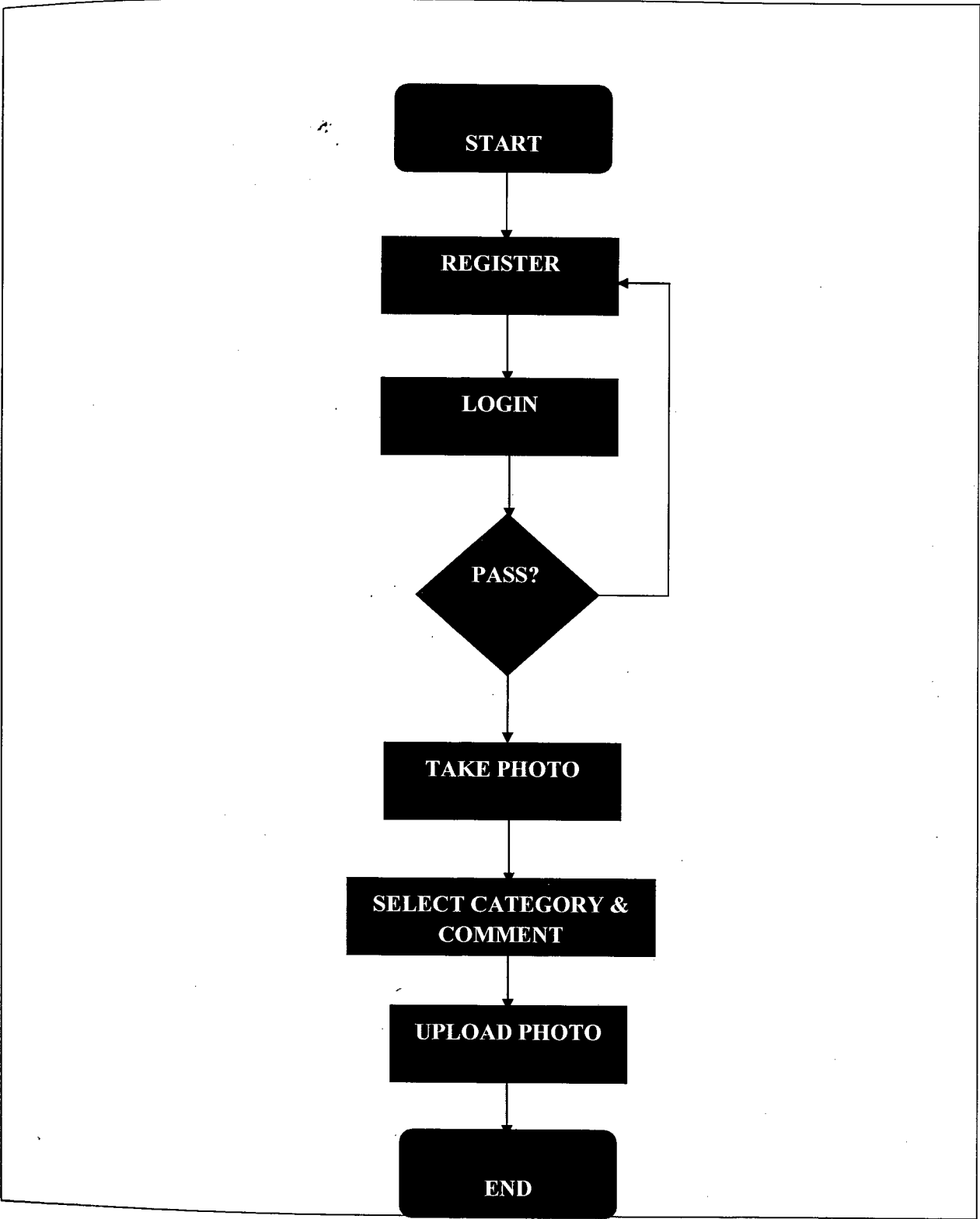


Figure 4.1: Process Flow

4.3 SYSTEM ARCHITECTURE

In this project, an android device which has been installed with Smart Complains needs to be connected to the server so that all of the web services would run successfully. If the system is tested in offline mode, the laptop must and the phone must be in the same network. So, the laptop would have to create a wireless access point to be used.

This system will be using a Samsung Galaxy S4 as the testing device. A laptop which is acting as a server will create a wireless access point so that the device would be able to connect to the server. Then a user has to be registered before he can log into the system. The registration can be easily done himself.

Once he has logged in, he can now capture the photo which he wants to make a report about. After that, he needs to pick a category related on the report. Before he can continue to upload his report into the server, he has to put some brief comment related on the report.

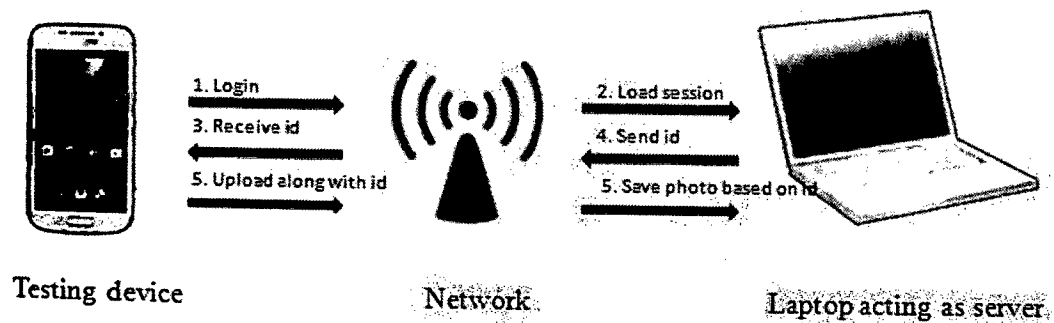


Figure 4.2: System Architecture

4.4 CODING

This section will show the coding of Smart Complains' engine. There are two parts, which are; the web application and the android application.

4.4.1 Android Application

```
@Override
protected void onPreExecute()
{
    super.onPreExecute();
    //do initialization of required objects here
    progressDialog = new ProgressDialog(this.context);
    progressDialog.setTitle("Smart Complains"); //title
    progressDialog.setMessage("Registering.."); // message
    progressDialog.setCancelable(false);
    progressDialog.show();
};

@Override
protected String doInBackground(Void... params) {
    HttpClient httpClient = new DefaultHttpClient();
    HttpContext localContext = new BasicHttpContext();
    HttpPost httpPost = new HttpPost("http://hadi.my/sc/webservice/register.php");
    String text = null;
    try {
        List<NameValuePair> nameValuePair = new ArrayList<NameValuePair>(2);
        nameValuePair.add(new BasicNameValuePair("fullname", this.fullname));
        nameValuePair.add(new BasicNameValuePair("matric", this.matricNo));
        nameValuePair.add(new BasicNameValuePair("password", this.password));
        httpPost.setEntity(new UrlEncodedFormEntity(nameValuePair));

        HttpResponse response = httpClient.execute(httpPost, localContext);

        HttpEntity entity = response.getEntity();
        text = getASCIIContentFromEntity(entity);
    } catch (Exception e) {
        return e.getMessage();
    }
    return text;
}

protected void onPostExecute(String results) {
    if (results != null) {

        Log.d("Hadian", "Jadi dgn result: "+results);
        progressDialog.dismiss();
        //dismiss activity
        AlertDialog.Builder builder;
        builder = new AlertDialog.Builder(context);
        builder.setMessage(results);
        builder.show();
    }
}
}
```

Figure 4.3 Register Service

```

protected String doInBackground(Void... params) {
    HttpClient httpClient = new DefaultHttpClient();
    HttpContext localContext = new BasicHttpContext();
    HttpPost httpPost = new HttpPost("http://hadi.my/sc/webservice/login.php");
    String text = null;
    try {
        List<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(2);
        nameValuePairs.add(new BasicNameValuePair("username", this.username));
        nameValuePairs.add(new BasicNameValuePair("password", this.password));
        httpPost.setEntity(new UrlEncodedFormEntity(nameValuePairs));

        HttpResponse response = httpClient.execute(httpPost, localContext);
        HttpEntity entity = response.getEntity();

        text = getASCIISContentFromEntity(entity);
    } catch (Exception e) {
        return e.getLocalizedMessage();
    }
    return text;
}

protected void onPostExecute(String results) {
    if (results != null) {

        Log.d("Hadian", "Jadi dgn result: "+results);
        progressDialog.dismiss();

        AlertDialog.Builder builder;
        builder = new AlertDialog.Builder(context);

        if(results.equals("Success")){
            //go to report page
            Intent myIntent = new Intent(context, ReportActivity.class);
            myIntent.putExtra("username", this.username);
            context.startActivity(myIntent);
        }else{
            builder.setMessage("Check your username and password");
            builder.show();
        }
    }
}

```

Figure 4.4: Login Service

```

public class SendReportService extends AsyncTask<Void, Void, String> {
    ProgressDialog progressDialog;
    ReportActivity context;
    String username;
    String category;
    String reportText;
    String reportImagePath;

    String reportId;

    public SendReportService(String reportText, String reportImage,
        String username, String category, ReportActivity context) {
        this.context = context;
        this.username = username;
        this.category = category;
        this.reportText = reportText;
        this.reportImagePath = reportImage;
    }

    protected String getASCIIContentFromEntity(HttpEntity entity)
        throws IllegalStateException, IOException {
        InputStream in = entity.getContent();

        StringBuffer out = new StringBuffer();
        int n = 1;
        while (n > 0) {
            byte[] b = new byte[4096];
            n = in.read(b);

            if (n > 0)
                out.append(new String(b, 0, n));
        }
        return out.toString();
    }
}

```

Figure 4.5: Send Report Service 1

```

@Override
protected void onPreExecute() {
    super.onPreExecute();
    // do initialization of required objects here
    progressDialog = new ProgressDialog(this.context);
    progressDialog.setTitle("Smart Complains"); // title
    progressDialog.setMessage("Reporting.."); // message
    progressDialog.setCancelable(false);
    progressDialog.show();
}

@Override
protected String doInBackground(Void... params) {
    HttpClient httpClient = new DefaultHttpClient();
    HttpContext localContext = new BasicHttpContext();
    HttpPost httpPost = new HttpPost(
        "http://hadi.my/sc/webservice/report.php");
    String text = null;
    try {
        List<NameValuePair> nameValuePairList = new ArrayList<NameValuePair>(2);
        nameValuePairList
            .add(new BasicNameValuePair("username", this.username));
        nameValuePairList
            .add(new BasicNameValuePair("category", this.category));
        nameValuePairList.add(new BasicNameValuePair("reportText",
            this.reportText));
        httpPost.setEntity(new UrlEncodedFormEntity(nameValuePairList));
        HttpResponse response = httpClient.execute(httpPost, localContext);
        HttpEntity entity = response.getEntity();
        text = getASCIIContentFromEntity(entity);
    } catch (Exception e) {
        return e.getMessage();
    }
    return text;
}

@SuppressWarnings("NewApi")
protected void onPostExecute(String results) {
    if (results != null) {
        Log.d("Hadian", "Jadi dgn result: " + results);
        if (!results.isEmpty()) {
            reportid = results;
            if (reportImagePath != null) {
                context.runUploadService(reportImagePath, results, progressDialog);
            } else {
                progressDialog.dismiss();
                AlertDialog.Builder builder;
                builder = new AlertDialog.Builder(context);
                builder.setMessage("Report Send Complete");
                builder.show();
            }
        }
    }
}
}

```

Figure 4.6: Send Report Service 2

4.4.2 Web Application

```
<?php
session_start();

ini_set('display_errors',1);

require '../config/database.php';

$username = $_POST['username'];
$password = $_POST['password'];

$sql = "select * from login where username = :username and password = :password";
$params = array(':username' => $username, ':password' => $password);
$stmt = $db->prepare($sql);
$stmt->execute($params);
$data = $stmt->fetchAll(PDO::FETCH_ASSOC);

foreach($data as $row){
    if($row['role'] == 'Client'){
        echo 'Success';
    } else {
        echo 'Fail';
    }
}

if($data == null){
    echo 'Fail';
}

}>
```

Figure 4.7: Login Service Web


```

<?php
session_start();

ini_set('display_errors',1);

require '../config/database.php';
$file_path = "../reportimage/";
$reportid = $_GET['id'];

$file_path = $file_path . basename($_FILES['uploaded_file']['name']);
if (move_uploaded_file($_FILES['uploaded_file']['tmp_name'], $file_path)) {
    $filepathtrim = trim($file_path, "../");
    $sql = 'UPDATE complains SET picture = :url WHERE id = :id';
    $params = array(':id' => $reportid, ':url' => $filepathtrim);
    $stmt = $db->prepare($sql);
    $stmt->execute($params);
    $data = $stmt->fetchAll(PDO::FETCH_ASSOC);
    echo "success";
} else {
    echo "fail";
}

?>

```

Figure 4.10: Upload Service Web

4.5 TESTING

In this process, testing is one part of methodology and it will keep running during development process until the project has been represented to the client successfully. It will be done one code per time, or the whole system, depending on update or upgrade that has or may be made.

4.6 CONCLUSION

This chapter describes the whole process of the development. Conclusively there are two applications that have been developed for this project which are Smart Complains Web; for the administrators and Smart Complains Android; for the users.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 INTRODUCTION

In this chapter, the achievement of this project will be discussed and any feedback regarding this system will also be discussed and changes will be made if developer finds it necessary for the user. The expected outcomes of this system should be met for it to be considered as a success.

5.2 EXPECTED RESULTS

Listed below are the expected results from Smart Complains:

1. Users and administrators will be able to register and log into Smart Complains.
2. The database and the system can be connected and fully functioning.
3. Uploading reports would not take a long time.
4. Reports uploaded can be viewed in the web application (for administrators).

5.3 RESULTS OF THE SYSTEM

For this system, users will need a smart phone which operates on android operating system. This will be their interface or medium to be connected to Smart Complains. Meanwhile, administrators will be using web application to manage Smart Complains. The default username and password have been set to “admin” and “admin1234” for testing.

To be registered, users will need to key in their details with the matric/id number, his full name and a password. The username is not case sensitive which it can be either in capital or small letters.

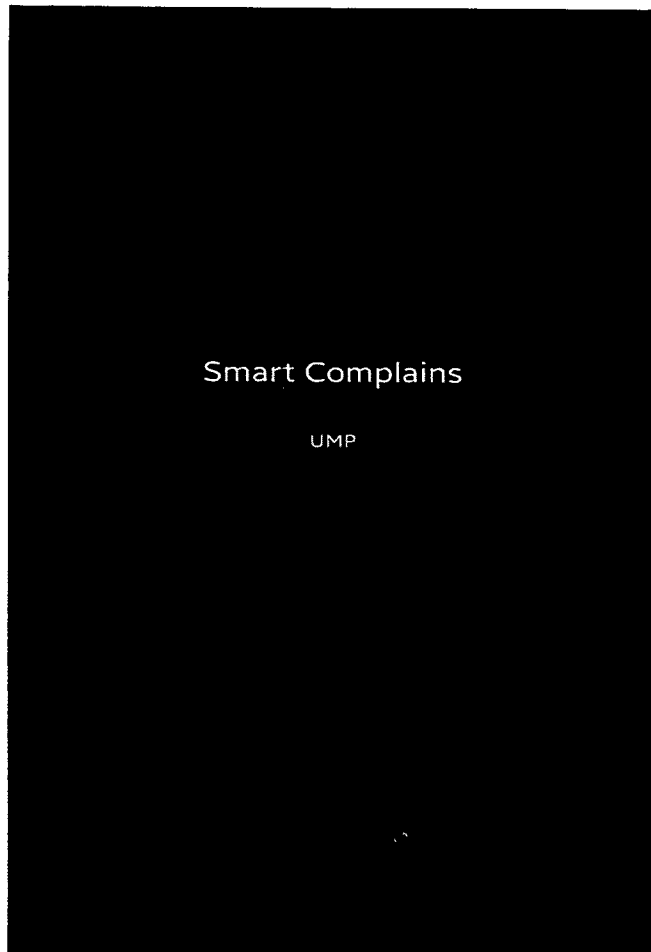


Figure 5.3.1 Android apps splash screen

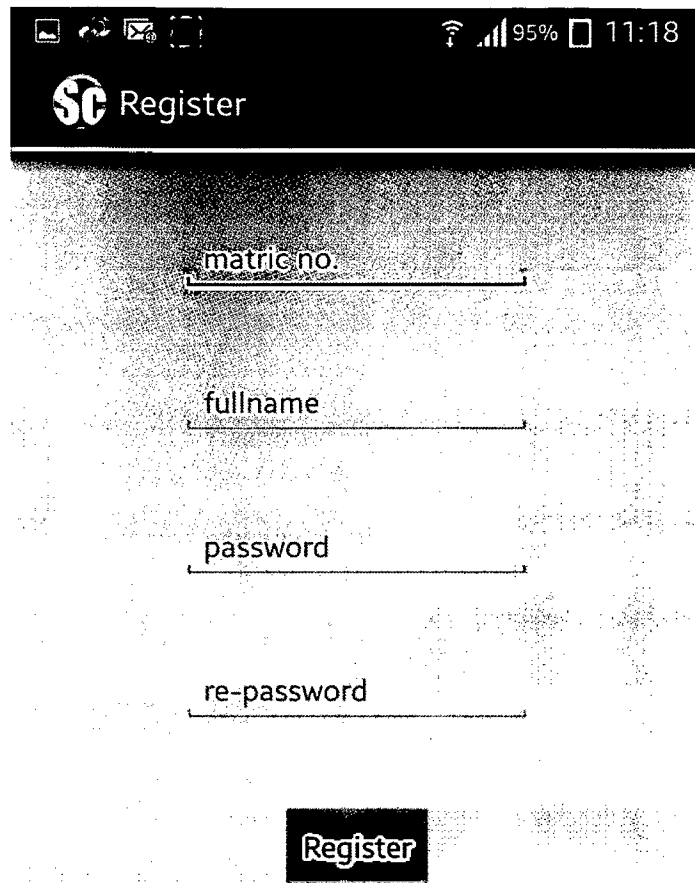


Figure 5.3.2 Android apps register page

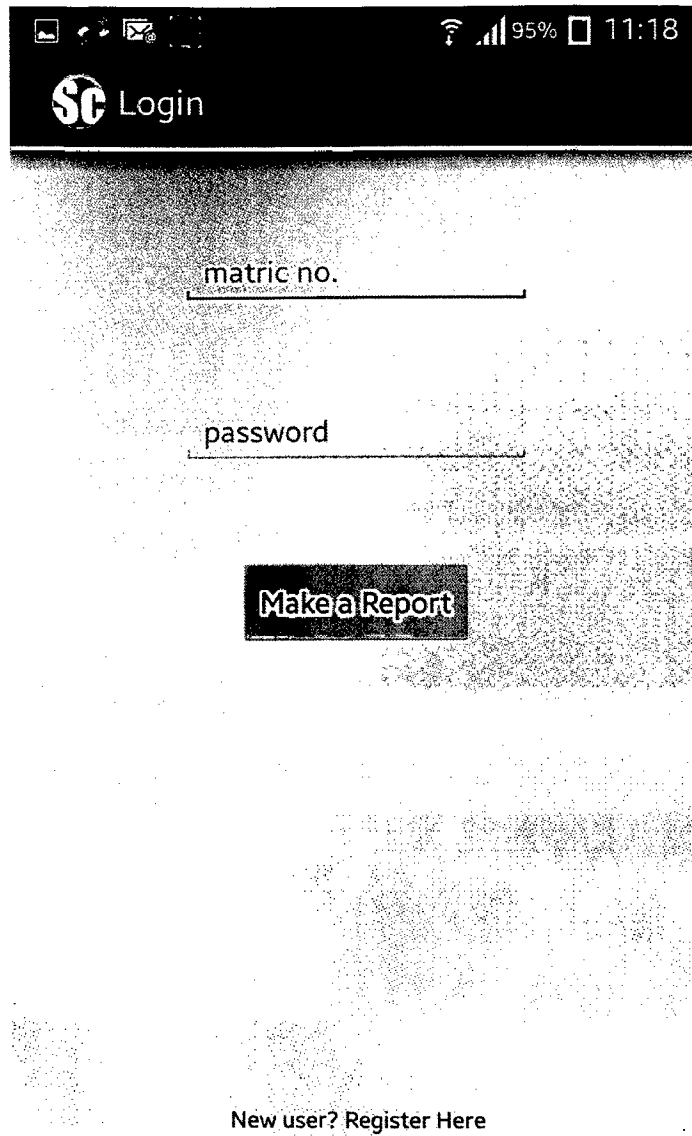


Figure 5.3.3 Android apps login page

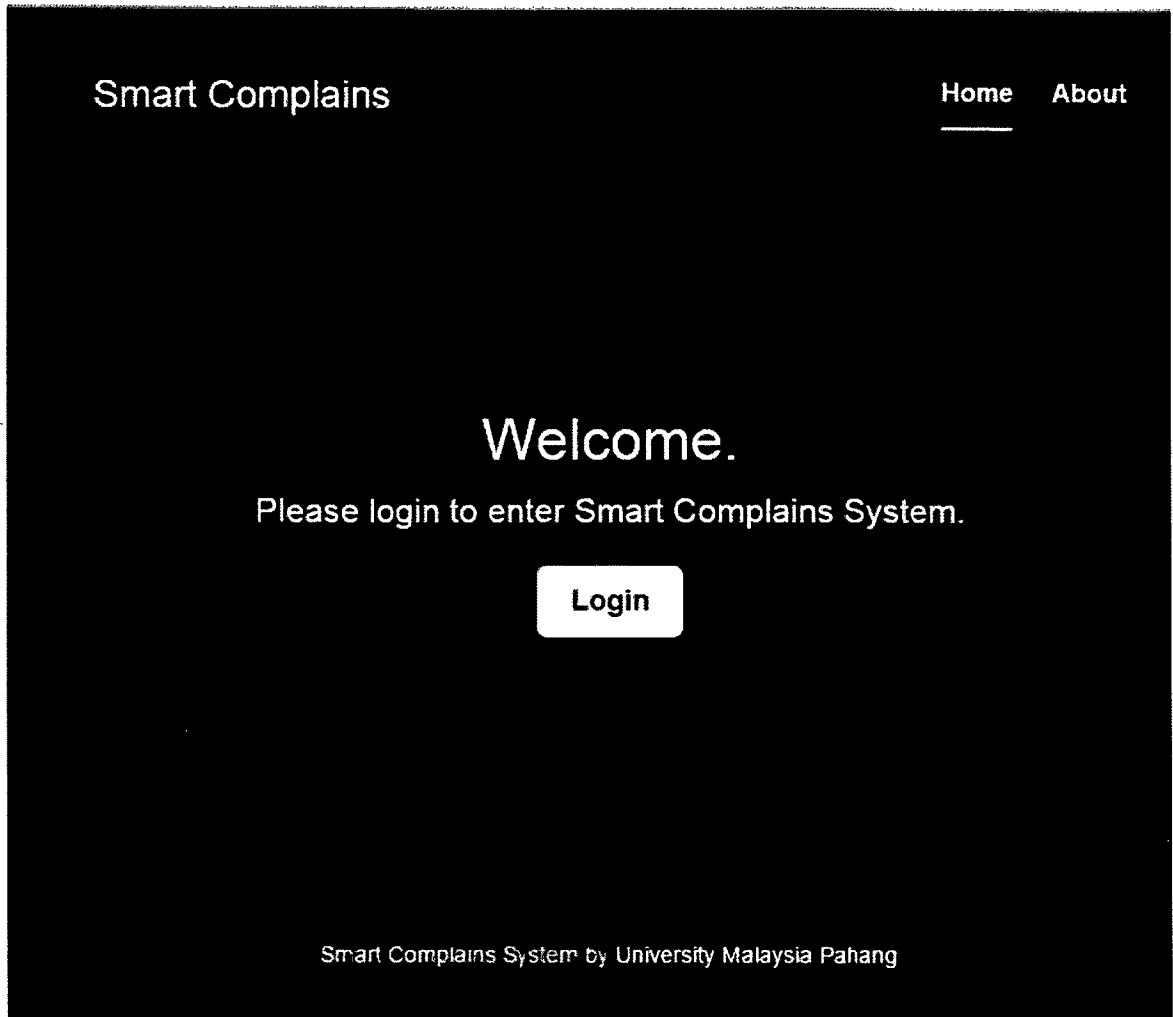


Figure 5.3.4 web app login page (admin)


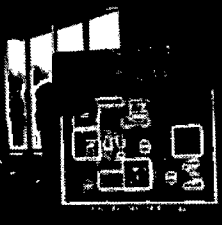
Smart Complains					Logout
No.	From	Date	Category	Picture	Action
1	Nursaifullah Mat Nor (ca11122)	2014-12-08 17:22:53	Peralatan		<input type="button" value="Waiting for action"/> <input type="button" value="Delete"/>
2	Nursaifullah Mat Nor (ca11122)	2014-12-08 17:49:10	Makanan		<input type="button" value="Waiting for action"/> <input type="button" value="Delete"/>
3	Nursaifullah Mat (ca11123)	2014-12-08 16:14:29	Peralatan		Completed ✓ <input type="button" value="Details"/> <input type="button" value="Delete"/>

Figure 5.3.5 web app report list

From : ca11122 - Nursaiyullah Mat Nor Category : Peralatan Date : 2014-12-08 17:22:53



Text :
Haha . ok2 the same as a new job, but the most important things. The list is not responsible. . The may have a great deal of time. I have a look at the top

Action

Comment:

Activity Window

Figure 5.3.6 Web app report details

5.4 SYSTEM LIMITATION

However, there are several limitations or constraints and they may affect the system.

5.4.1 Private Offline Network

For this project, it will only use dummy offline network just to connect the android application to the server on the laptop. For a real world situation, larger network needs to be implemented so that students can lodge reports from anywhere in the university.

5.4.2 No actual server used

One other limitation to this project is that it only uses a laptop as a server. Basically it does the same concept nevertheless, if a real server can be implemented in this project, it would help better in practical exercise.

5.5 ADVANTAGES OF SMART COMPLAINS

1. Remote report lodging system is more practical these days since the technology has become easier to access.
2. Increases the life style in University Malaysia Pahang.
3. It creates a better image for UMP in the eyes of other institutions.

5.6 FUTURE ENHANCEMENT

1. Better UI in android apps.
2. Implement wider online network.
3. Use a real server

CHAPTER 6

CONCLUSION

6.1 INTRODUCTION

This chapter will be the summarization of all of the contents in this system including the summary for the literature review and methodology.

Smart Complains for Android is one of the systems which act as a medium for the students in University Malaysia Pahang to lodge reports from their smart phones providing that they are operating on Android Operating System. For example, there are two categories of reports; the food category and the equipments category.

1. Food
 - a. Smelly food or bad taste
 - b. Dirty environment in the cafeteria
 - c. Unreasonable price
2. Equipments
 - a. Broken gym equipments
 - b. Stolen equipments
 - c. Harmful recreational environment

Listed above are some examples of the reports that can be made. They can also be printed out by the administrators from the database to be passed to technicians or whoever in charge of those matters for further action to be taken.

6.2 SUMMARY ON LITERATURE REVIEW

From the literature review section, developers will know learn more details on the problem statement, scopes of the system and comparison among current systems and this information would be a big help for them in designing their proposed projects. Usually, problems faced by current systems are the main cause that motivates a developer to develop a project.

6.3 SUMMARY ON METHODOLOGY

Methodology is vital to project developers as it will determine on how they are going to construct the project. As for this project, I chose Rapid Application Development (RAD). It has several advantages as it is flexible and adaptable, it reduces project risks and it increases the quality of the project. Rapid Application Development (RAD) involves four phases; Requirement Planning, Analysis, System Design and Testing and finally the Debugging phase. A lot of studies need to be done in order to develop this system. This project only has a very short time to be finished and RAD is flexible and adaptable to any changes that need to be done.

[REFERENCES]

- [1] *Mobile Application Development*. (2014, May 22). Retrieved from Wikipedia:
http://en.wikipedia.org/wiki/Mobile_application_development

- [2] Goodwin, S. (2013). *Smart Home Automation with Linux and Raspberry Pi*.

- [3] *World Economic Outlook*. (2009, April). Retrieved from International Monetary Fund:
<http://www.imf.org/external/pubs/ft/weo/2009/01/weodata/groups.htm#oem>

- [4] Ravenscraft, E. (2013, May 29). *How Can I Get Started with Home Automation?*
Retrieved from Lifehacker:
<http://lifehacker.com/how-can-i-get-started-with-home-automation-510246491>

- [5] Kleback, M. (n.d.). *Tutorial: Raspberry Pi GPIO Pins and Python*. Retrieved from Makezine:
<http://makezine.com/projects/tutorial-raspberry-pi-gpio-pins-and-python/>

- [6] Rossem, G. v. (2009, January 20). *Brief Timeline of Python*. Retrieved from Python History:
<http://python-history.blogspot.com/2009/01/brief-timeline-of-python.html>

- [7] Gayomali, C. (2012, December 3). *The Text Message Turns 20: A Brief History of SMS*. Retrieved from The Week:
<http://theweek.com/article/index/237240/the-text-message-turns-20-a-brief-history-of-sms#axzz33CdC5oVD>

- [8] *Hayes Command Set*. (2014, April 17). Retrieved from Wikipedia:
http://en.wikipedia.org/wiki/Hayes_command_set

- [9] MyPiandMe. (2013, September 13). *Short Message Texting (SMS) with Huawei E220*. Retrieved from My Raspberry Pi and Me:
<http://myraspberrypiandme.wordpress.com/2013/09/13/short-message-texting-sms-with-huawei-e220/>
- [10] Patavalis, N. (n.d.). *Picocom*. Retrieved from Die:
<http://linux.die.net/man/8/picocom>
- [11] *BaudRate*. (n.d.). Retrieved from Math Works:
http://www.mathworks.com/help/matlab/matlab_external/baudrate.html
- [12] WormFood. (2005). *WormFood's AVR Baud Rate Calculator*. Retrieved from WormFood:
<http://www.wormfood.net/avrbaudcalc.php>

APPENDIX A

GANTT CHART

