DIGITAL DOCUMENT STORAGE MANAGEMENT SYSTEM (e-JRM)

NUR HAZWANI BINTI ABU HASSAN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS*

JUDUL: DIGITAL DOCUMENT STORAGE MANAGEMENT SYSTEM (e-JRM)

SESI PENGAJIAN: 2015/2016

Saya NUR HAZWANI BINTI ABU HASSAN

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis dan projek adalah hakmilik Universiti Teknikal Malaysia Melaka.

2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja

3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi 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

(TANDATANGAN PENULIS)
Alamat tetap: Block J 6-4/5,
Kem PGA , 31150 Ulu Kinta,
Ipoh Perak Darul Ridzuan

Tarikh: 25/08/2016

(TANDATANGANPENYELIA)
MDM.SYAHIDA MOHTAR

Tarikh: 25/8/2016.

DIGITAL DOCUMENT STORAGE MANAGEMENT SYSTEM
(e-JRM)

NUR HAZWANI BINTI ABU HASSAN

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Database Management)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY UNIVERSITY TEKNIKAL MALAYSIA MELAKA

2016

DECLARATION

I hereby declare that this project report entitled
DIGITAL DOCUMENT STORAGE MANAGEMENT SYSTEM (e-JRM)
is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : _____ Date : 25/08/2016
(NUR HAZWANI BINTI ABU HASSAN)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of Computer Science (Database Management) With Honours.

SUPERVISOR : _____ Date : 25/8/2016.
(MDM.SYAHIDA MOHTAR)

DEDICATION

I dedicate this report to my parents.
Without their patience, understanding, support, and most of all their love and prayers, the completion of this work would not have been possible.

Besides that, to my PSM supervisor, Madam Syahida Mohtar the advices, comments and support to make sure that this project completed successfully.

I also want to thank to my friends who have always supported me who are directly and indirectly with the PSM.

Thank you.

ACKNOWLEDGEMENTS

First of all, thank to ALLAH for given me a chance and strenght for finishing this Projek Sarjana Muda (PSM) until the end.

I would like to express my deepest gratitude to my supervisor, Madam Syahida Mohtar for all the guidance and help throughout this project.

I would like to extend my thanks to Dr.Akmar Emran as my evaluator and all my fellow friends and to whose help me a lot in completing this project.

ABSTRACT

Digital Document Storage Management System (e-JRM) is a web-based system that helps the users, JRM's staff and management to update the information and document company management . Before this, JRM company use the manual system because they have no computerized system. The manager must fill out a form to make a staff registration,filing, and search document in the cabinet and they need to fill out the form to claim overtime . In order to make the processes within the system running smoothly, Administrator (Manager) is responsible for monitoring all the processes. Four (4) modules have been created on these systems which are registration module, management module and searching and storage module also reporting module. The e-JRM uses an Oracle Database Express Edition and is developed using Hypertext Preprocessor (PHP) programming language, JavaScript and Hypertext Markup Language (HTML).

ABSTRAK

Digital Document Management System Storage (e-JRM) adalah satu sistem berasaskan web yang membantu pengguna, kakitangan dan pengurusan JRM untuk mengemas kini maklumat dan dokumen pengurusan syarikat. Sebelum ini, syarikat JRM gunakan secara manual sistem kerana mereka tidak mempunyai sistem berkomputer. Pengurus mesti mengisi borang untuk membuat pendaftaran kakitangan, fileling dan mencari dokumen dalam kabinet dan mereka perlu mengisi borang keluar untuk menuntut kerja lebih masa. Dalam usaha untuk membuat proses dalam sistem berjalan dengan lancar, Pentadbir (Pengurus) bertanggungjawab untuk memantau semua proses. Empat (4) modul telah diwujudkan dalam sistem ini yang merupakan modul pendaftaran, modul pengurusan dan mencari dan modul penyimpanan juga melaporkan modul. e-JRM menggunakan Pangkalan Data *Oracle Express Edition* dan dibangunkan menggunakan *Hypertext Preprocessor (PHP)* bahasa pengaturcaraan, *JavaScript* dan *Hypertext Markup Language (HTML).*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURE

LIST OF ATTACHMENT

CHAPTER I

INTRODUCTION

1.1 Project Background

JRM is a company that performs variety of services and take tender with several companies from various state. Examples of services provided JRM are chemical services, air conditioning, electrical, facility management and so on.

Some problems occurred, especially for the staff to get the data JRM important document project. As a measure to solve this problem, proposals and suggestions have been made and agreed with the implementation of the Digital Document Management Storage (e-JRM).

The e-JRM is new database systems that will provide an integrated and coordinated policy that focuses on preventing or reducing the problems of storage space for documents, facilitate the search for documents and a safe and systematic disposition. This system is efficient and effective to be used. Implementation of this system is a good effort to reduce the company's document storage space and reduce the problem of search and view also in the disposition of documents. Document management could be done systematically and eventually reduce the duplication information of the document. Manager & staff of JRM Services Sdn. Bhd, also can

search old documents and client's name quickly, view salary details, manage the system and easily track supervisors who manage project based on project location.

1.2 Problem Statement

In JRM company operation tasks performed by JRM, office workers conducted daily in a company. Works on document storage become increasing causing problems on document storage space become limited and consequently. They have to buy new cabinet and new file to solve the document storage problem.

When the storage become limited and file storing become increasing, the document search process becomes difficult. Staffs always complain about this cases or situation. Moreover, deposit document is not systematic and difficult to achieve and disorganized.

In addition, the staff also find difficulty to trace document and supervisor's details who handle the project in different place and to retrieve staff and registration details because currently, JRM is still registering staff manually.

Thus, proposals and suggestions have been made and agreed with the implementation of the Digital Document Management Storage (e-JRM) to solve the current issues or problems.

1.3 Project Objectives

The objectives of this project are to:

- Reduce of document physical storage space.
- Facilitate staff during document searching.
- In order to eliminate anomalies in a different location.
- Generate report in order regarding uploaded document details.

1.4 Project Scopes

1.4.1 USER SCOPE

This project is intended for admin and staff accessing the Digital Document Management Storage (e-JRM).

i) The staff of JRM Services Sdn.Bhd.

- staff cannot access , edit, delete the database in the system because of the users least privilege.
- staff authentication, staff can register, view salary, manage account, upload document, search document, download document.

ii) Administrators of the JRM company

- Controls and have the privilege to access the system to manage system and database.
- Admin authentication, view salary, manage salary, manage staff, upload document, search document, download document.

1.4.2 SYSTEM SCOPE

- Users or staff - attempt to log into and they are required to enter identification number and password. If a matching password is not found in the database users file, the system sends an access-reject popup, which indicates the users that the authentication attempt has failed.

- Administrators – Administrator is the manager who has fully managed & controls the e-JRM system.

- Module :-

i. Login
- Staff and admin must enter identity number and password for login.

ii. Registration
-For a new user who wants to use system must be registered by admin.

iii. System Management
-The administration needs to check the new staff registration and application over time salary updated by staff.

iv. Storage and Searching
-Staff can store and check their wanted document.
-Admin can check the new document, registration details of new staff.

v. Reporting

-The user can see the total documents uploaded into the system.

-The user can know the details documents that have uploaded.

vi. Software

-Using PHP for the system development process

-Using oracle database as the database.

1.5 Project Significance

The staff JRM company can easy to tackling the problem of document storage space, easy to facilitate the search for documents, saves the document in a more systematic and organized, trace project supervisor that handle the project . Staff can use the e-JRM system without worry about the document saving and searching, insert and update their overtime salary also can decrease the habits in using the difficult manually system. The admin can have full management such as easy to manage system, staff information and documents by using the e-JRM system.

1.6 Expected Results

The system is expected to be able to record and monitor the documents from staff ,the registration of staff and the list of project and also a supervisor.By having this digital document management system (e-JRM), this will help staff and manager to keep their record documents in systematically and organized.

This system will also make easy for the user especially for admin to see the claim staff overtime with the calculated overtime. Generate report in the form of a table which counts the total number documents uploaded into the system.

1.7 Conclusion

Staff can storage with the new proposed database system, it is expected that the staff hassle systematically. Staff also can apply overtime claim in computerized system. The new database is developed in an open source environment using ORACLE database and PHP language.

CHAPTER II

PROJECT METHODOLOGY AND PLANNING

2.1 Introduction

This chapter explains about the system development that very important to estimate the time of the system to be delivered on the time. For this Digital Document Storage Management (e-JRM) project, waterfall model will be applied because if there are any problems in any stages, we can always refer to the stage before and make an error correction to further on the next stages. In advance, the waterfall is simple approach and easy to understand and explained phases.

There are stages in waterfall model, which are Analysis, Design, Implementation, Testing, and Maintenance. Every stage will only start if the stage before has been finished or nearly finish. Thus, we decided to use Waterfall model based on Development Life Cycle (DBLC) as our methodology to develop our system.

2.2 Project Methodology

The methodology used to implement this system development is Waterfall Development. Waterfall Model consists of six phase that is Analysis, Design, Implementation, Testing, and Maintenance. Besides, there is a combination with the Database Lifecycle (DBLC), because the main of the project methodology is to synchronize between the system and the databases. Figure 1 shows every phase will be shown as below:

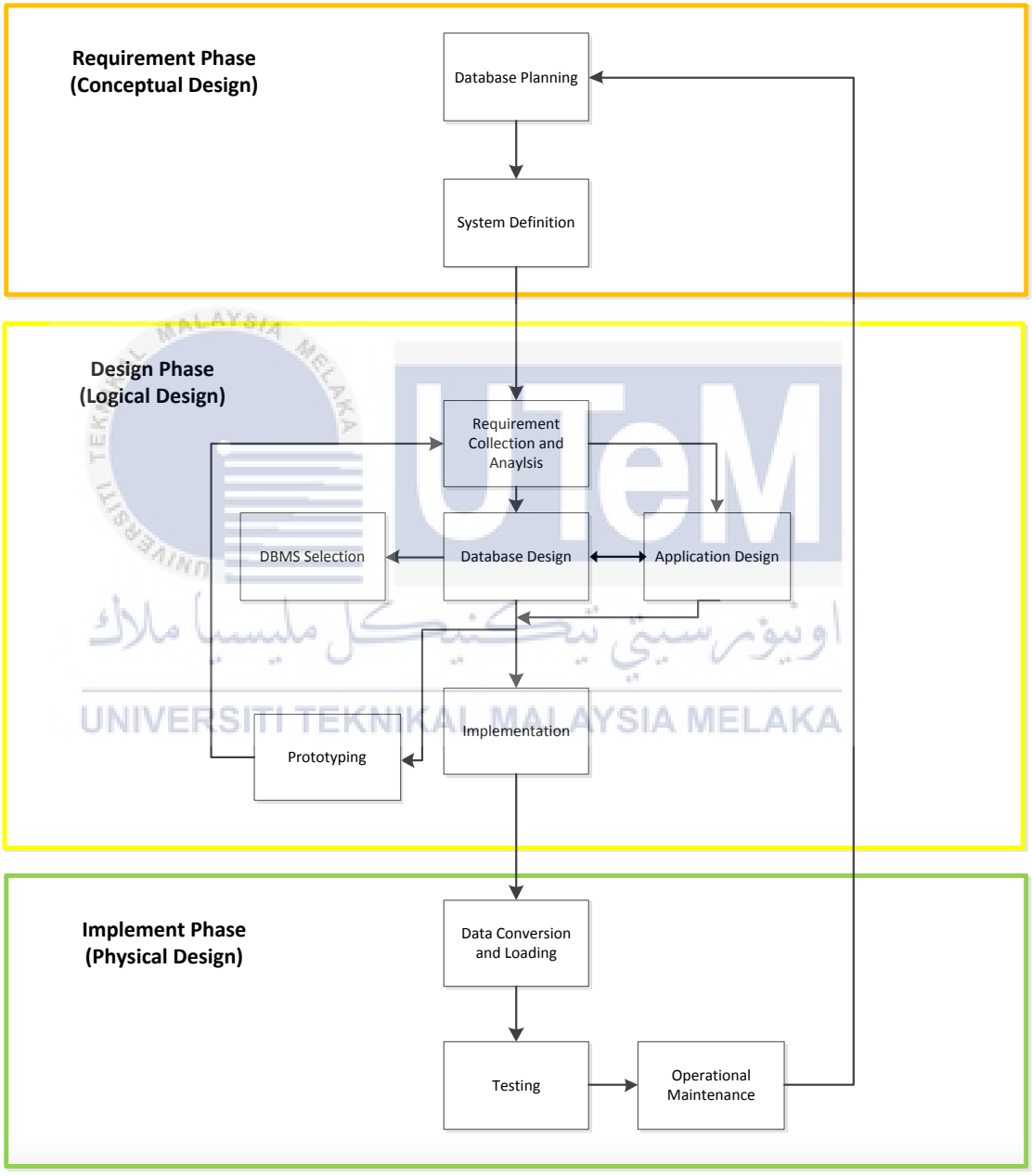


Figure 1: An Overview of Database Life Cycle (DBLC)

REQUIREMENT PHASE (Conceptual Design)

There are two ways to understand Conceptual Design by using Entity Relationship Diagram (ERD) and Business Rules .

Database Planning :

At this phase, database planning must be done by deciding platform of the database which is Oracle.

Deciding operating system that wants to implement the database.

System Definition :

System Definition at this phase means that in finding the scope of the project , deciding Operating System and database system connecting each of tables for manage data.

DESIGN PHASE (Logical Design

Another phase is design phase which is the logical design of the system. The Data Normalization is to validate the creation of ERD and the Data Dictionary is for the validation ERD .

There is six (6) process in this phase which are Requirements Collection and Analysis, Database Design,Application Design, DBMS Selection, Prototyping & Implementation. The Requirements Collection And Analysis is collecting important data ,making research from the interview (JRM) , references and websites . The Database Design is designing the data flow between database by using Waterfall Model and create a conceptual design , DBMS software selection,create logical and physical design. The DBMS Selection using Windows 7 as the platform operating system and using Oracle database. For the prototyping use PHP languages for interfaces connection between the database and operating system .Implementation process is installing the DBMS and create or designing the database.

IMPLEMENT PHASE (Physical Design)

The implement phase is for a physical designing system which is generated Data Definition Language (DDL) Schema. In this phase also doing the data conversion and loading , testing, operational and maintenance .

For the data conversion and loading is loading and converting the data. At testing process means that for testing , fine-tune the database and evaluating database also its applications. The operational process producing the required information flow and for the maintenance process is introduced changes and make enhancements.

a) Planning

The project planning starts in this phase. To begin with, the data is accumulated from the JRM administration staff and the bundle reservation administrations about the stream of the business process furthermore the normal framework. At that point, the degree targets and the objectives for the proposed framework are situated up.

The conditional timetable comprises of venture work arrangement and Gantt outline is produced.

b) Analysis

In the database initial study phase, study the situation of JRM. From that, identify the business process. The problem statement of the system can be defined throughout the observation. It will become the objective for the system. The scope can be extracted from the objective to develop the system.

c) Design

Database design is defined as the third phase, where a design for the database is formed. It can bolster the Digital Document Storage Management (e-JRM) operational and target, for example, Oracle is deciding for the database administration. The base prerequisite for the establishment should be affirmed first all together for the DBMS in the server to run easily.

Additionally, the Entity Relationship Diagram (ERD), and information lexicon is made where it will clarify the fundamental essential work process of the framework. All relationship between the tables characterize the capacity structures and the entrance ways will be known. In the applied

configuration stage, information displaying is utilized to make a theoretical database structure which speaks to this present reality objects. There are two sorts of database outline such commercial top-down configuration and base up configuration. At the top-down level, the

information set is recognized and information component is characterized. This procedure includes the ID of distinctive element sorts and the meaning of every element's traits. Other than that, in base-up configuration information component will be recognized which additionally assembled things and the gathering them in datasets.

d) Implementation

At this stage, the database administration that has been conFigured will be loaded and executed. The information will be load to make tables and characterized the relationship in the database system.

e) Testing

The next phase of DBLC is testing and evaluations. Once the data have been load into the database, the database is tested for performance, integrity, and concurrent access and security constraints. Other than that, testing and evaluate the system parallel with application programming is done.

After the evaluation stage, it can pass through the operational system. This phase involves all the users in Digital Document Storage Management (e-JRM) that will use this system. The testing and assessment stage happens in parallel with applications programming.

f) Evaluation and Maintenance

Support and advancement is the last stage in the approach and it likewise lifetime stage. The framework engineer will perform routine support to the Digital Document Storage Management (e-JRM) which occasional upkeep presuppose doing on the framework reinforcement, recuperation, upgrading or typical upkeep.

2.3 Project Schedule and Milestones

In this section will represent the milestones, work plan and dateline of project development. In table 2.1 shows the development plan:

Table 2.1: Project Schedule and Milestones

Milestones	Expected Documents	Dates
Problem Identification and analysis	- Chapter 1 and Chapter 2 (Introduction and Planning) - Flow chart of the proposed system. -State the requirement specification of the proposed system.	25th-March-2016
Conceptual design of the proposed system	- A complete ERD - Create the data dictionary - Design-storyboard	26th-March-2016
System Development	- Install the DBMS - Create the databases - Develop a full system based on the requirement specification.	1st -April-2016
Testing and Evaluation	- Testing the system - Evaluate all the errors - Repair all the errors	15th-May-2016
Implementation	- The final system approved and satisfies with all user requirements.	23rd –May-2016
Presentation	- The final system will be present to supervisor and coordinator.	30td –June-2016

Project Schedule

This project will be conducted for about 14 weeks. This is a list of stages based on Gantt chart have been created. Besides that, all the stages have been shaded by referring to the schedule and milestone. This table 2.2 shows the preparation of the work according to its phases set.

Table 2.2: Gantt Chart

WEEK / STAGES	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Problem Identification and analysis														
Conceptual design of the proposed system														
System Development														
Testing Evaluation														
Implementation														
Presentation														

2.4 Conclusion

This chapter examines about task approach to characterize the arranging of the project. Other than that, it will cover venture procedure, and undertake timetable and point of reference as an arrangement to arrange the idea of the task. The next chapter will be clarified the investigation of the present framework and the new framework.

CHAPTER III

ANALYSIS

3.1 Introduction

The analysis phase is describing the problem analysis and requirement for the project. The analysis is the description of the proposed project and it's characteristic. The analysis for the implementation of database documentation management system was carried out for two weeks. The main purpose of analysis report was to inform the technical developments of the project, analyze the techniques used to collect the data and interpreting the fact that obtained. There are various ways or techniques to find and collect the data such as interviews, surveys, observation, and questionnaire. Besides that, when to analyze the data pie charts or graphs can be used to interprets and summarize the analysis of findings. Thus, this report covers the analysis of findings. The deliverable at the end of this phase is a requirement document.

3.2 Problem Analysis

Problem Analysis is done to discover the reason for a constructive or adverse deviation when individuals, hardware, frameworks, or forms or not executing obviously. Issue Analysis focuses on the pertinent data and leads the best approach to foot cause. The procedure is utilized to accumulate and dissect only the data expected to discover and right the genuine reason for an issue, making it especially compelling in the today information rich environment. This advances quick and exact issue determination.

3.2.1 Current System Analysis

Currently, JRM Services Sdn.Bhd used a manual system to record the overtime salary and documents details. Figure 3.1 below shows the current system used by JRM Services Sdn.Bhd to upload and download documents. The data currently kept in the file and the original copy will be sent to the headquarters.

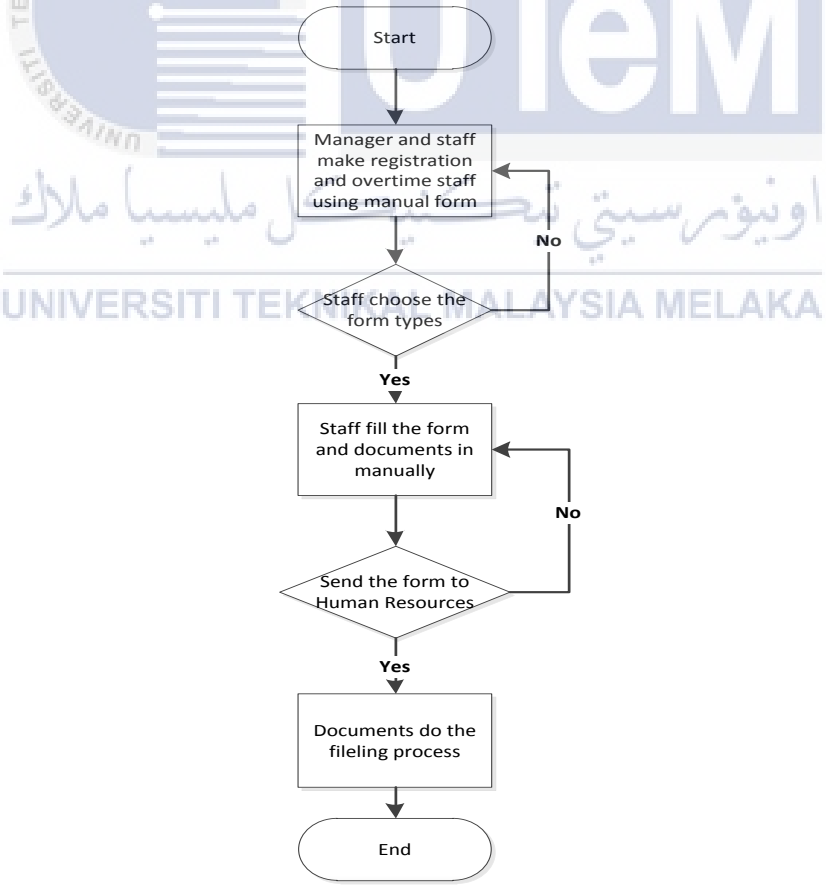


Figure 3.1: Flow Chart for Current System

Step 1: Manager and staff register overtime time details using manual form. The form will be saved in a file . This required large space to store files.

Step 2: A complete form will be sent to the Human Resources Departments (HR) **.** The incomplete form will be sent back from HR and make the amendment.

Step 3: Staff HR departments will check the forms, make the filling and keep it in cabinets in the store room. Manually documents file searching on cabinets by the staff make difficult and wasting time to find their documents.

3.3 The Proposed Improvements/Solutions

The purpose of improvement for the company is to give their works more easily and give the new environment of works based on the new concept ideas.

3.3.1 New System Concept for Level Admin(Manager).

Figure 3.2 shows the improvement of the new system for JRM Services Sdn.Bhd. towards the implementation of the new system.

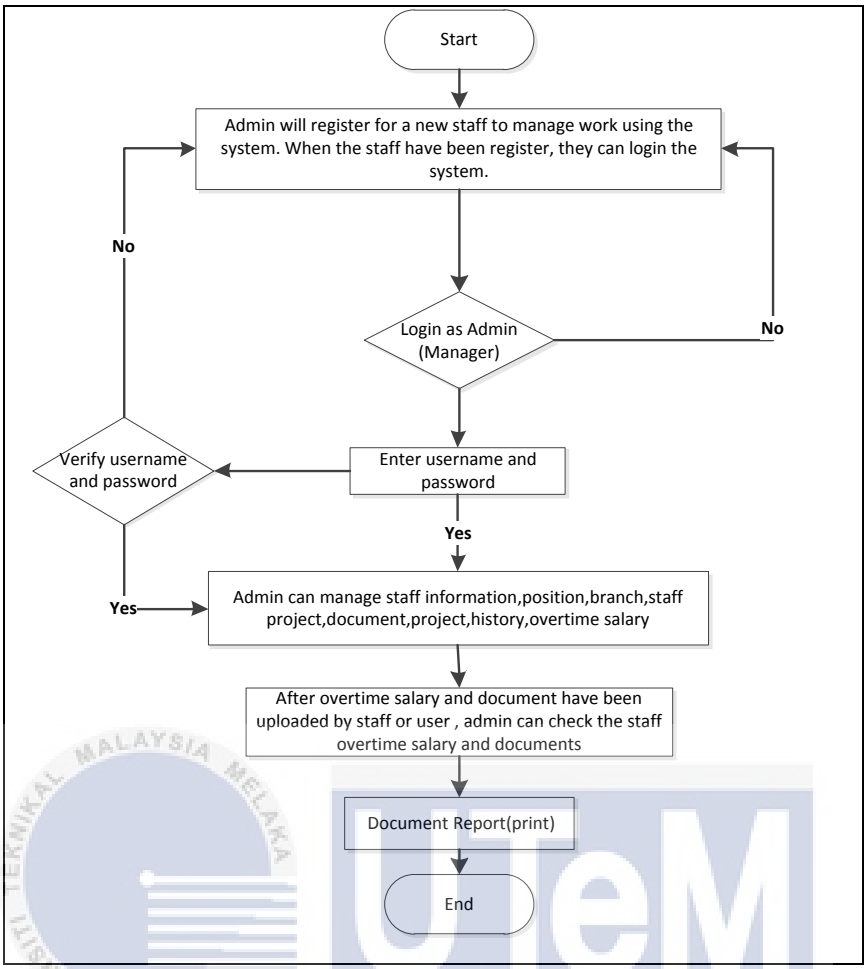


Figure 3.2: Flow Chart for New System Concept for Admin (Manager).

This Figure shows the managing documents storage and overtime salary. All the documents will be included and the information that has been made by the staff and manager will be view in the manager web page.

Step 1: Admin will make staff registration. The staff can be login into the system after registration.

Step 2: Manager manages the entire system JRM documents including the staff information, position, branch,staff project, projects,history and overtime salary. When the staff has uploaded overtime salary and documents , the manager can view and manage all documents.

Step 3: Manager gets the documents report of uploading documents via the system.

3.3.2 New System Concept for Level Staff.

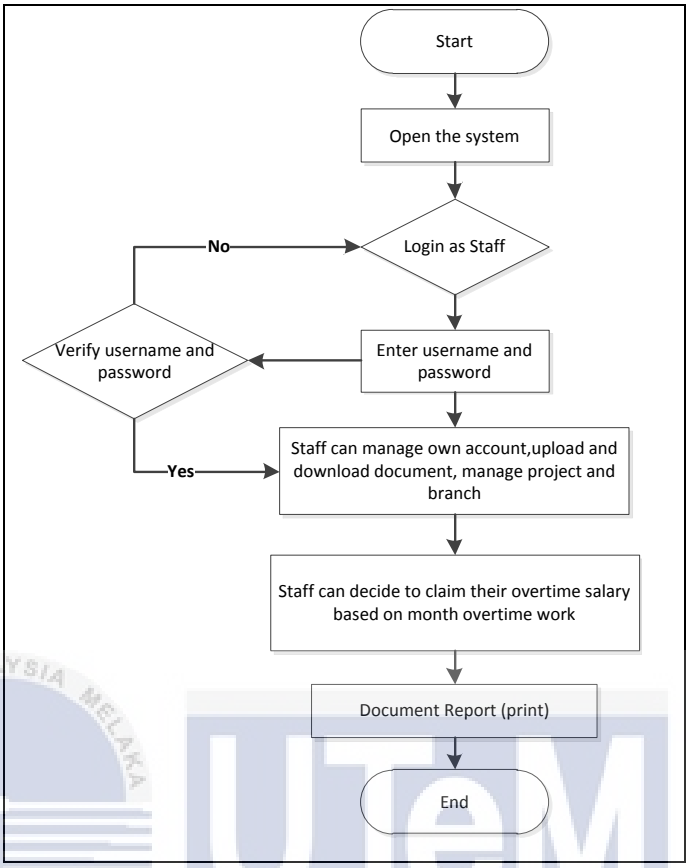


Figure 3.3: New System Concept for Level Staff

Figure 3.3 shows a new system that will use by a staff JRM Services Sdn.Bhd for using the system. All the overtime and documents types information will be stored in the database and can be view by the manager.

Step 1: Staff will make changes to their account staff , upload and download documents based on the type of documents that they want it through online. They need to open the JRM website and login into the system by using username and password from their manager . They can ask to obtain their new password from the manager which have registered by the manager for using the system.

Step 2: Staff can do their work after successfully login account. They can do a claim of their overtime salary in online at the end of the month.

Step 3: Staff can easily download documents report if they want it.

3.4 Requirement analysis of the to-be system

The requirements that need for this system during the system developments are:

3.4.1 Functional Requirement (Process Model)

Functional requirements are the decrypting the behavior of the system as it's related to the system functionality. It is what the system should be doing.

Table 3.1: Shows Functional Requirements

No .	Functional Requirements	Description
FR_1	User Registration	Before manager and staff can log into the system, they need to register first. For the staff, they will be given Identity Number and Password from the admin (manager). This is because manager(admin) will register for them before they can log in.
FR_2	Staff and Documents Management	The manager will create a new staff for new staff registration. The staff can create new information for branch, project, and documents that the want uploaded into the system.
FR_3	Upload and Download Type of Documents and Claim Overtime Salary	Staff who has been registered can make an upload and download type of documents. Staff can make their claims of overtime salary at the end of months. Besides that, manager can view the updated documents,branch, project,staff and overtime salary
FR_4	Generate Report	Admin and staff can generate documents report from date to date, by month and also by year.

3.4.1.1 Context Diagram

The Data Flow Diagram (DFD) is a graphic representation of the system flow and visualized the process of a system. DFD can help the developer to understand a system. Figure 3.4 will illustrate the context diagram of the DFD .

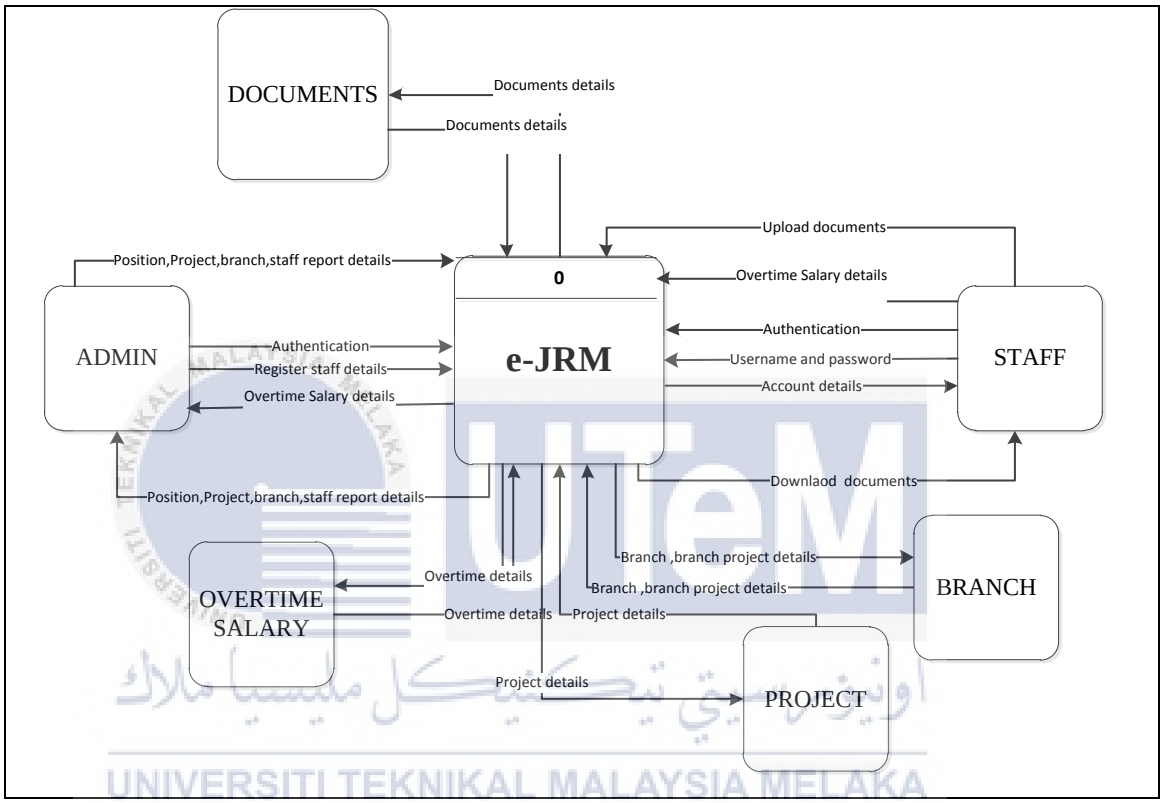


Figure 3.4: Context Diagram

3.4.1.2 Data Flow Diagram: Level 0

In this Figure 3.5 show the flows of the system process includes the process and the data store that has been provided and develop by the developer.

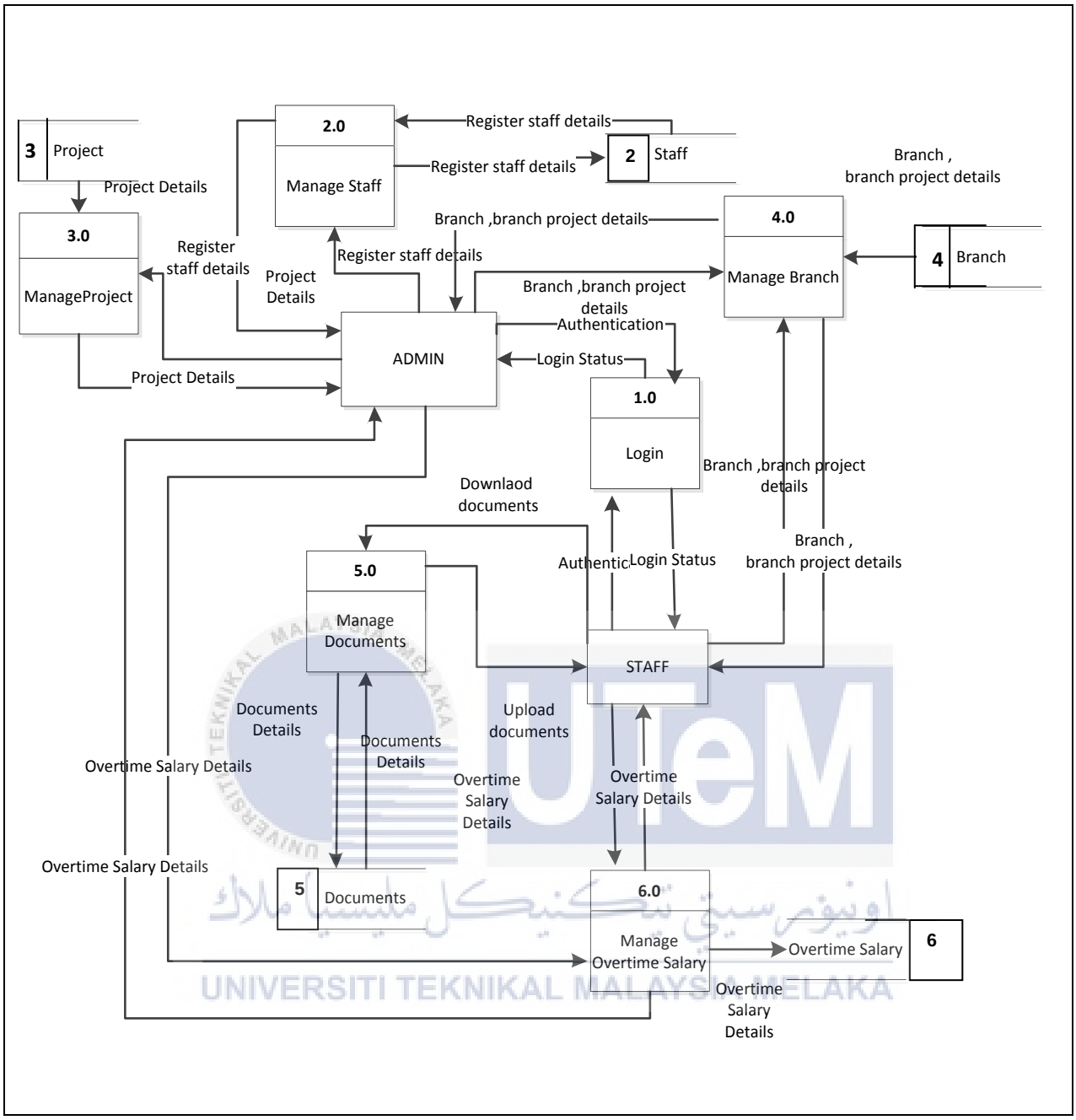


Figure 3.5: Data Flow Diagram Level 0

3.4.1.3 Level 1: Manage Staff

This Figure 3.6 shows the process for the major process which is Manage Staff. This Figure includes four (4) minor processes such as view staff, add new staff, update staff and delete staff. In this process will be handling by the admin the manager to give an information to the staff any new update.

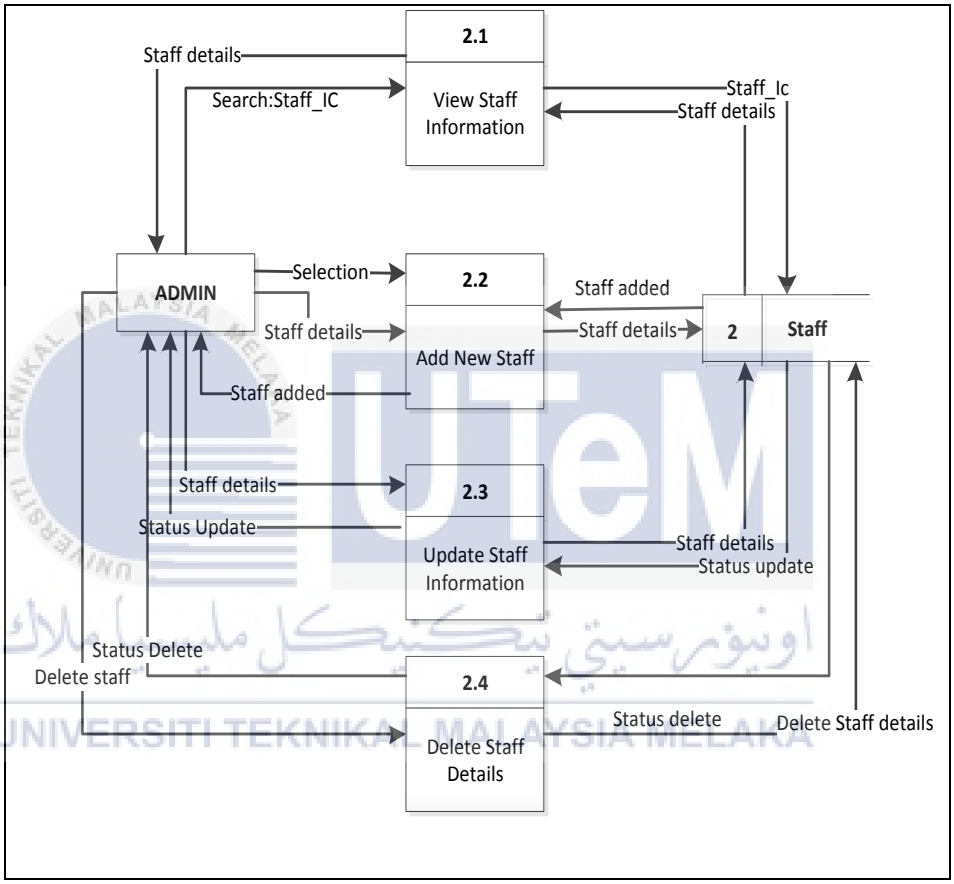


Figure 3.6: Level 1 of Manage Staff

3.4.1.3.2 Level 1: Manage Overtime Salary

This Figure 3.7 shows the progress from the major process which is Manage Overtime Salary. This Figure includes four (3) minor processes such as view overtime salary, check and update overtime salary and delete overtime salary.

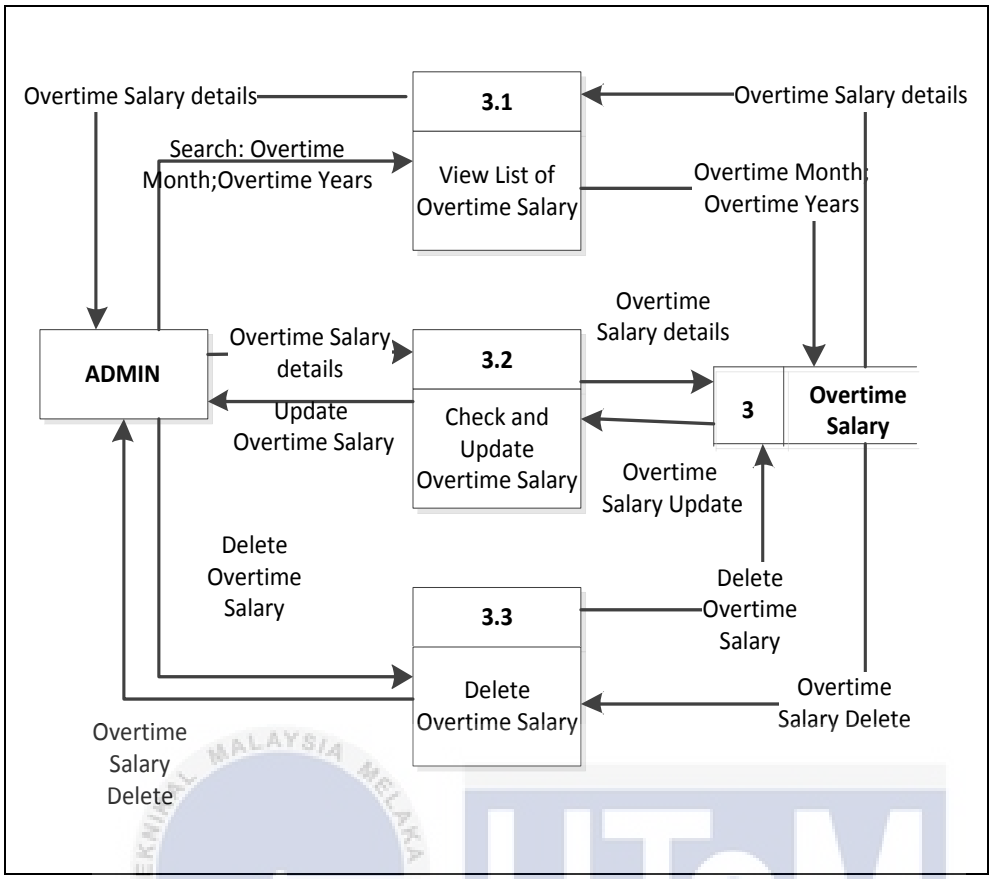


Figure 3.7: Shows Level 1 of Manage Overtime Salary

.

3.4.1.3.3 Level 1: Manage Type of Documents

This Figure 3.8 shows the progress from the major process which is Manage Type of Documents.

This Figure includes three(3) minor processes such as upload documents, download documents, and print documents. In this process,after staff has the select type of documents that they want , then the process upload file and download file happen. Then, staff can decide to print documents if they have chosen their wanted file.

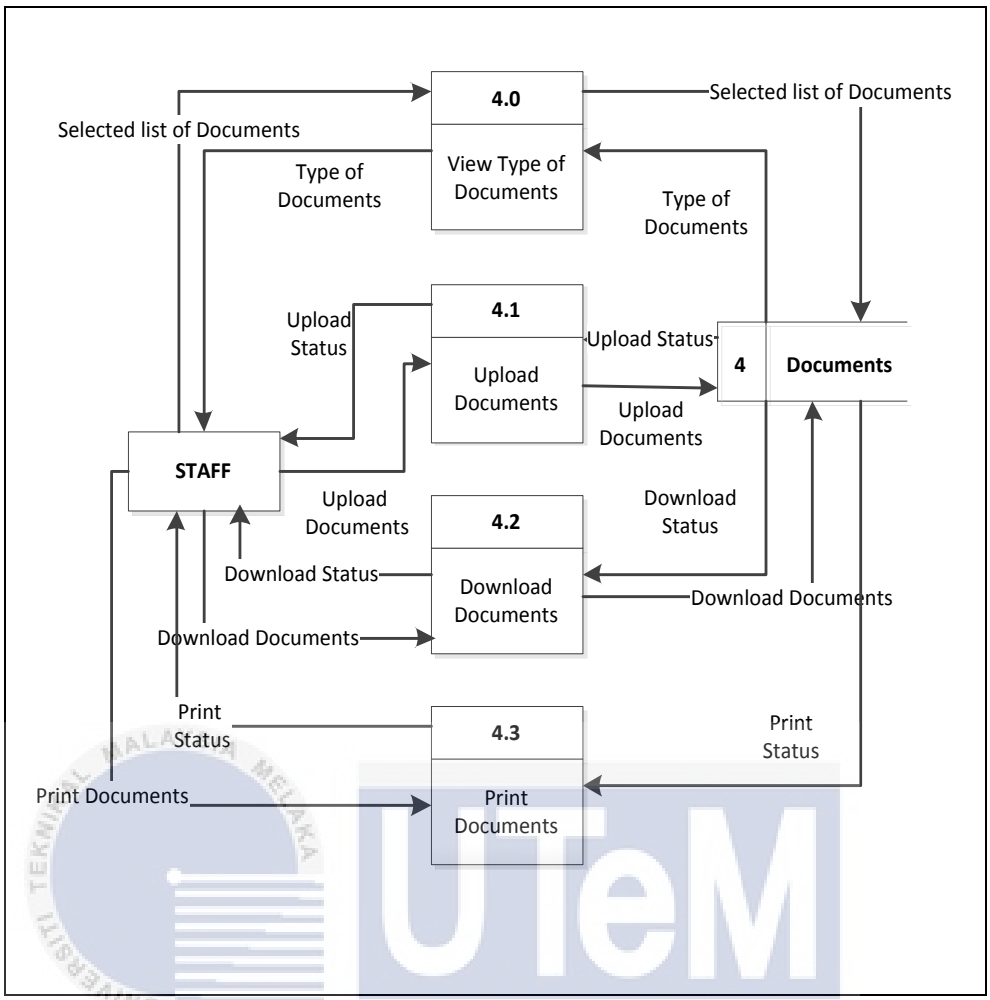


Figure 3.8: Shows Level 1 for Manage Type of Documents

3.4.1.3.4 Level 1: Generate Report

This Figure 3.9 shows the progress from the major process which is Generate Report. This report can be view by manager and staff .

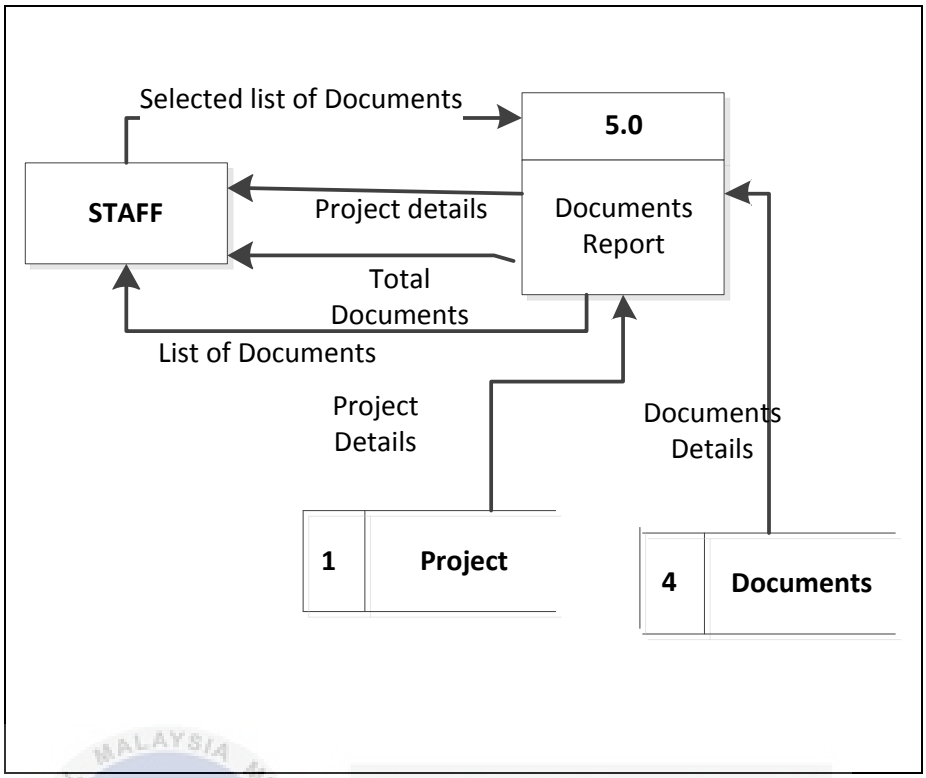


Figure 3.9: Generate Report.

3.4.2 Non-Functional Requirements

Non-Functional requirements are the elaborate a performance characteristic of the system. It's contained how the system will do so as in Table 3.2:

Table 3.2: Non-Functional Requirements

No .	Non-Functional Requirements	Description
NFR_1	Availability	The system available for service when requested by users. Example: An authorize users can access the system when needs.
NFR_2	Security	One or more requirements about protection of your system and its data. Example: Authenticate users during a login session.

NFR_3	Accuracy	Requirements about the accuracy and precision of the data. Example: Must fill the entire required field before users can send the forms.

3.4.3 Other Requirement

3.4.3.1 Software Requirement

The system need to have at least software requirements that have been list as below in table 3.4:

Table 3.3: Shows The Software Requirement.

Type Of Software	Description
Adobe Dreamweaver CS3	Adobe Dreamweaver is a sophisticated authoring package that enables to build complex interactive Web sites using HTML, JavaScript, and server-side programs languages.
Windows 7- Microsoft System	This working framework as a stage for DBMS and framework improvement introduced on it. Window 7 propelled from Microsoft and has preferable apparatuses over Windows XP.
Microsoft Office Word 2010	Microsoft Office Word 2010 for writing a report and design a diagram.
Google Chrome	It is utilized as a web program to dispatch the site. It is prescribed for the client to utilizing a most recent variant of it.
Microsoft Visio Studio	Microsoft Visio Studio is accustomed to drawing the Entity Relationship Diagram which is for configuration the databases.
Xampp Server 2.5	Xampp Server is a form of mini-server that can run on almost any Windows operating

3.4.3.2 Hardware Requirement

These equipment prerequisites are extremely least necessity that needs to a customer and server needs. These days, maybe the equipment for customer and server is vastly improved than these base prerequisites. The equipment prerequisites are demonstrated in Table 3.4:

Table 3.4: Shows The Hardware Requirement.

Type Of Hardware	Description	Server	Client
Hard Disk Storage	The hard disk is fundamental main storage on a PC where the entire product introduced on it.	Minimum 100GB free disk space	Minimum 300 MB free disk space.
Random Access Memory(RAM)	Memory is characterized as Random Access Memory (RAM) gives Space to the PC to peruse and compose information to be gotten to by the CPU (central processing unit) or processor.	Minimum requirement of memory required is 2GB, though 3GB is recommend ed.	Minimum 512 MB of memory, though 1 GB is recommended.
System Processor	The processor is the electronic part which goes about as "brain" for of a PC. The higher the preparing space is vastly improved.	Minimum 2.25 GHz speed of CPU processor.	Minimum 1.3 GHz speed of CPU processor.

3.4.3.3 Other Requirement

The system needs to have at least to support the process of project development that has are listed as below in Table 3.5:

Table 3.5: Shows The Other Requirement.

Type of Requirement	Description
UTP Cable	Get a connection to the internet to find information.
Laptop	As the main platform to develop a system, to install all the software required and run the system.

3.5 Conclusion

This chapter discusses the analysis of system based on the current and the improvement of the system to be developed. It also recovers about the requirement such as software, hardware, and others to define what kind of things is needed during system development. On the next chapter will explain about entity relationship diagram and the system architecture.

CHAPTER IV

DESIGN

4.1 Introduction

This chapter will discuss the details of system design. The conceptual model that defines the structure, behavior, and more views of a system. Database Design is the procedure of delivering a point by point information model of a database. This conceptual model contains all the intelligent required and physical configuration decisions and physical storage parameters expected to produce an outline in a Data Definition Language (DDL), which can be utilized to make a database.

The conceptual design which is Entity Relationship Diagram (ERD) of the project is illustrated to make the idea of the system-to-be become more understandable. Besides that, the conditions of the system are stated clearly with the help of the Business Rule. The Data Dictionary of the Entity Relationship Diagram (ERD) is provided in this report. Data Dictionary contains all the attributes in entities with its format and type and the primary key of the entity also stated in the Data Dictionary too. Data Definition Language (DDL) is produced based on the conceptual and logical design of the database.

4.2 System Architecture Design

This outline demonstrates the framework structural engineering. It accommodates enhanced versatility, sensibility, and great asset use.

Figure 4.0 shows common “tiers” include:

- Web server tier gives HTTP convention bolster or handles web demands.
- Application server tier: gives backing to web administrations, business rationale, and so on.
- Database tier provides data storage and retrieval support.

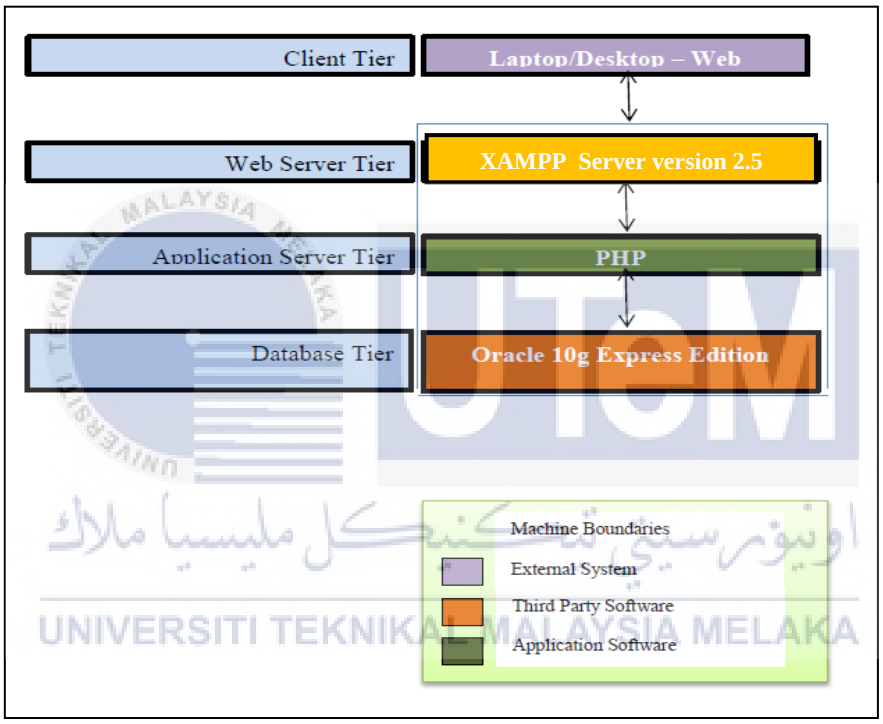


Figure 4.0: Shows System Architecture Design

4.3 Database Design

Database designs are an architecture that focuses on the configuration of the database structure that will be to store and deal with clients' information. Database architect needed to recognize decisively the database's normal utilized.

An all-around composed database encourages information administration and creates exact and important data.The design of the database use the waterfall model and some progress is been done by created conceptual design, database management system DBMS software selection, created logical also physical design.

4.3.1 Conceptual Design

The applied database outline will show the principle information characters, qualities, connections, and a requirement of a space venture. The configuration of a database will be autonomy of database programming and physical points of interest. It knowingly made out of a graphical representation and in addition literary means of principle information components, connections and constraints. The concept of the system includes the planning of database , the definition of the system and Entity Relationship Design (ERD) also include the Business Rule. This has been elaborate at the requirement phase on chapter 2.

4.3.1.1 Entity Relationship Diagram (ERD)

The Entity Relationship Diagram (ERD) is a model that demonstrates the logical relationship and collaboration among framework element. This chart show gives a general perspective of the framework and a diagram in making the physical information structure. Figure 4.1 show the ERD for this system.

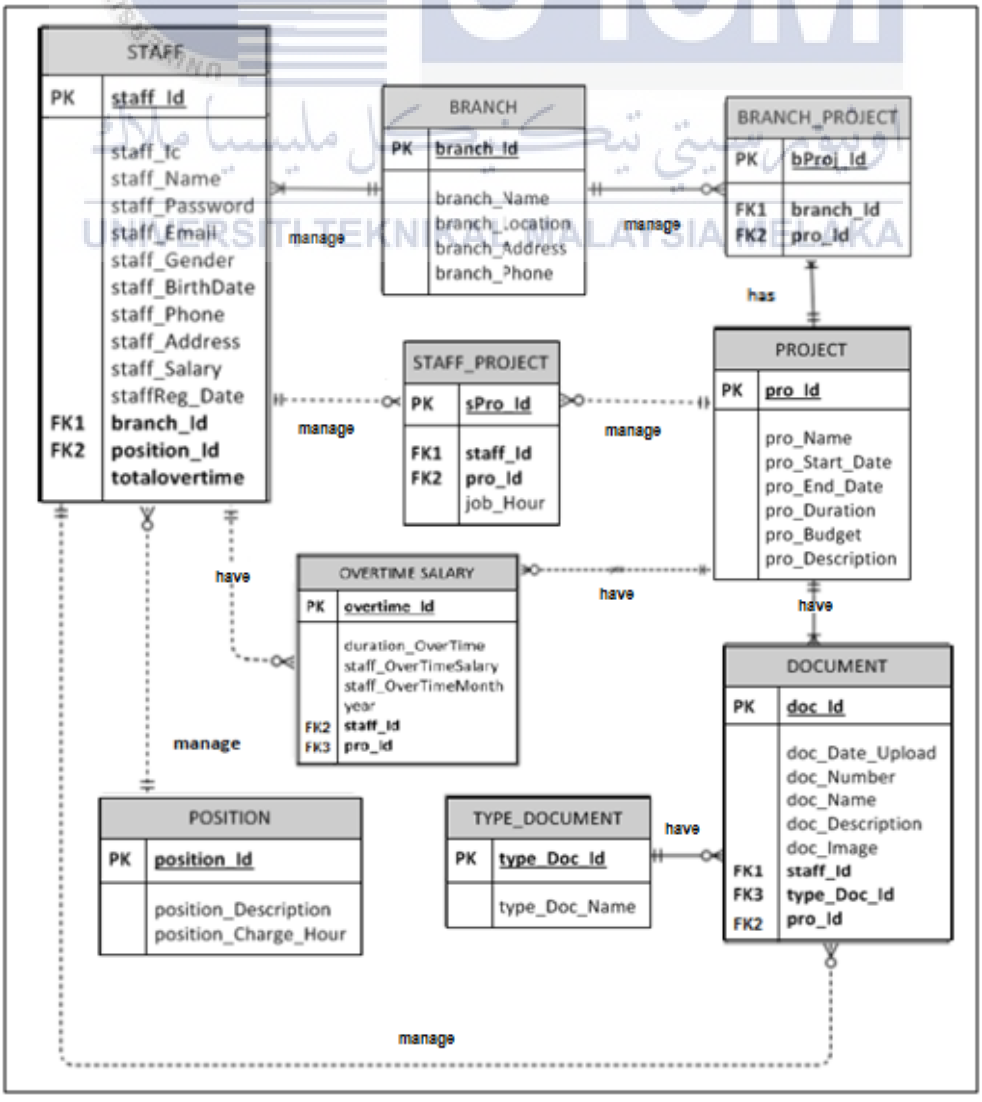


Figure 4.1: Entity Relationship Diagram (ERD)

4.3.1.2 Business Rules

i. One staff have one or many overtime salaries.

- Staff can have one or many overtime salaries also , they may not have apply overtime salary.

Many overtime salaries can apply on one project

- This mean that a project may have much overtime claims that claimed by staff

ii. One position manages by one or many staff.

- The manager can be a staff and also can be an admin too that managed the full system.

One staff can manage by one position only.

- Staff such as a programmer have only one position that has registered by the manager.

iii Staff will manage one or many staff project.

- One staff manage one project or can be manage many projects at one time.

One staff project will be managed on one project in certain time job hour.

- One staff managed one project and may be managed many projects on certain job hour

iv. One branch can manage by many staff

- A branch managed by many staff because to reduce work pressure.

One staff can manage one branch on one time.

- Only one staff can handle job on one branch

v. One branch manages by one or many branch project.

- A branch managed by one branch project and can be managed by many branch project.

One branch project managed by one branch in one time.

- A branch project included project which managed in one time by one branch

vi. One project can have many branch project.

- A project can have many branch project.

One branch project only has one project.

- A branch project only has one project.

vii. One project has many documents.

- A project has many documents to manage by staff.

One document has one project at one time.

- A document that created is hold one project at one time.

viii. One type document has many documents.

- A type of document has many documents created.

One document has one type of document.

- One document has one type of document that differentiates the document.

ix. One staff can manage one or many documents.

- Staff managed one document and may manage many documents.

One document is managed by one staff at one time.

- A document managed by staff at one time

4.3.2 Logical Design

Logical database outline is to plan an undertaking – wide database in light of particular information display yet autonomous of physical – level points of interest. It obliged all article in the reasonable database configuration to be mapped to the particular develops utilized by the chosen database model. Logical database design for the system was shown in Table 4.1 until Table 4.10.

Table 4.1: Data Dictionary for Staff Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**staff_Id**	Varchar2	4	Yes		Not Null
2.	staff _Ic	Varchar2	25			
3.	staff _Name	Varchar2	50			
4.	staff _Password	Varchar2	20			
5.	staff _Email	Varchar2	20			
6.	staff_Gender	Varchar2	10			
7.	staff_BirthDate	Date				
8.	staff_Phone	Number				
9.	staff_Address	Varchar2	100			
10.	staffReg_Date	Varchar2	50			
11.	totalOvertime	Varchar2	20			
12.	**branch_Id**	Varchar2	4		Yes	Not Null
13.	**position_Id**	Varchar2	4		Yes	Not Null

i. Schema for Staff table:

Staff (**staff_Id**, staff_Ic, staff_Name, staff_Password, staff_Email, staff_Gender,staff_BirthDate,staff_Phone,staff_Address, staffReg_Date,totalOvertime,branch_Id,position_Id)
Primary key staff_Id
Foreign key branch_Id references Branch(branch_Id)
Foreign key position_Id referencs Position(position_Id)

Table 4.2: Data Dictionary for Position Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**position_Id**	Varchar2	10	Yes		Not Null
2.	position _Description	Varchar2	50			
3.	position _Charge_Hour	Varchar2	50			

i. Schema for position table:

Position (**position_Id**, position_Description,position_Charge_Hour) Primary key position_Id

Table 4.3: Data Dictionary for Branch Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**branch_Id**	Varchar2	10	Yes		Not Null
2.	branch_Name	Varchar2	20			
3.	branch_Location	Varchar2	30			
4.	branch_Address	Varchar2	100			
5.	branch_Phone	Varchar2	10			

i. Schema for branch table:

Branch(**branch_Id**,branch_Name,branch_Location,branch_Address,branch_Phone) Primary key branch_Id

Table 4.4: Data Dictionary for Type_Document

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**type_Doc_Id**	Varchar2	10	Yes		Not Null
2.	type_Doc_Name	Varchar2	100			

i. Schema for type_document table :

Type_Document(**type_Doc_Id**, type_Doc_Name) Primary key type_Doc_Id

Table 4.5: Data Dictionary for Project Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**pro_Id**	Varchar2	10	Yes		Not Null
2.	pro_Name	Varchar2	100			
3.	pro_Start_Date	Varchar2	20			
4.	pro_End_Date	Varchar2	20			
5.	pro_Duration	Number				
6.	pro_Budget	Varchar2	20			
7.	pro_Description	Varchar2	200			

i. Schema for project table:

Project(**pro_Id**, pro_Name,pro_Start_Date,pro_End_Date,pro_Duration,pro_Budget,pro_Description)
Primary key pro_Id

Table 4.6: Data Dictionary for Staff Project Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**sPro_Id**	Varchar2	10	Yes		Not Null
2.	job_Hour	Varchar2	20			
3.	staff_Id	Varchar2	10			
4.	pro_Id	Varchar2	10			

i. Schema for staff project table :

Staff_Project(**sPro_Id,** job_Hour,staff_Id,pro_Id)
Primary key sPro_Id
Foreign key staff_Id references Staff(staff _Id)
Foreign key pro_Id reference Project(pro _Id)

Table 4.7: Data Dictionary for Branch Project Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**bPro_Id**	Varchar2	10	Yes		Not Null
2.	branch_Id	Varchar2	10			
3.	pro_Id	Varchar2	10			

i. Schema for branch project table:

Branch_Project(**bPro_Id,** branch_Id,pro_Id)
Primary key bPro_Id
Foreign key branch_Id references Branch(branch _Id)
Foreign key pro_Id reference Project(pro _Id)

Table 4.8: Data Dictionary for Overtime Salary

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**overtime_Id**	Varchar2	10	Yes		Not Null
2.	overtime_Duration	Varchar2	20			
3.	staff_OverTimeSalary	Varchar2	20			
4.	staff_OverTimeMonth	Varchar2	15			
5.	year	Varchar2	10			
6.	staff_Id	Varchar2	10			
7.	pro_Id	Varchar2	100			

i. Schema for overtime salary table :

Overtime_Salary(**overtime_Id,** overtime_Duration, staff_OverTimeSalary, staff_OverTimeMonth,year,staff_Id,pro_Id)
Primary key overtime_Id
Foreign key staff_Id references Staff(staff _Id)
Foreign key pro_Id referencs Project(pro _Id)

Table 4.9: Data Dictionary for Document Table

No.	Name	Data Type	Length	Primary Key	Foreign Key	Null / Not Null
1.	**doc_Id**	Varchar2	10	Yes		Not Null
2.	doc_Date_Upload	Varchar2	15			
3.	doc_Number	Varchar2	100			
4.	doc_Description	Varchar2	300			
5.	doc_Image	Blob				
6.	staff_Id	Varchar2	30			
7.	type_Doc_Id	Varchar2	10			
8.	pro_Id	Varchar2	10			

i. Schema for document table:

Document(**doc_Id,**doc_Date_Upload, doc_Number,doc_Description, doc_Image,staff_Id, type_Doc_Id ,pro_Id)
Primary key doc_Id
Foreign key staff_Id references Staff(staff _Id)
Foreign key staff_Id references Type_Document(type_Doc _Id)
Foreign key pro_Id referencs Project(pro _Id)

4.3.3 Physical Design

The physical database design is a process to identify the data storage organization and data access characteristics of a database in order to ensure its integrity, security, and performance.

4.3.3.1 Data Definition Language (DDL)

DDL is a language that permits a client to depict and name the substances, traits, and relationship needed for the application, together with any related uprightness and security requirement. It used to characterize a pattern or to alter existing outline.

a. Create Tables

```
create table staff
(staff_Id varchar2(10) PRIMARY KEY,
staff_Ic varchar2(20),staff_Name varchar2(60),
staff_Password varchar2(10),staff_Email
varchar2(30),staff_Gender varchar2(10),
staff_Age NUMBER,
staff_Phone NUMBER,
staff_Address varchar2(30),
staff_Salary NUMBER,
staff_OverTimeSalary NUMBER, staffReg_Date
DATE,branch_Id REFERENCES branch(branch_Id),
position_Id REFERENCES position(position_Id));
```

Figure 4.2: Create table "staff"

```
create table branch
(branch_Id varchar2(10) PRIMARY KEY,
branch_Name varchar2(20),
branch_Location varchar2(30),
branch_Address varchar2(30),
branch_Phone NUMBER
);
```

Figure 4.3: Create table "Branch"

```
create table type_document
(type_Doc_Id varchar2(10) PRIMARY KEY,
type_Doc_Name varchar2(100)
);
```


Figure 4.4: Create table "Type_Document"

```
create table Project
(pro_Id varchar2(10) PRIMARY KEY,
pro_Name varchar2(100),
pro_Start_Date Date,
pro_End_Date Date,
pro_Duration NUMBER,
pro_Budget  NUMBER(9,2),
pro_Description varchar2(200) );
```

Figure 4.5: Create table "Project"

```
create table staff_project
(sPro_Id varchar2(10) PRIMARY KEY,
job_Hour NUMBER(9,2),
staff_Id REFERENCES STAFF(staff_Id),
pro_Id REFERENCES PROJECT(pro_Id));
```

Figure 4.6: Create table "staff_project"

```
create table branch_project
(bProj_Id varchar2(10) PRIMARY KEY,
branch_Id REFERENCES branch(branch_Id),
pro_Id REFERENCES PROJECT(pro_Id)
);
```

Figure 4.7: Create table "Branch Project"

```
create table overtime_salary
(overtime_Id varchar2(10) PRIMARY KEY,
overtime_Duration NUMBER(9,2),
staff_OverTimeSalary NUMBER,
staff_OverTimeMonth varchar2(15),
year  VARCHAR2(10),
staff_Id REFERENCES STAFF(staff_Id),
pro_Id REFERENCES PROJECT (pro_Id)
);
```

Figure 4.8: Create table "Overtime Salary"


```
Create Table Document
(Doc_Id Varchar2(10) Primary Key,
Doc_Date_Upload Vachar2(15),
Doc_Number Varchar2(100),
Doc_Name Varchar2(100),
Doc_Description Varchar2(300),
Doc_Image Blob,
Staff_Ic Varchar2(30),
Type_Doc_Id References
Type_Document(Type_Doc_Id),
Pro_Id References Project(Pro_Id)
);
```

Figure 4.9: Create table "Document"

```
create table branch_audit
(branch_Id varchar2(10)primary key,
branch_Name varchar2(20),
branch_Location varchar2(30),
branch_Address varchar2(100),
branch_Phone varchar2 (10)
);
```

Figure 4.10: Create table "Branch_Audit"

b. Create Trigger and Procedure

Please refer trigger at Appendix A and stored procedure at Appendix B for Figure 4.11 until Figure 4.58.

4.4 Graphical User Interface (GUI) Design

This interface in Figure 4.59 shows the first page that will be implementing the system for all users. Besides that, customers can log in through this page also.

Figure 4.59: ora1Index.html (Home)Page

4.4.1 Admin Page

4.4.1.2 Admin Login Page with True Validation

Figure 4.60 shows the Admin and staff login at the same login page. Then, the interface in Figure 4.61 shows true validation since the manager click 'login' button to show that identity number and password is valid.

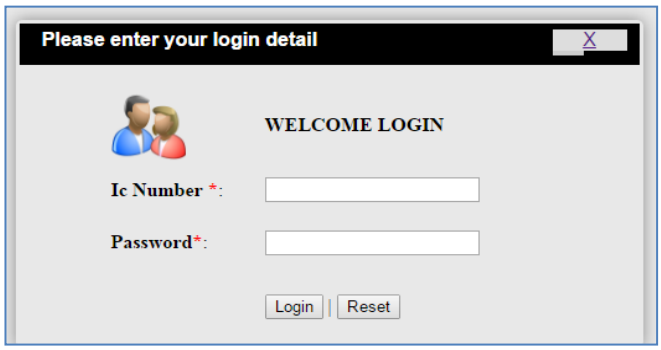


Figure 4.60 : Login Page

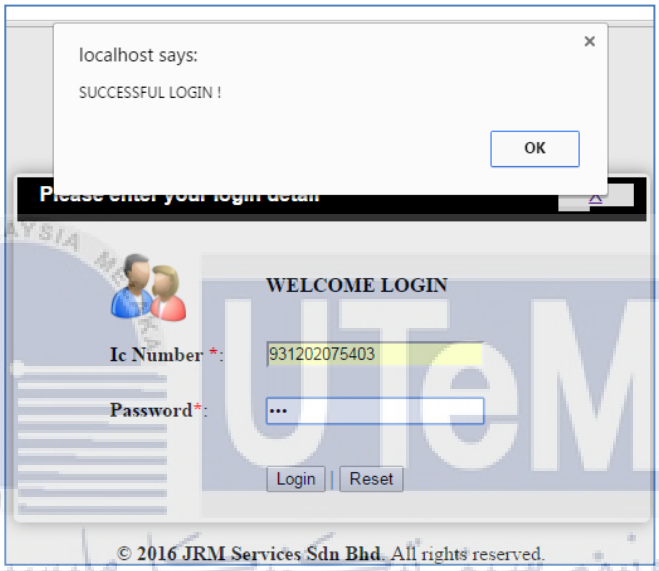


Figure 4.61: Manager/Admin Login with True Validation

4.4.1.3 Admin Login Page with Wrong Validation

This interface in Figure 4.62 shows wrong validation since the manager click 'login' button to show that identity card number and password is not valid.

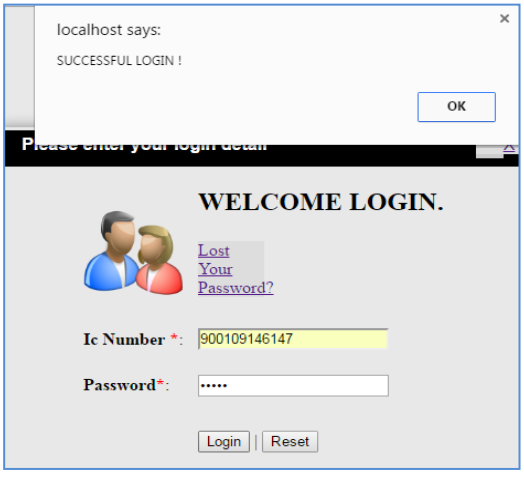


Figure 4.62: Manager/Admin Login with Wrong Validation

4.4.1.4 Admin Home Page

This interface in Figure 4.63 shows admin home page or "oraAdmin.php" after login session is valid.

Figure 4.63: Admin Home Page.

4.4.1.5 List of Staff

A manager is able to view list of registered staff from the new system by clicking the provided "View" button on the screen. Please refer to Appendix C for Figure 4.64 and Figure 4.65.

4.4.1.6 List of Overtime Claim by Staff

This is the form of overtime claim by staff that has submitted or claim by staff by using this system. Figure 4.66 show the details and calculated overtime claimed by staff. Admin can delete the wrong overtime claim. Staff must tell to the manager that they have to make mistakes (human behavior). If they do not do so, they cannot claim their overtime.

Figure 4.66 : The details of overtime claim

The Figure 4.67 , show the search on finding the project that has progressed at the branch location. The user can find the project by choosing the list out branch location.

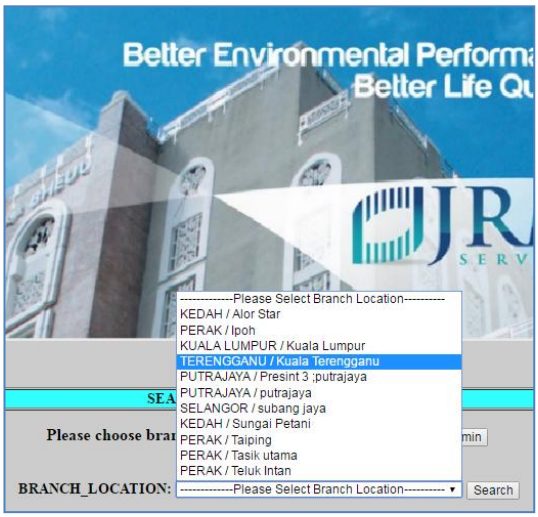


Figure 4.67: The search of project based on branch location

After that, user click on button 'search'. Then, the output comes out which tell the project name that located at the branch location. This shown in Figure 4.68.

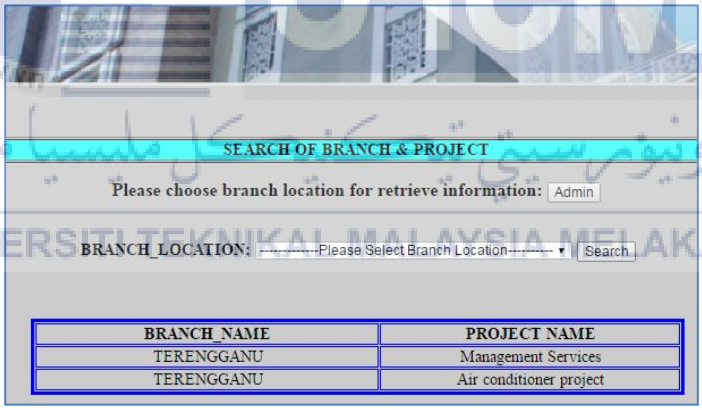


Figure 4.68: The project name based on branch location

4.4.1.7 Upload Document Image

The Figure 4.69 above show the upload form document . The user must fill in the details and uploaded the document image and submit by clicking the button "Add New Document". Please refer at Appendix C for Figure 4.69 until Figure 4.72.

Searching method is to check for duplicate document number and document uploaded by the user. Then, the user can delete the document that duplicated.

Figure 4.73 shown the search document and delete if happen duplication uploaded document. Figure 4.74 and Figure 4.75 shown the details of documents after click view the

document and trace (search) project supervisor / staff who handle the project.Please refer at Appendix C for view Figure 4.73 until Figure 4.75.

4.4.1.8 Security Account .

```
</script>
<script type="text/javascript">
idleMax = 1;// Logout after 1 minutes of IDLE
idleTime = 0;
$(document).ready(function () {
    var idleInterval = setInterval("timerIncrement()", 60000);
    $(this).mousemove(function (e) {idleTime = 0;});
    $(this).keypress(function (e) {idleTime = 0;});
})
function timerIncrement() {
    idleTime = idleTime + 1;
    if (idleTime > idleMax) {
        alert('You have been logged out.You will now be redirected to home page.');
                document.location.href = "ora1index.html";
    }
}
</script>
```

Figure 4.76: Coding Security Logout automatic after 1 minutes login

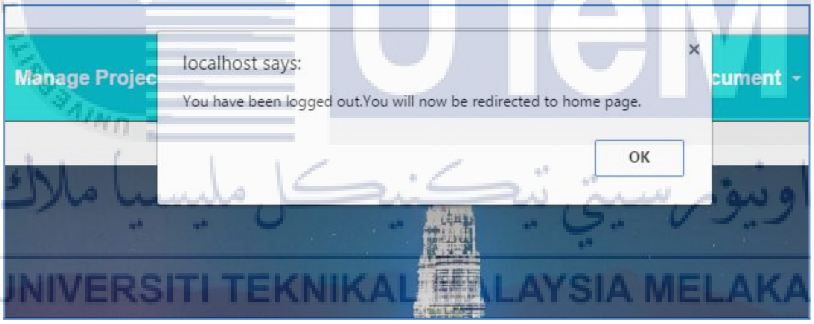


Figure 4.77: Logout automatic if account user not managed by the user

Admin can register staff during business hour which is from 8:00 a.m. until 6:00p.m. are shown on Figure 4.78. Please refer Appendix C for Figure 4.78 until Figure 479.

4.4.1.9 Overtime Claim

Figure 4.80 show the process staff for claim overtime salary . Staff must insert their overtime details for the claim. If they did not follow,they cannot get their claim overtime based on month and year claim.

For future; this part may include the biometric sensor thumb screen to make sure the staff come and dismiss their work from the office. Please refer Appendix C for Figure 4.80 until Figure 4.84.

Figure 4.83 and Figure 4.84 shows the overtime checking by admin which is the manager/admin can check the overtime claim by staff .Manager searches the chosen name for view the overtime claim details. Please refer Appendix C for Figure 4.85 and Figure 4.86.

4.4.2.0 Document Report

This Figure 4.87 and Figure 4.88 shows the report of document uploaded and the report total document uploaded. Figure 4.89 and Figure 4.90 show bar graph report and view print bar graph report. Please refer at Appendix C for Figure 4.87 until Figure 4.90.

4.4.2.1 Reset Password

If staff forgot their password, they must inform the manager for reset their password. The manager will reset their new password and give to them. After staff got their new password, they begin to login back for change their password. Figure 4.91 show the pop-up came out when user click at "Lost Your Password?" tells staff to contact admin for resetting their password.

Figure 4.92 shows the checking of identity card for making sure that the user that want to reset password is approved as staff JRM. Please refer at Appendix C for Figure 4.91, Figure 4.92, Figure 4.93 and Figure 4.94

4.4.2.2 Coordinate Company

Figure 4.95 show the maps and location of JRM Services Sdn.Bhd. Company based on the satellite view. There are location address, contact number and operation day of the company. The user can zoom into the map location and see the real strategic location of JRM Services Sdn. Bhd. Company. Please refer at Appendix C for Figure 4.95.

4.5 Conclusion

This chapter explains about the system architecture design and also database design. In database design phase have been explained about the conceptual design, logical design and physical design. On the next chapter, will be describing the system implementation.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

This chapter discusses the usage of the undertaking those two (2) sections which are the framework advancement and database execution. The system development environment will be explained on how the installation step, assign admin login and starting the database service. Besides that, it also consists of the database creation and database object.

For the database, implementation includes the DDL or DCL statements in the chosen DBMS which is Oracle 10g Express Edition. In this database includes main processes such as stored procedures and trigger by using this programming language.

5.2 Software Development Environment Setup

The initial setup and the component to develop e-JRM project will be explained at implementation section. The architecture used by the project is the three (3) – tier of system architecture.

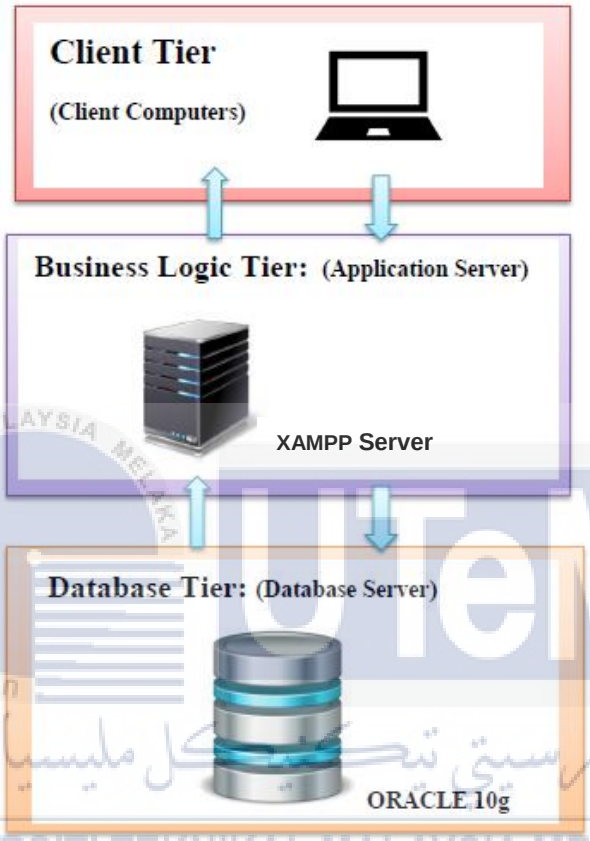


Figure 5.1: Three (3) – Tier System Architecture.

5.2.1 Software Development Setup

In order to develop the e-JRM system, a developer need to have a personal computer or notebook that has an authoring tool to design and compile a PHP programming language such as Adobe Dreamweaver CS3. Besides that, the developer needs to install XAMPPSERVER 2.5 as a server. The project is using an Oracle 10g Express Edition as a database of a project. Project data will be stored at 'http://127.0.0.1:8080/apex' . The link will be created automatically after completed installation of the Oracle Database Express Edition.

For the server parts, the client keeps running as “HTTP:\\localhost” in the web browser, the web browser will display the main page of the XAMPPSERVER. An organizer name as "Jim" was made in the registry "C:\xampp\htdocs at the point when the client keeps typing so as to run as a localhost "HTTP:\\localhost" in the web program, the program will show the fundamental page of the XAMPP Server.

5.2.1.1 Software Development Setup - Server

XamppServer in Windows Platform (Installation of XAMPP Server Guide) used to launch the web that has been designed in PHP.

STEP 1:

Download the software at https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/1.7.7/xampp-win32-1.7.7-VC9-installer.exe/download.

Then,double click on that exe document - xampp-win32-1.7.7-VC9-installer.exe.The installer language will pop up. Choose English language and click ‘Ok’.

Figure 5.2: Xampp Server Installer Language .

STEP 2:

Click ‘Next’ button to continue the installation process after welcome Xampp Setup Wizard popup .

Figure 5.3: Xampp Server 1.7.7 Setup Wizard .

STEP 3:

The software will install XAMPPServer in C drive. Now, click on 'Next' button after selecting installation location for XAMPPServer 1.7.7.

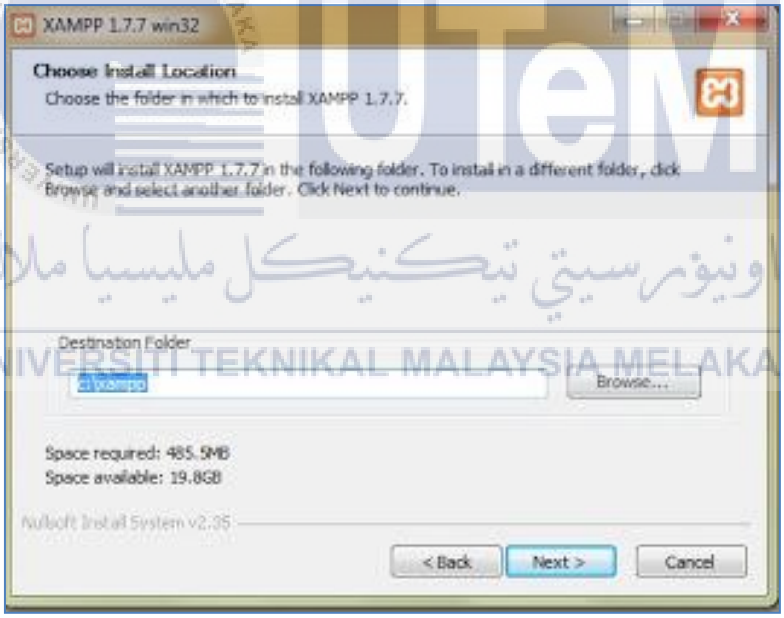


Figure 5.4: Install XamppServer in "C:\" drive.

STEP 4:

Choose any options on XAMPP Options .Then,click 'Install'.

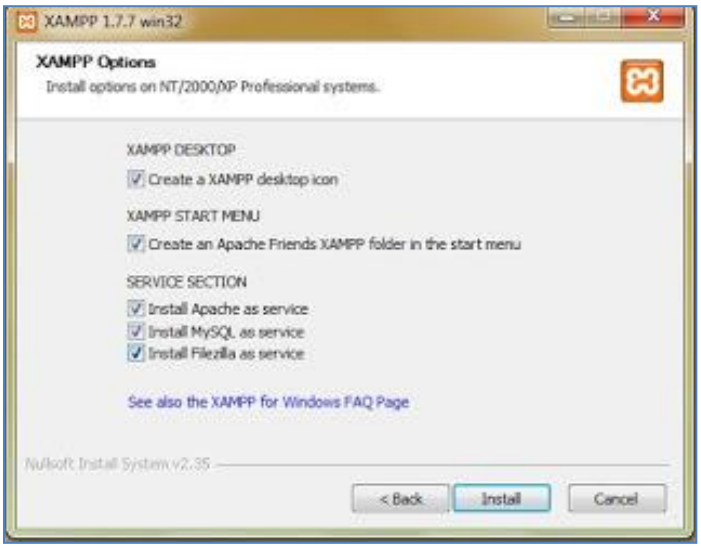


Figure 5.5: Create a Xampp dekstop icon.

STEP 5:

Now, XamppServer is starting to install in a machine.

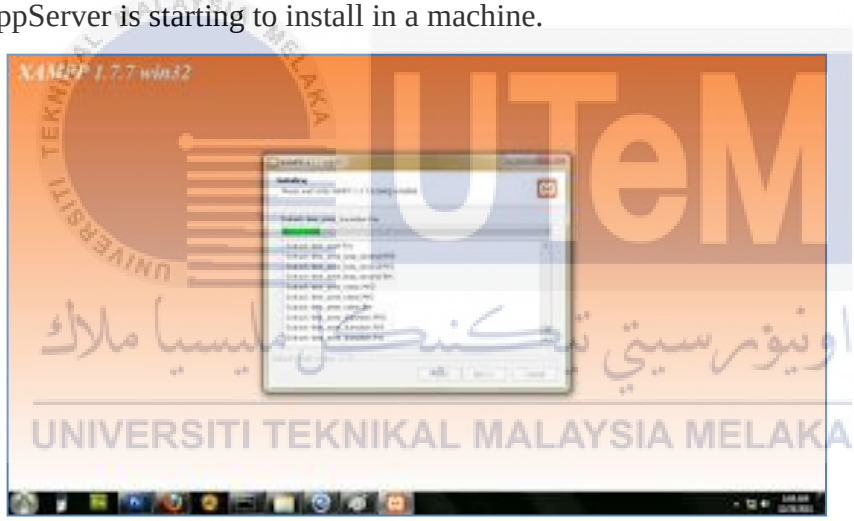


Figure 5.6: Starting Installation.

STEP 6:

The XamppServer icon will appear on screen taskbar .Then click on it. After that, click 'Yes' button when a popup user account control come out.

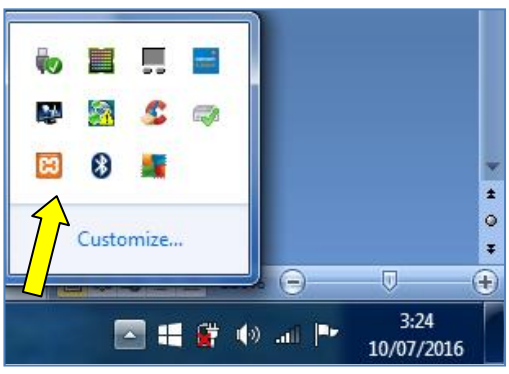


Figure 5.7: Local XamppServer.

STEP 7:

A Xampp control panel application will view.It will show the installation XamppServer 1.7.7 dialog with Apache, MySQL, FileZilla,Mercury and Tomcat at the computer. Checked on the checkbox, then click on 'Start' button.

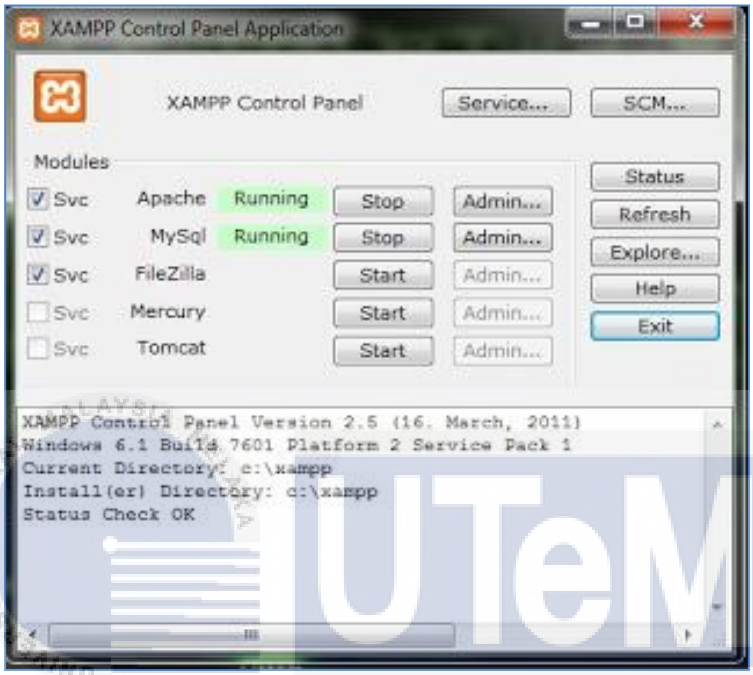


Figure 5.8: Successful Installation

5.2.2 Database Development Setup

Developer setup the database in particular connection with a database which includes the configuration database setup that has described at 5.2.2.1.

5.2.2.1 Configuration Database Setup - Oracle Database Express Edition

Step 1: Installing and configuring Oracle Database

Go to http://www.oracle.com/technology/products/database/xe/index.html, download and install Oracle Database Express Edition (most recent version available):

Follow the installation instructions. When prompted for a password for SYS and SYSTEM accounts, enter "dba".

Step 2: Logging in as the Database Administrator

The first thing needs to do is to log in as the Oracle Database XE Administrator. Follow these steps. Open the Database Home Page login window. On Windows, from the **Start**

menu, select **Programs** (or **All Programs**), then **Oracle Database 10g Express Edition**, and then **Go To Database Home Page**.

While on Linux, click the **Application** menu (on Gnome) or the **K** menu (on KDE), then point to **Oracle Database 10g Express Edition**, and then **Go To Database Home Page**.

At the Database Home Page login window, enter the following information:

• **Username:** Enter system for the username. (JRM)

• **Password:** Enter the password that was specified when Oracle Database XE was installed. (oracle)

Figure 5.9: Database Home Page Login.

Step 3: Click **Login**. The Oracle Database XE home page appears.

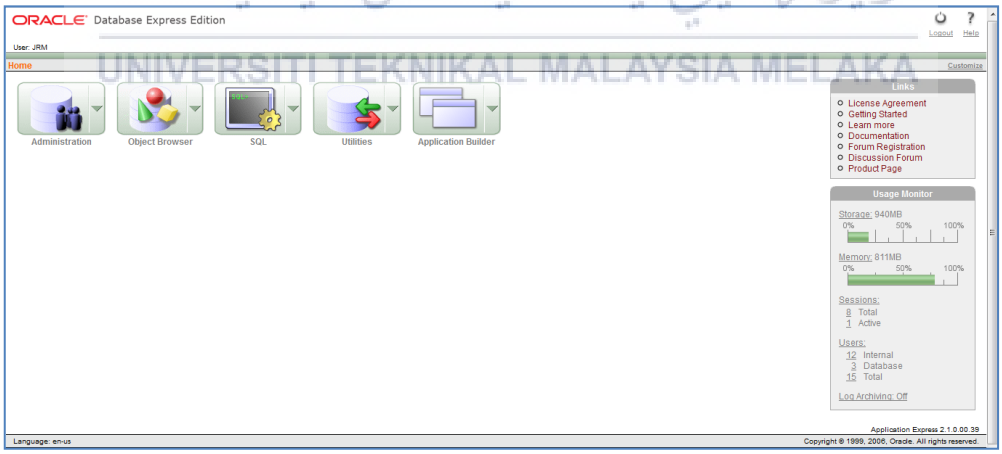


Figure 5.10: Oracle Database XE Home Page.

Step 4: Creating a new account

Select “Administration/Database Users/Create User” via the Oracle Database XE home page. Add an account Scott/tiger with all the privileges checked. After finish chooses the privilege click button “Create” to generate a new user.

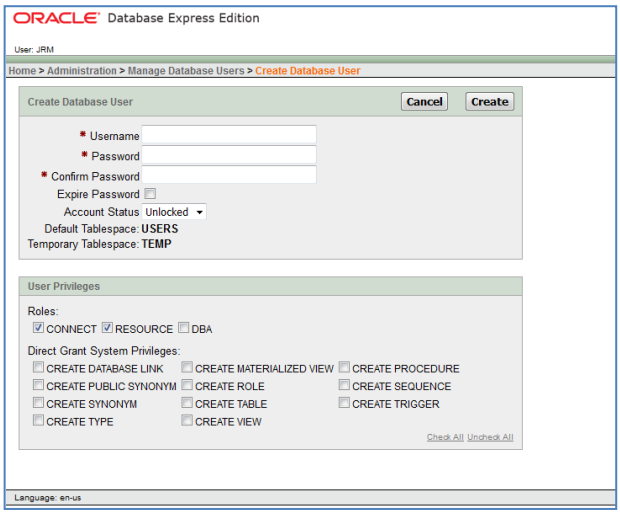


Figure 5.11: Create New User.

5.2.3 Database Creation and Database Objects

Step 1: To create a table use a script that has been created on the notepad it is more easier. Then to build the tables, must generate the script and paste it into the SQL command. To go to the SQL Command page:

i. In the Oracle Database Express Edition home page, click button "SQL".

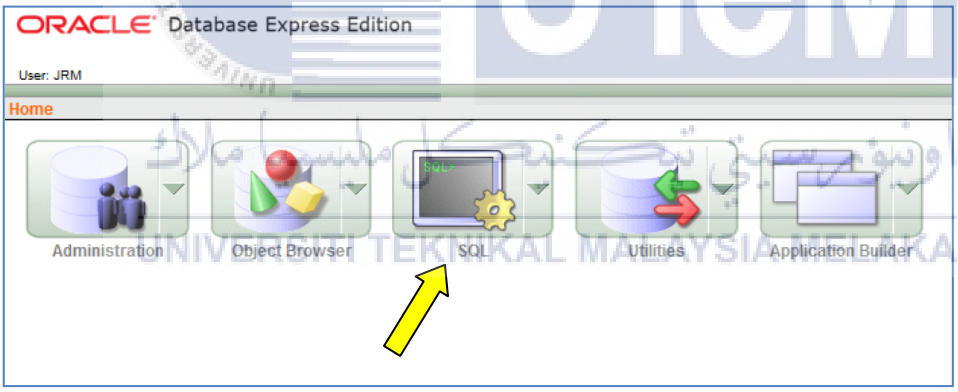


Figure 5.12: Click To "SQL".

ii. After that, on the page, SQL click button "SQL Commands".

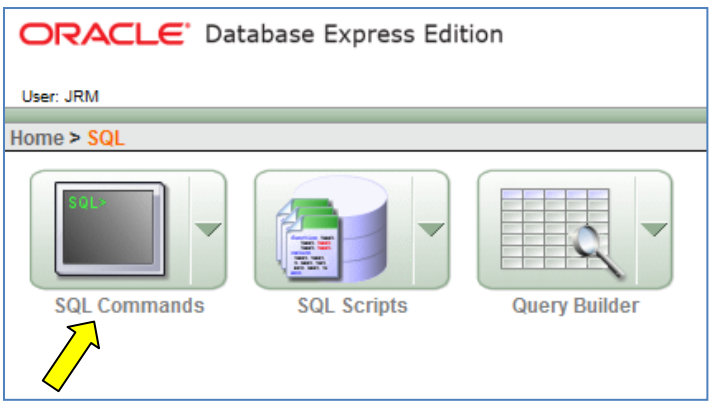


Figure 5.13: Click To "SQL Commands".

iii. On the SQL Commands page, paste all the script at run the script. After that, the that on the Object Browser will show all the table has been created.

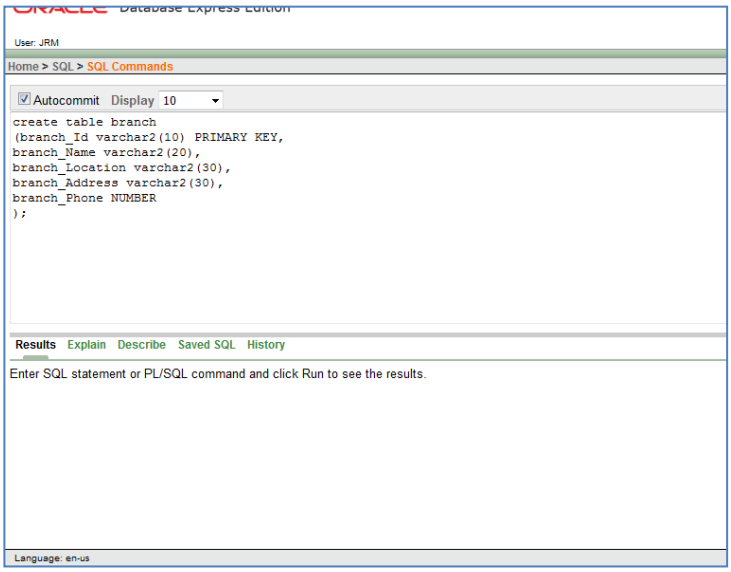


Figure 5.14: Run Script.

Step 2: To view the database object

i. Click to the "Object Browser" to go to the Object Browser page.

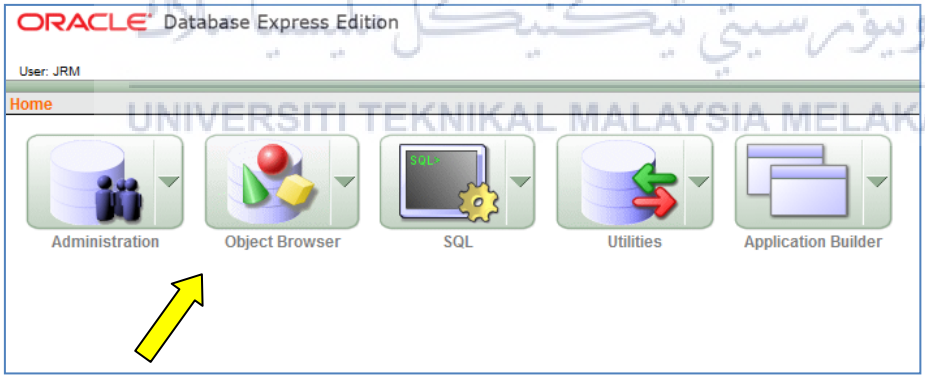


Figure 5.15: Click To "Object Browser".

ii. After go to the "Object Browser page" on the drop down select "Tables". Then the system will display the entire object that has been created by a user.

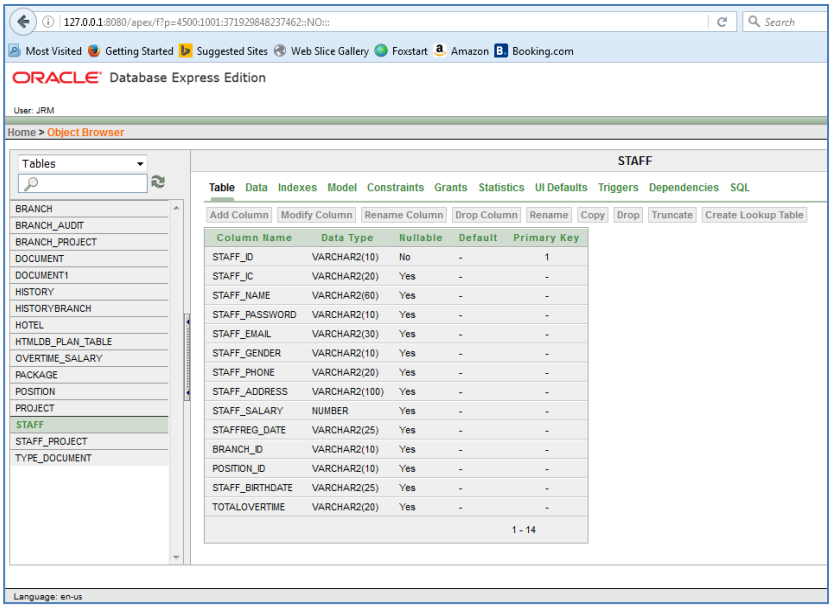


Figure 5.19: Object Browser page.

5.3 Database Implementation

This section will explain about how the database accessing using Oracle query during the development of the URS system. There are a few ways to access the database data:

a. SELECT statement

SELECT statement query uses to show and to display data from the database. The example of query as in Figure 5.3.1:

```
SELECT * FROM STAFF ;
```

Figure 5.3.1: SELECT statement.

b. WHERE clause statement

WHERE clause statement is used to display an information with some condition and the example of query as shown in Figure 5.3.2:

```
select * from staff where staff_ic='910724026146';
```

Figure 5.3.2: WHERE clause.

c. TRIGGER clause

TRIGGER clause will automatically execute when a certain event occurs in the table of the database. The example of the trigger can refer to the Appendix A.

d. Stored Procedure

Stored procedure is used to control the mechanism of the database. This stored procedure will return the result set of the information from the database. The example of the procedure can refer to the Appendix B.

5.4 Conclusion

This chapter describes the implementation phase of the project. It shows the software development setup, database creation and database object, database implementation and the DDL/DCL statements. This phase will cover whether the project fulfilled the requirements or not and can meet user satisfaction or not. The following chapter will cover in the testing phase of the project.

CHAPTER VI

TESTING

6.1 Introduction

This chapter clarifies about the testing period of the e-JRM System. Testing is a procedure of a performing a verity of testing on a venture to investigate the usefulness or to recognize issues. A couple of systems are presupposed amid the testing process and illuminate the analyzer how the task ought to perform and where the regular misstep may found. The test normally attempts to break the framework by entering information that may bring about the framework to return erroneous data.

This part additionally talks about around a test arrangement and it comprises of the test environment and test calendar. The test technique area will examine about the classes of tests. The test outline segment will clarify the test depiction and test information and ultimately is the test outcome and investigation segment.

6.2 Test Plan

A test plan consists of a detailed procedure that specifies how and when the testing will be performed, who will participate, and what test will be used. It should contain scenarios for every possible situation the project could encounter. In this phase, will describe in test organization, testing environment,test schedule,test strategy, unit testing, security testing, stress testing and much more testing that has been done. This is because to ensure no bugs or error happen that can make trouble to system performance and follow the requirement end-users

6.2.1 Test Organization

The test organization is an organization person or company that tests products, materials, software and else according to the requirements. The system is tested by the end-user which is the JRM manager and the staff.

The purpose of the test organization is to determine and to verify that the requirements of a project specification are met. Furthermore, it also makes sure the project is on track and is suitable for end-users.

6.2.2 Test Environment

The test environment is the setup of programming and equipment on which the analyzer perform a testing. The setup of the earth relies on upon a venture that should be tried.

Table 6.1: Test Environment

Description
Personal Computer with minimal requirement: i. RAM - 3.00 GB. ii. Processor – Intel Core i5. iii. Operating System – Windows 7 Profesional.

6.2.3 Test Schedule

The test scheduling is used to arrange the test execution process so that all tester are accommodated.

Table 6.2: Test Schedule

Module/ Component	Activity	Duration	Start Date	End Date
System Login	Unit Integration Testing , Security Test	1 day/5 times	01/07/2016	03/07/2016
Registration Module	Unit Integration Testing	1 day/4 times	04/07/2016	08/07/2016
Management Module	Unit Integration Testing	1 day/3 times	11/07/2016	14/07/2016
Claimation Overtime Module	Unit Integration Testing	1 day/3 times	15/07/2016	19/07/2016

6.3 Test Strategy

The test strategy will describe the project risk are mitigated at the test – level which is types of test are to be performed and which entry and exit requirements apply.

i. Black Box Testing

The black box testing is also known as a functional testing. This testing is based on the project requirement without reference to its internal working. Black box testing is testing on the software design, the tester only knows the input and what expected output should be and not how the project arrives at that output. The tester will not examine the programming code and not need any knowledge of programming compared with the requirements.
This test is used for user acceptance test because it helps to discover bugs that could never identify by using white box testing.

ii. White Box Testing

The white box testing is also known as glass testing or clear box testing or structural testing. This testing based on the knowledge of how the project is implemented. This testing includes analyzing data flow, control flow, information flow, coding practices, and exception and error handling within the system. It can be performed to validate codes that implement

followed intended design, to validate implement security functionality and to uncover exploitable vulnerabilities. This testing also required accessing the project source code. This white box testing can be performed at anytime after the codes develop.

6.3.1 Classes of Tests

Project testing can be done in a several ways such as:

i. Unit testing

Unit testing is about to test the source code for individual unit or method by the developer. A unit to be tested after an initial development, and again after any change or modification. This has been done on code that performs a specific function. Unit testing also important to demonstrate every method in the module works correctly according to the requirements.

ii. Functionality / Project Test

This test is done after the project implementation is completely done. The purpose of this test is to identify defect when the project is complete and the defect cannot be identified during the unit testing. Project testing includes the testing of the project performances, security, the configuration of sensitivity, startup, and recovery from failure mode.

iii. Security test

Security test is used to protect an information and data from unauthorized access, use, disclosure, disruption, modification, or destructions. This process is interrelated with the common goals of protecting of confidentiality, authentication, integrity, and availability of data and information of the project.

6.4 Test Design

6.4.1 Test Description

Tables 6.3 until table 6.7 below are test cases and expected the result to each module and function which had been designed and documented.

Table 6.3 Test Description for User Login

Test Case ID	Description	Testing Type	Expected Result
TC_01-1	Invalid Identity Card Number and Invalid Password	Unit Integration Testing, Security Test	Invalid ID or Password message will appear
TC_01-2	Invalid Identity Card Number	Unit Integration Testing , Security Test	'Please fill out the field' message will appear
TC_01-3	Invalid Password	Unit Integration Testing , Security Test	'Please fill out the field' message will appear
TC_01-4	Blank Identity Card Number and password	Unit Integration Testing , Security Test	'Please fill out the field' message will appear
TC_01-5	Valid Identity Card Number and Password	Unit Integration Testing , Security Test	Successful login message will appear and will go to next page.

Table 6.4: Registration Staff Module

Test Case ID	Description	Testing Type	Expected Result
TC_02-1	Staff Identity Card Number blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-2	Staff Email blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-3	Staff Name blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-4	Staff Position blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-5	Staff Date of Birth	Unit Integration Testing	'Please fill up all the

	blank		form correctly' message will appear
TC_02-6	Staff Address blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-7	Register/Hire Date blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-8	Branch blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-9	Staff Password blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_02-10	Fill the form without blank	Unit Integration Testing	'Successfully Register staff' message will appear

Table 6.5: Add New Type Document Name

Test Case ID	Description	Testing Type	Expected Result
TC_03-1	Type of Document Name blank	Unit Integration Testing	'Please enter type of document'message will appear

Table 6.6: Add Upload New Document

Test Case ID	Description	Testing Type	Expected Result
TC_04-1	Document Category blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear
TC_04-2	Document Number blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear
TC_04-3	Document Name blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear

TC_04-4	Upload Document/image Not upload/blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear
TC_04-5	Document Description blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear
TC_04-6	Project Name blank	Unit Integration Testing	'Filename cannot be empty' a warning message appear

Table 6.7: Add New Project

Test Case ID	Description	Testing Type	Expected Result
TC_05-1	Project Name blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_05-2	Project Start Date blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_05-3	Project End Date blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_05-4	Project Duration blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_05-5	Project Budget blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_05-6	Project Description blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear

Table 6.8: Add New Branch Project

Test Case ID	Description	Testing Type	Expected Result
TC_06-1	Branch Name blank	Unit Integration Testing	'Please fill up all the form' message will appear
TC_06-2	Project Name blank	Unit Integration Testing	'Please fill up all the form' message will appear

Table 6.9: Add New Branch

Test Case ID	Description	Testing Type	Expected Result
TC_07-1	Branch Name blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_07-2	BranchLocation blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_07-3	Branch Address blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_07-4	BranchPhone Number blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear

Table 6.10: Add New Position

Test Case ID	Description	Testing Type	Expected Result
TC_08-1	Position Name blank	Unit Integration Testing	'Please fill up Position Name form correctly' message will appear
TC_08-2	Position Charge Hour blank	Unit Integration Testing	'Please fill up Position Charge Hour form correctly' message will appear

Table 6.11: Add New Overtime Claim

Test Case ID	Description	Testing Type	Expected Result
TC_09-1	Overtime Duration blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_09-2	Claim by blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear
TC_09-3	Project Name blank	Unit Integration Testing	'Please fill up all the form correctly' message will appear

Table 6.12: Upload Document File Type Format

Test Case ID	Description	Testing Type	Expected Result
TC_10-1	Upload JPG.document	Unit Integration Testing	'Successful Upload Document' message will appear
TC_10-2	Upload PNG.document	Unit Integration Testing	'Successful Upload Document' message will appear
TC_10-3	Upload PDF.document	Unit Integration Testing	'Successful Upload Document' message will appear

6.4.2 Test Data

During the test data process, it should contain both correct data and erroneous data and should test all possible situations that will occur. Table 6.12 until table 6.11 below shows the test data that has been implemented:

Table 6.13: Test Data for User Login

Column Name	Test Case ID	Identity Card	Password	Test Result
TD_01-1	TC_01-1	900109146177	Irf90	Invalid IC and Password
TD_01-2	TC_01-2	90010914614	irfan	Invalid IC
TD_01-3	TC_01-3	900109146147	irf	Invalid Password
TD_01-4	TC_01-4			Please fill out this field
TD_01-5	TC_01-5	900109146147	irfan	Successfully login

Table 6.14: Test Data for Staff Registration Module

Column Name	Test Case ID	IC no	Email	Name	Position	Date of birth	Address	Hire Date	Branch	Password	Test Result
TD_02-1	TC_02-1			Wani	Technician	24-07-1991	Perak	11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-2	TC_02-2	910724026146		Wani	Technician	24-07-1991	Perak	11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-3	TC_02-3	910724026146	Haz @gmail.com		Technician	24-07-1991	Perak	11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-4	TC_02-4	910724026146	Haz @gmail.com	Wani		24-07-1991	Perak	11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-5	TC_02-5	910724026146	Haz @gmail.com	Wani	Technician		Perak	11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-6	TC_02-6	910724026146	Haz @gmail.com	Wani	Technician	24-07-1991		11-06-2016	Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-7	TC_02-7	910724026146	Haz @gmail.com	Wani	Technician	24-07-1991	Perak		Perak	123456	'Please fill up all the form correctly' message will appear
TD_02-8	TC_02-8	910724026146	Haz @gmail.com	Wani	Technician	24-07-1991	Perak	11-06-2016		123456	'Please fill up all the form correctly' message appear
TD_02-9	TC_02-9	910724026146	Haz @gmail.com	Wani	Technician	24-07-1991	Perak	11-06-2016	Perak		'Please fill up all the form correctly' message appear
TD_02-10	TC_02-10	910724026146	Haz @gmail.com	Wani	Technician	24-07-1991	Perak	11-06-2016	Perak		Successfully Add New Staff

\

Table 6.15: Test Data for Type Document Name

Column Name	Test Case ID	Type of Document Name	Test Result
TD_03-1	TC_03-1		Please fill out this field
TD_03-2	TC_03-2	INVOICE	Successfully Add New Type Doc. Name

Table 6.16: Test Data for Upload New Document

Column Name	Test Case ID	Document Category	Document Number	Document Name	Document Upload	Document Description	Project Name	Test Result
TD_04-1	TC_04-1		00001356793	Receipt Minyak Petronas 30 Apr 2014	PNG	PRIMAX 95 XTRA	Outstation work	'Filename cannot be empty' a warning message appear
TD_04-2	TC_04-2	RECEIPT		Receipt Minyak Petronas 30 Apr 2014	PNG	PRIMAX 95 XTRA	Outstation work	'Filename cannot be empty' a warning message appear
TD_04-3	TC_04-3	RECEIPT	00001356793		PNG	PRIMAX 95 XTRA	Outstation work	'Filename cannot be empty' a warning message appear
TD_04-4	TC_04-4	RECEIPT	00001356793	Receipt minyak petronas 30 Apr 2014		PRIMAX 95 XTRA	Outstation work	'Filename cannot be empty' a warning message appear
TD_04-5	TC_04-5	RECEIPT	00001356793	Receipt minyak petronas 30 Apr 2014	PNG		Outstation work	'Filename cannot be empty' a warning message appear
TD_04-6	TC_04-6	RECEIPT	00001356793	Receipt minyak petronas 30 Apr 2014	PNG	PRIMAX 95 XTRA		'Filename cannot be empty' a warning message appear
TD_04-7	TC_04-7	RECEIPT	00001356793	Receipt minyak petronas 30 Apr 2014	PNG	PRIMAX 95 XTRA	Outstation work	Succesfully Upload Document File

Table 6.17: Test Data for Add New Project

Column Name	Test Case ID	Project Name	Project Start Date	Project End Date	Project Duration (days)	Project Budget(RM)	Project Description	Test Result
TD_05-1	TC_05-1		2016-06-03	2016-31-07	58	10.000	Outstation Local & Foreign Country	'Please fill up all the form correctly' message will appear
TD_05-2	TC_05-2	Oustation work		2016-31-07	58	10.000	Outstation Local & Foreign Country	'Please fill up all the form correctly' message will appear
TD_05-3	TC_05-3	Oustation work	2016-06-03		58	10.000	Outstation Local & Foreign Country	'Please fill up all the form correctly' message will appear
TD_05-4	TC_05-4	Oustation work	2016-06-03	2016-31-07		10.000	Outstation Local & Foreign Country	'Please fill up all the form correctly' message will appear
TD_05-5	TC_05-5	Oustation work	2016-06-03	2016-31-07	58		Outstation Local & Foreign Country	'Please fill up all the form correctly' message will appear
TD_05-6	TC_05-6	Oustation work	2016-06-03	2016-31-07	58	10.000		'Please fill up all the form correctly' message will appear
TD_05-7	TC_05-7	Oustation work	2016-06-03	2016-31-07	58	10.000	Outstation Local & Foreign Country	Succesfully Add New Project

Table 6.18: Test Data for Add New Branch Project

Column Name	Test Case ID	Branch Name	Project Name	Test Result
TD_06-1	TC_06-1		Air Conditioner project/Repair Services	'Please fill up all the form' message will appear
TD_06-2	TC_06-2	KUALA LUMPUR		'Please fill up all the form' message will appear
TD_06-3	TC_06-3	KUALA LUMPUR	Air Conditioner project/Repair Services	Successfully Add New Branch Project

Table 6.19: Test Data for Add New Branch

Column Name	Test Case ID	Branch Name	Branch Location	Branch Address	Branch Phone	Test Result
TD_07-1	TC_07-1		Ipoh	Wisma TM Jalan Sultan Idris Shah 30673 Ipoh ,Perak	055336674	'Please fill up all the form correctly' message will appear
TD_07-2	TC_07-2	PERAK		Wisma TM Jalan Sultan Idris Shah 30673 Ipoh ,Perak	055336674	'Please fill up all the form correctly' message will appear
TD_07-3	TC_07-3	PERAK	Ipoh		055336674	'Please fill up all the form correctly' message will appear
TD_07-4	TC_07-4	PERAK	Ipoh	Wisma TM Jalan Sultan Idris Shah 30673 Ipoh ,Perak		'Please fill up all the form correctly' message will appear
TD_07-5	TC_07-5	PERAK	Ipoh	Wisma TM Jalan Sultan Idris Shah 30673 Ipoh ,Perak	055336674	Successfully Add New Branch

Table 6.20: Test Data for Add New Position

Column Name	Test Case ID	Position Name	Position Charge Hour/Hour	Test Result
TD_08-1	TC_08-1		35	'Please fill up all the form' message will appear
TD_08-2	TC_08-2	WEB DESIGNER		'Please fill up all the form' message will appear
TD_08-3	TC_08-3	WEB DESIGNER	35	Successfully Add New Position

Table 6.21: Test Data for Add New Overtime Claim

Column Name	Test Case ID	Overtime Duration(Hour)	Claim by	Project Name	Test Result
TD_09-1	TC_09-1		Wan Kamarul Bin Wan Zainal	Repair & Services	'Please fill up all the form' message will appear
TD_09-2	TC_09-2	5		Repair & Services	'Please fill up all the form' message will appear
TD_09-3	TC_09-3	5	Wan Kamarul Bin Wan Zainal		'Please fill up all the form' message will appear
TD_09-4	TC_09-4	5	Wan Kamarul Bin Wan Zainal	Repair & Services	Successfully Add New Overtime Claim

Table 6.22: Test Data for View Document Type Format

Column Name	Test Case ID	Type of Document Upload	Test Result
TD_10-1	TC_10-1	Upload JPG.document	Successfully View JPG. document file
TD_10-2	TC_10-2	Upload PNG. document	Successfully View PNG. document file
TD_10-3	TC_10-3	Upload PDF. document	Unsuccessfully View PDF. document file

6.5 Test Result and Analysis

The test result is the result obtains from the testing activities that have been executed by the tester. Table 6.12 until table 6.15 shows the results that have been generating based on the system execution.

Table 6.23: Test Result and Analysis for User Login

Module/Component		Result		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_01-1	TC_01-1	Invalid Username	✓	
TD_01-2	TC_01-2	Invalid Password	✓	
TD_01-3	TC_01-3	Invalid Username/Password	✓	
TD_01-4	TC_01-4	Username/Password field blank	✓	

Table 6.24: Test Result and Analysis for Staff Registration

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_02-1	TC_02-1	Identity Number Blank	✓	
TD_02-2	TC_02-2	Email Blank	✓	
TD_02-3	TC_02-3	Position Blank	✓	
TD_02-4	TC_02-4	Date of Birth blank	✓	
TD_02-5	TC_02-5	Address Blank	✓	
TD_02-6	TC_02-6	Branch Blank	✓	
TD_02-7	TC_02-7	Password Blank	✓	

Table 6.25: Test Result and Analysis for New Type Of Document

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_03-1	TC_03-1	Type Of Document Blank	✓	

Table 6.26: Test Result and Analysis for Upload New Document

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_04-1	TC_04-1	Document Category Blank	✓	
TD_04-2	TC_04-2	Document Number Blank	✓	
TD_04-3	TC_04-3	Document Name Blank	✓	
TD_04-4	TC_04-4	Document Upload Blank	✓	
TD_04-5	TC_04-5	Document Description Blank	✓	
TD_04-6	TC_04-6	Project Name	✓	

Table 6.27: Test Result and Analysis for New Project

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_05-1	TC _05-1	Project Name Blank	✓	
TD_05-2	TC _05-2	Project Start Date Blank	✓	
TD_05-3	TC _05-3	Project End Blank	✓	

TD_05-4	TC _05-4	Project Duration (days)Blank	✓	
TD_05-5	TC _05-5	Project Budget(RM) Blank	✓	
TD_05-6	TC_05-6	Project Description	✓	

Table 6.28: Test Result and Analysis for New Branch Project

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_06-1	TC _06-1	Branch Name Blank	✓	
TD_06-2	TC _06-2	Project Name Blank	✓	

Table 6.29: Test Result and Analysis for New Branch

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_07-1	TC _07-1	Branch Name Blank	✓	
TD_07-2	TC _07-2	Branch Location Blank	✓	
TD_07-3	TC _07-3	Branch Address Blank	✓	

TD_07-4	TC _07-4	Branch Phone Blank	✓	

Table 6.30: Test Result and Analysis for New Position

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_08-1	TC _08-1	Position Name Blank	✓	
TD_08-2	TC _08-2	Position Charge Hour/Hour	✓	

Table 6.31: Test Result and Analysis for New Overtime Claim

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**
TD_09-1	TC _09-1	Overtime Duration(Hour) Blank	✓	
TD_09-2	TC _09-2	Claim by Blank	✓	
TD_09-3	TC _09-3	Project Name Blank	✓	

Table 6.32: Test Result and Analysis for View Document Type Format

Module/Component		**Result**		
Test Data ID	**Test Case ID**	**Description**	**OK**	**Failed**

TD_10-1	TC _10-1	Upload JPG.document	✓	
TD_10-2	TC _10-2	Upload PNG.document	✓	
TD_10-3	TC _10-3	Upload PDF.document		x

The system successfully uploaded document in JPG, PNG and PDF format but only cannot view the PDF format.

6.6 Conclusion

For the conclusion, this project met the end-user requirement. During testing the system, there is no error happen because the project has undergone several testing phases like always make discussion with user and supervisor project to fulfill the requirements before the project can be released to the user.

Most of all, the testing purpose in project development is to make sure that there is no bug or errors happen when using this system.

CHAPTER VII

CONCLUSION

7.1 Introduction

This chapter will discuss the conclusion of the project which are strengths and the weaknesses of the project. In addition, a proposition for improvement and contribution to the project in the future also discuss in this chapter.

7.2 Observation on Strength and Weaknesses

The development of this project, e-JRM Digital Document Storage Management System has their strength and weaknesses. The strength and weaknesses have been identified at the end of the project development.

7.2.1 Strength

There are strengths of the e-JRM project :-

i. Only an authorized person (Manager/Admin) can access system to manage the whole system because it has been protected with an identity card and password based on position

ii. Only an authorized staff can access system and make information such as project details, branch details,document details ,upload document and overtime claymation.

iii. The manager gives the reset password to the staff for login and then , they can reset their password.

iv. The system is secured as it will automatically log out after one minute not use the system .

v. Registration and claymation overtime can be done on working hour between 8 a.m until 6 p.m only.

vi. The system can detect duplicate/overlapping details position charge hour when to create a new position , so the manager doesn't need to fill in the same position charge hour again.

vii. The user can detect duplicate document uploaded when inserting document number at the searching form in the systems. So, admin can delete the wrong and duplicate document uploaded.

7.2.2 Weaknesses

There are project weaknesses that have been identified during the project development:-

i. User unable to re-insert image document for the second time.

ii. If the user forgets their password, they need to inform manager to give the reset password.

iii. The user can upload a document in format jpg. and png. only.

7.3 Proposition for Improvement

For future improvement of the e-JRM, the system needs to make a different format for upload document which is pdf format. In some situation it may have many conditions will be happened such as if need to upload large pages document . The system can check for overtime claim by click on monthly and year . Furthermore, the e-JRM system needs to be more graphical functions to make it become more interactive, easy to access and handle by admin and staff.

7.4 Contribution

Based on the system development, this system can help the user on managing the duplicate document and position charge details also manage the registration staff, inserting new project and branch,searching location staff ,branch and project, uploading document and convenience for staff for instant calculation overtime claim at the end of the month .

7.5 Conclusion

As a conclusion, objectives of this project have been achieved during the project development. However, this project still has a few weaknesses in several functions and need some improvement in the future.

REFERENCES

1. JRM Services Facebook Page . (2012).
   Retrieved February 27, 2016, from https://www.facebook.com/jrmservices

2. Google Maps .
   Retrieved February 27, 2016, from https://www.google.com/maps/place/JRM+Services+Sdn+Bhd

3. MESWARA website (2012).
   Retrieved March 04, 2016, from http://www.meswara.biz/cms/blog/sistem-penyimpanan-dokumen

4. DDMS Digital Document Management System MAMPU. (2014).
   Retrieved March 04, 2016, from https://ddms.mampu.gov.my/

APPENDIX A
(Create Trigger Schema)

APPENDIX A

Figure 4.11 until Figure 4.58 shows the list of trigger(s) and procedure(s). This trigger divided into two categories which are trigger before and trigger after.

Figure 4.11: Create "Trigger before Insert into Staff Tables"

‘

```
create or replace trigger pk_staff_Id
Before insert on staff
for each row
declare s_Id staff.staff_Id%TYPE;
BEGIN
Select staff_seq.nextval into s_Id from dual;
:new.staff_Id:='S'||s_Id;
end;
```

Figure 4.12: Create "Trigger before Insert into Branch Tables"

```
create or replace trigger pk_branch_Id
Before insert on branch1
for each row
declare b_Id branch1.branch_Id%TYPE;
BEGIN
Select branch_seq.nextval into b_Id from dual;
:new.branch_Id:='B'||b_Id;
end;
```

Figure 4.13: Create "Trigger before Insert into Type_Document Tables"

```
Create Or Replace Trigger Pk_Type_Doc_Id
Before Insert On Type_Document
For Each Row
Declare Type_Doc_Id  TYPE_DOCUMENT.Type_Doc_Id %TYPE;
BEGIN
Select Type_Doc_Seq.Nextval Into Type_Doc_Id  From Dual;
:New.Type_Doc_Id :='TD'||Type_Doc_Id ;
End;
```

Figure 4.14: Create "Trigger before Insert into Project Tables"

```
create or replace trigger pk_pro_Id
Before insert on project
for each row
declare pro_Id project.pro_Id%TYPE;
BEGIN
select project_seq.nextval into pro_Id from dual;
:new.pro_Id:='PR'||pro_Id;
end;
```

Figure 4.15: Create "Trigger before Insert into staff_project Tables"

```
create or replace trigger pk_sPro_Id
Before insert on staff_project
for each row
declare sPro_Id staff_project.sPro_Id%TYPE;
BEGIN
Select sPro_seq.nextval into sPro_Id from dual;
:new.sPro_Id:='SP'||sPro_Id;
end;
```


Figure 4.16: Create "Trigger before Insert into branch_project Tables"

```
Create Or Replace Trigger Pk_Bproj_Id
Before Insert On Branch_Project
For Each Row
Declare Bproj_Id Branch_Project.Bproj_Id%Type;
Begin
Select Bproj_Seq.Nextval Into Bproj_Id From Dual;
:New.Bproj_Id:='Bp'||Bproj_Id;
End;
```

Figure 4.16: Create "Trigger before Insert into overtime_salary Tables"

```
Create or replace trigger pk_overtime_Id
Before insert on overtime_salary
For each row
Declare overtime_Id
Overtime_salary.overtime_Id%TYPE;
BEGIN
Select overtime_seq.nextval into overtime_Id from dual;
:new.overtime_Id:= 'OT'||overtime_Id;
End;
```

Figure 4.18: Create "Trigger before Insert into document Tables"

```
Create Or Replace Trigger Pk_Doc_Id
Before Insert On Document
For Each Row
Declare Doc_Id Document.Doc_Id%Type;
Select Doc_Id_Seq.Nextval Into Doc_Id From Dal;
:New.Doc_Id:= 'D'||Doc_Id;
End;
```


Figure 4.19: Create "Trigger before Insert into staff project Tables"

```
create or replace TRIGGER session_clientOT
 BEFORE INSERT ON STAFF_PROJECT
BEGIN
IF(TO_CHAR(SYSDATE,'DY') IN ('SAT','SUN')) OR
    (TO_CHAR(SYSDATE,'HH24:MI')
        NOT BETWEEN '08:00' AND '18:00')
THEN RAISE_APPLICATION_ERROR (-20500,'YOU MAY
INSERT INTO CLIENT TABLE ONLY DURING BUSINESS
HOURS.');
END IF;
END;
```

Figure 4.20: Create "Trigger before Insert into staff Tables"

```
create or replace TRIGGER session_clientStaffReg
 BEFORE INSERT ON STAFF
BEGIN
IF(TO_CHAR(SYSDATE,'DY') IN ('SAT','SUN')) OR
    (TO_CHAR(SYSDATE,'HH24:MI')
          NOT BETWEEN '08:00' AND '18:00')
THEN RAISE_APPLICATION_ERROR (-20500,'YOU MAY INSERT
INTO CLIENT TABLE ONLY DURING BUSINESS HOURS.');
END IF;
END;
```

Figure 4.21: Create "Trigger before Insert or update into type document Tables"

```
Create Or Replace Trigger Td
 Before Insert Or Update On Type_Document
For Each Row
Begin
:New.Type_Doc_Name := Upper(:New.Type_Doc_Name);
End;
```

Figure 4.22: Create "Trigger before Insert or update into type document Tables"


```
Create Or Replace Trigger Check_Duplicatetype
   Before Insert Or Update On Type_Document
   For Each Row  Declare
   V_Dup Number;
   Begin   Select Count(Type_Doc_Name) Into V_Dup From
Type_Document Where Type_Doc_Name=:New.Type_Doc_Name ;
  If V_Dup > 0 Then
     Raise_Application_Error (-20100, 'This Property Already Exists. The
Insert Is Cancelled.');
   End If;
   End;
```

Figure 4.23: Create "Trigger before Insert or update into branch Tables"

```
Create Or Replace Trigger Branch_Name
Before Insert Or Update On Branch
For Each Row
Begin
:New.Branch_Name := Upper(:New.Branch_Name);
End;
```

Figure 4.24: Create "Trigger after Insert into staff project Tables"

```
Create Or Replace Trigger Trigot
After Insert On Staff_Project
For Each Row
Declare Totalovertime Varchar2(20);
V_Charge_Hour  Position.Position_Charge_Hour%Type;
Begin Select P.Position_Charge_Hour
Into V_Charge_Hour
From Position P, Staff S
Where S.Staff_Id = :New.Staff_Id
And P.Position_Id = S.Position_Id;
Update Staff
Set Totalovertime=
 (Nvl(:New.Job_Hour,0)* V_Charge_Hour)
Where Staff_Id =:New.Staff_Id; End;
```


Figure 4.25: Create "Trigger after update into position Tables"

```
Create Or Replace Trigger Rec_Position
After Update Of Position_Description On Position
 For Each Row
 When (New.Position_Description = Old.Position_Description)
Begin
 Raise_Application_Error (-20508, 'This Position_Charge_Hour Have
Inserted,Please Insert Another Again !!!');
End;
```

Figure 4.26: Create "Trigger after delete into branch_audit Tables"

```
create or replace trigger branch_trig
after delete on branch for each row
begin insert into branch_audit
values (:old.branch_Id,:old. branch_Name,
:old. branch_Location,
:old. branch_Address,
:old. branch_Phone); end;
```

APPENDIX B
(Create Stored Procedure Schema)

APPENDIX B

Figure 4.27: Create Stored Procedure "insert_registerStaff_proc**"**

```
Create Or Replace Procedure Insert_Registerstaff_Proc
(
V_Ic Varchar2,
V_Name Varchar2,
V_Password Varchar2,
V_Email Varchar2,
V_Gender Varchar2,
V_Phone Number,
V_Address Varchar2,
V_Regdate Varchar2,
V_Branch_Id Varchar2,
V_Position_Id Varchar2,
V_Birthdate Varchar2) IS
BEGIN
INSERT INTO STAFF
(Staff_Ic, Staff_Name, Staff_Password, Staff_Email,Staff_Gender,
Staff_Phone, Staff_Address, Staffreg_Date,
Branch_Id,Position_Id,Staff_Birthdate)
VALUES(V_Ic, V_Name,V_Password, V_Email
,V_Gender,V_Phone,V_Address,V_Regdate,
V_Branch_Id,V_Position_Id,V_Birthdate);
End;
```


Figure 4.28: Create Stored Procedure "updatestaff_proc**"**

```
Create Or Replace Procedure Updatestaff_Proc
(S_Ic In Out Staff.Staff_Ic%Type,
S_Name In Out Staff.Staff_Name%Type,
S_Address In Out Staff.Staff_Address%Type,
S_Phone In Out Staff.Staff_Phone%Type,
S_Email In Out Staff.Staff_Email%Type,
S_Salary In Out Staff.Staff_Salary%Type
)
As
Begin Update Staff Set Staff_Ic = S_Ic,Staff_Name = S_Name,
Staff_Address = S_Address,  Staff_Phone = S_Phone ,Staff_Email  =
S_Email,
Staff_Salary = S_Salary
Where Staff_Ic = S_Ic; Commit;
End;
```

Figure 4.29: Create Stored Procedure "list_OTclaim_proc**"**

```
Create Or Replace Procedure List_Otclaim_Proc
(
Rc Out Sys_Refcursor)
As
Begin
 Open  Rc For Select * From Overtime_Salary;
End;
```

Figure 4.30: Create Stored Procedure "DELETEListOT**"**

```
Create Or Replace Procedure  Deletelistot (
O_Overtime_Id In Overtime_Salary.Overtime_Id%Type)
Is
Begin
Delete From Overtime_Salary Where Overtime_Id=O_Overtime_Id;
End;
```

Figure 4.31: Create Stored Procedure "deleteot**"**


```
Create Or Replace Procedure  Deleteot (
S_Spro_Id In Staff_Project.Spro_Id%Type)
Is
Begin
Delete From Staff_Project Where Spro_Id=S_Spro_Id;
End;
```

Figure 4.32: Create Stored Procedure "listdata_SP**"**

```
create or replace procedure listdata_SP
(
rc out sys_refcursor)
as
begin
 open  rc for select * from STAFF_PROJECT ;
end;
```

Figure 4.33: Create Stored Procedure "insert_Position**"**

```
create or replace procedure insert_Position
(p_position_Description varchar2,
p_position_Charge_Hour NUMBER)
IS
BEGIN
insert into
position(position_Description,position_Charge_Hour)values(p_position_D
escription,p_position_Charge_Hour);
END;
```

Figure 4.34: Create Stored Procedure "listPOSITION_proc**"**

```
Create Or Replace Procedure Listposition_Proc
(Rc Out Sys_Refcursor)
As Begin
 Open  Rc For Select * From Position ;
End;
```

Figure 4.35: Create Stored Procedure "updateposition_proc**"**


```
Create Or Replace Procedure Updateposition_Proc
(B_Id In Out Position.Position_Id%Type,
B_Desc In Out Position.Position_Description%Type,
B_Charge In Out Position.Position_Charge_Hour%Type)
As Begin
Update Position Set Position_Id = B_Id,Position_Description = B_Desc,
Position_Charge_Hour  =B_Charge
Where Position_Id = B_Id;
Commit;End;
```

Figure 4.36: Create Stored Procedure "deleteposition **"**

```
Create Or Replace Procedure  Deleteposition (
P_Position_Id In Position.Position_Id%Type)
Is
Begin
Delete From Position Where Position_Id=P_Position_Id;
End;
```

Figure4.3:Create Stored Procedure "insert_registerBranch_proc**"**

```
Create Or Replace Procedure Insert_Registerbranch_Proc
(
V_Name Varchar2,
V_Location Varchar2,
V_Address Varchar2,
V_Phone Number)
Is
Begin
Insert Into Branch
( Branch_Name, Branch_Location, Branch_Address,Branch_Phone)
Values(V_Name, V_Location, V_Address ,V_Phone);
End;
```

Figure 4.38: Create Stored Procedure"insert_ RegisterTD _proc**"**

```
Create Or Replace Procedure Insert_Registertd_Proc
(
V_Address Varchar2)
Is
Begin
Insert Into Type_Document
( Type_Doc_Name)
Values( V_Address );
End;
```


Figure 4.39: Create Stored Procedure "insert_ typeofDoc _proc**"**

```
Create Or Replace Procedure Insert_Typeofdoc_Proc
(
V_Type_Doc_Name Varchar2
)
Is
Begin
Insert Into Type_Document
( Type_Doc_Name)
Values(V_Type_Doc_Name);
End;
```

Figure 4.40: Create Stored Procedure "insert_ Project **"**

```
Create Or Replace Procedure Insert_Project
(P_Pro_Name Varchar2,
P_Pro_Start_Date Varchar2,
P_Pro_End_Date Varchar2,
P_Pro_Duration Number,
P_Pro_Budget Varchar2,
P_Pro_Description Varchar2
)Is Begin
Insert Into Project(Pro_Name,Pro_Start_Date, Pro_End_Date,
Pro_Duration,Pro_Budget,  Pro_Description)
Values(P_Pro_Name,P_Pro_Start_Date,P_Pro_End_Date,P_Pro_Duration,
P_Pro_Budget,P_Pro_Description);End;
```

Figure 4.41: Create Stored Procedure "insert_ Overtime **"**

```
Create Or Replace Procedure Insert_Overtime
(V_Overtime_Duration Varchar2,
V_Staff_Overtimesalary Varchar2,
V_Staff_Overtimemonth Varchar2,
V_Year Varchar2,
V_Staff_Id Varchar2,
V_Pro_Id Varchar2)
Is Begin Insert Into
Overtime_Salary(Overtime_Duration,Staff_Overtimesalary ,
Staff_Overtimemonth, Year, Staff_Id,  Pro_Id )
Values(V_Overtime_Duration,V_Staff_Overtimesalary,
V_Staff_Overtimemonth,V_Year,V_Staff_Id,V_Pro_Id);End;
```


Figure 4.42: Create Stored Procedure "insert_ BP"

```
Create Or Replace Procedure Insert_Bp
(P_Branch_Id Varchar2,
P_Pro_Id Varchar2)
Is Begin
Insert Into Branch_Project(Branch_Id,Pro_Id)
Values(P_Branch_Id,P_Pro_Id);
End;
```

Figure 4.43: Create Stored Procedure "list_ Branch_Project**"**

```
Create Or Replace Procedure Listbranchproproc
( V_Branch_Id In Branch.Branch_Id%Type,
Rc Out Sys_Refcursor)
As
Bbranch_Id Varchar2(10);
 Begin
 Select Branch_Id Into Bbranch_Id From Branch Where
Branch_Location = V_Branch_Id;
 Open Rc For Select * From Branch_Project Bp,Branch B ,Project Pr
Where B.Branch_Id =Bp.Branch_Id And Bp.Branch_Id=  Bbranch_Id
And Bp.Pro_Id =Pr.Pro_Id;
End;
```

Figure 4.44: Create Stored Procedure "listBranchPROproc**"**

```
Create Or Replace Procedure Listbranchproc
( V_Branch_Name In Branch.Branch_Name%Type,
Rc Out Sys_Refcursor)
As
 Begin
 Open Rc For Select * From Branch
Where Branch_Name = V_Branch_Name;
End;
```


Figure 4.45: Create Stored Procedure "listStaffPROproc**"**

```
Create Or Replace Procedure Liststaffproproc
(
V_Staff_Id In Staff.Staff_Id%Type,
Rc Out Sys_Refcursor
)
As
Sstaff_Id Varchar2(10);
 Begin
 Select Staff_Id Into Sstaff_Id From Staff Where Staff_Ic =
V_Staff_Id;
 Open Rc For Select * From Staff_Project Sp,Staff S ,Project Pr
Where S.Staff_Id =Sp.Staff_Id And Sp.Staff_Id=  Sstaff_Id And
Sp.Pro_Id =Pr.Pro_Id;
End;
```

Figure 4.46: Create Stored Procedure "listdocproc**"**

```
Create Or Replace Procedure Listdocproc
( V_Doc_Number In Document.Doc_Number%Type,
Rc Out Sys_Refcursor)
As Begin
 Open Rc For Select * From Document
Where Doc_Number = V_Doc_Number; End;
```

Figure 4.47: Create Stored Procedure "insert_RegisterTD_proc**"**

```
Create Or Replace Procedure Insert_Registertd_Proc
(V_Address Varchar2)
Is
Begin
Insert Into Type_Document
( Type_Doc_Name)
Values( V_Address );
End;
```

Figure 4.48: Create Stored Procedure "p_insert_doc


```
Create Or Replace Procedure P_Insert_Doc(
P1 In Varchar,
P2 In Varchar,
P3 In Varchar,
P4 In Varchar,
P5 In Number,
P6 In Varchar,
P7 In Varchar,
P8 In Varchar,
P9 In Varchar)
As Begin Insert Into Document (Doc_Id,
Doc_Date_Upload,
Doc_Number,
Doc_Name,
Doc_Description,
Doc_Image,
Staff_Ic,
Type_Doc_Id,
Pro_Id)
Values (P1,P2,P3,P4,P5,P6,P7,P8,P9);
Commit; End;
```

Figure 4.49: Create Stored Procedure "updatedoc_proc**"**

```
Create Or Replace Procedure Updatedoc_Proc
(S_Doc_Id In Out Document.Doc_Id%Type,
S_Doc_Date_Upload In Out Document.Doc_Date_Upload%Type,
S_Doc_Number In Out Document.Doc_Number%Type,
S_Doc_Name In Out Document.Doc_Name%Type,
S_Doc_Description In Out Document.Doc_Description%Type,
S_Pro_Id In Out Document.Pro_Id%Type,
S_Type_Doc_Id In Out Document.Type_Doc_Id%Type
)
As
Begin
Update Document Set Doc_Id = S_Doc_Id, Doc_Date_Upload =
S_Doc_Date_Upload, Doc_Number = S_Doc_Number,
Doc_Name = S_Doc_Name,  Doc_Description = S_Doc_Description ,
Pro_Id  = S_Pro_Id, Type_Doc_Id  = S_Type_Doc_Id
Where Doc_Id = S_Doc_Id;
Commit;
End;
```


Figure 4.50: Create Stored Procedure "update_proc"

```
Create Or Replace Procedure Update_Proc
(S_Ic In Out Staff.Staff_Ic%Type,
S_Name In Out Staff.Staff_Name%Type,
S_Address In Out Staff.Staff_Address%Type,
S_Phone In Out Staff.Staff_Phone%Type,
S_Email In Out Staff.Staff_Email%Type)
As
Begin
Update Staff Set Staff_Ic = S_Ic,Staff_Name = S_Name, Staff_Address =
S_Address,  Staff_Phone = S_Phone ,Staff_Email  = S_Email
Where Staff_Ic = S_Ic;
Commit;
End;
```

Figure 4.51: Create Stored Procedure "updatebranch_proc"

```
Create Or Replace Procedure Updatebranch_Proc
(B_Id In Out Branch.Branch_Id%Type,
B_Name In Out Branch.Branch_Name%Type,
B_Location In Out Branch.Branch_Location%Type,
B_Address In Out Branch.Branch_Address%Type,
B_Phone In Out Branch.Branch_Phone%Type)
As Begin
Update Branch Set Branch_Id = B_Id,Branch_Name = B_Name,
Branch_Location = B_Location,  Branch_Address=B_Address
,Branch_Phone  =B_Phone  Where Branch_Id = B_Id;
Commit;End;
```

Figure 4.52 :Create Stored Procedure "updateOTclaim_proc"

```
Create Or Replace Procedure Updateotclaim_Proc
(S_Overtime_Duration In Out
Overtime_Salary.Overtime_Duration%Type,
S_Staff_Overtimesalary In Out
Overtime_Salary.Staff_Overtimesalary%Type,S_Staff_Overtimemonth In
Out Overtime_Salary.Staff_Overtimemonth%Type,
S_Year In Out Overtime_Salary.Year%Type,
S_Staff_Id In Out Overtime_Salary.Staff_Id%Type,
S_Pro_Id In Out Overtime_Salary.Pro_Id%Type)
As Begin Update Overtime_Salary Set Overtime_Duration =
S_Overtime_Duration,Staff_Overtimesalary = S_Staff_Overtimesalary,
Staff_Overtimemonth = S_Staff_Overtimemonth,  Year = S_Year, Staff_Id
= S_Staff_Id, Pro_Id = S_Pro_Id
Where Overtime_Duration = S_Overtime_Duration;
Commit; End;
```

Figure 4.53: Create Stored Procedure "updateposition_proc"

```
Create Or Replace Procedure Updateposition_Proc
(B_Id In Out Position.Position_Id%Type,
B_Desc In Out Position.Position_Description%Type,
B_Charge In Out Position.Position_Charge_Hour%Type)
As Begin Update Position Set Position_Id = B_Id,Position_Description =
B_Desc, Position_Charge_Hour  =B_Charge
Where Position_Id = B_Id; Commit;End;
```

Figure 4.54: Create Stored Procedure "updateproject_proc**"**

```
Create Or Replace Procedure Updateproject_Proc
(P_Id In Out Project.Pro_Id%Type,
P_Name In Out Project.Pro_Name%Type,
P_Start_Date In Out Project.Pro_Start_Date%Type,
P_End_Date In Out Project.Pro_End_Date%Type,
P_Duration In Out Project.Pro_Duration%Type,
P_Budget In Out Project.Pro_Budget%Type,
P_Description In Out Project.Pro_Description%Type
) As Begin Update Project Set Pro_Id = P_Id,Pro_Name = P_Name,
Pro_Start_Date = P_Start_Date,  Pro_End_Date=P_End_Date
,Pro_Duration  =P_Duration, Pro_Budget =P_Budget, Pro_Description
=P_Description
Where Pro_Id = P_Id;
Commit; End;
```

Figure 4.55: Create Stored Procedure "updatestaff_proc"

```
Create Or Replace Procedure Updatestaff_Proc
(S_Ic In Out Staff.Staff_Ic%Type,
S_Name In Out Staff.Staff_Name%Type,
S_Address In Out Staff.Staff_Address%Type,
S_Phone In Out Staff.Staff_Phone%Type,
S_Email In Out Staff.Staff_Email%Type,
S_Salary In Out Staff.Staff_Salary%Type
)As Begin Update Staff Set Staff_Ic = S_Ic,Staff_Name = S_Name,
Staff_Address = S_Address,  Staff_Phone = S_Phone ,Staff_Email  =
S_Email,Staff_Salary = S_Salary
Where Staff_Ic = S_Ic;
Commit; End;
```

Figure 4.56 :Create Stored Procedure "updatesUser_proc**"**

```
Create Or Replace Procedure Updateuser_Proc
(S_Ic In Out Staff.Staff_Ic%Type,
S_Name In Out Staff.Staff_Name%Type,
S_Address In Out Staff.Staff_Address%Type,
S_Phone In Out Staff.Staff_Phone%Type,
S_Email In Out Staff.Staff_Email%Type)
As Begin Update Staff Set Staff_Ic = S_Ic,Staff_Name = S_Name,
Staff_Address = S_Address,  Staff_Phone = S_Phone ,Staff_Email  =
S_Email Where Staff_Ic = S_Ic;  Commit; End;
```

Figure 4.57: Create Stored Procedure "deletestaff**"**

```
Create Or Replace Procedure  Deletestaff (
S_Staff_Ic In Staff.Staff_Ic%Type)
Is
Begin
Delete From Staff Where Staff_Ic=S_Staff_Ic;
End;
```

Figure 4.58: Create Stored Procedure "deletebranch**"**

```
Create Or Replace Procedure  Deletebranch (
B_Branch_Id In Branch.Branch_Id%Type)
Is Begin
Delete From Branch Where Branch_Id=B_Branch_Id;  End;
```

APPENDIX C
(User Manual)

APPENDIX C

This interface in Figure 4.59 shows the first page that will be implementing the system for all users. Besides that, customers can log in through this page also.

Figure 4.59: ora1Index.html (Home)Page

Figure 4.60 shows the Admin and staff login at the same login page

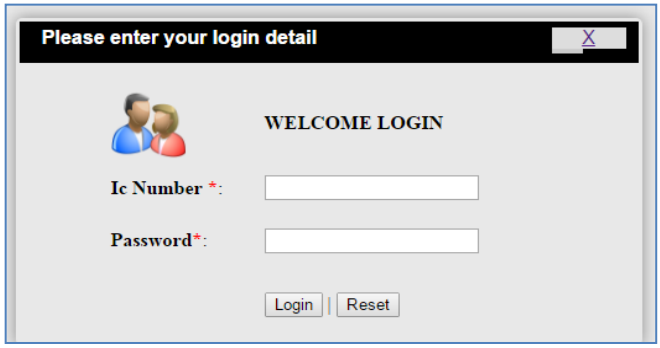


Figure 4.60 : Login Page

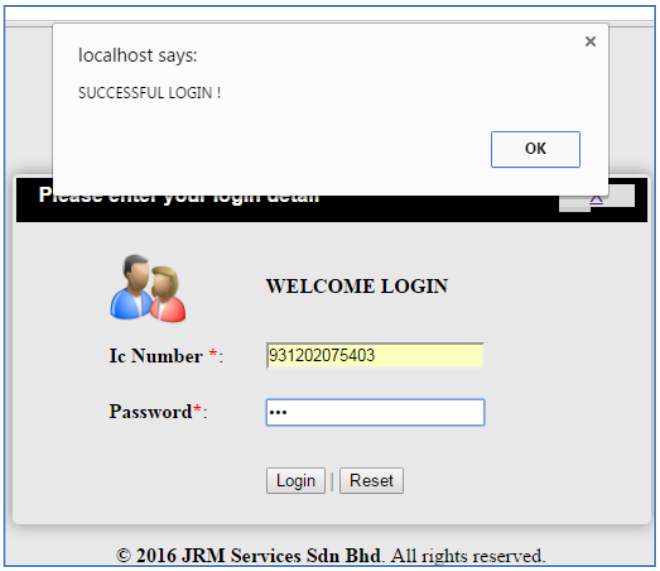


Figure 4.61: Manager/Admin Login with True Validation

This interface in Figure 4.62 shows wrong validation since the manager click 'login' button to show that identity card number and password is not valid.

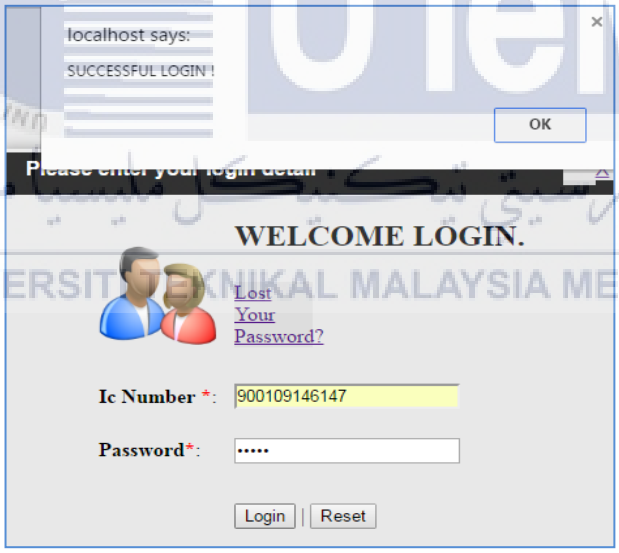


Figure 4.62: Manager/Admin Login with Wrong Validation

Figure 4.63 shows the admin home page or "oraAdmin.php" after login session is valid.

Figure 4.63: Admin Home Page

This interface in Figure 4.64 shows a list of staff that registered by the admin/manager.

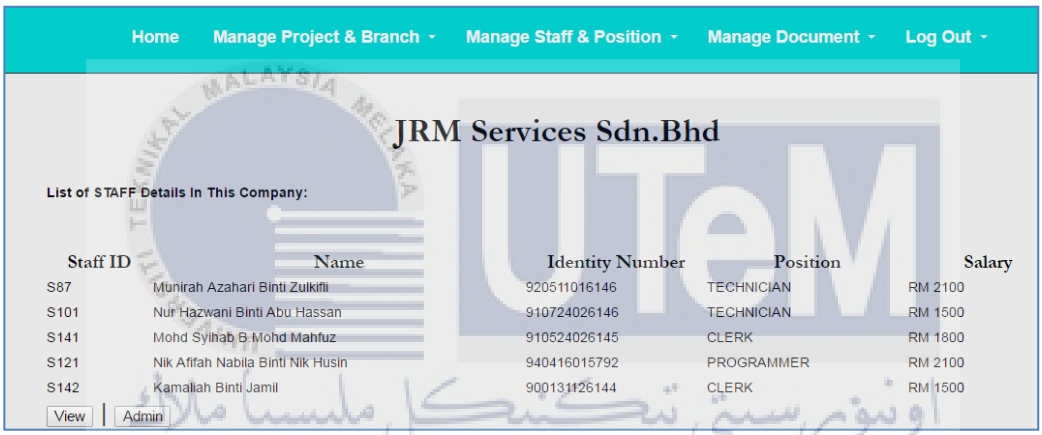


Figure 4.64: List of staff

Figure 4.65 show the details of staff registered by clicking the button 'View'.

Figure 4.65: The details of staff registered

List of Staff OverTime Claim

Overtime Duration	Overtime Claim	Overtime Month	Year	Claim By Staff Name	Job/Project Name	Action Admin
5 hours	RM172.75	MAY	2016	Wan Amir Arham Bin Yusuf	Replacement	DELETE
5 hours	RM172.75	MAY	2016	Wan Amir Arham Bin Yusuf	Repair & services	DELETE
2 hours	RM414.6	APRIL	2016	Nur Hazwani Binti Abu Hassan	Replacement	DELETE
6 hours	RM207.3	MAY	2016	Nur Hazwani Binti Abu Hassan	Replacement	DELETE

Figure 4.66 : The details of overtime claim

The Figure **4.67 ,** show the search on finding the project that have progress at branch location. User can find the project by choose the list out branch location.

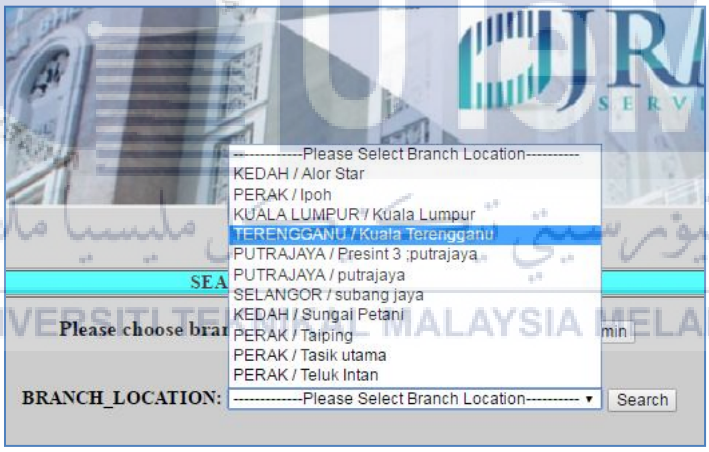


Figure 4.67 : The searching of project based on branch location

After that, user click on button ‘search’ . Then , the output come out which tell the project name that located at branch location. This shown at Figure 4.68.

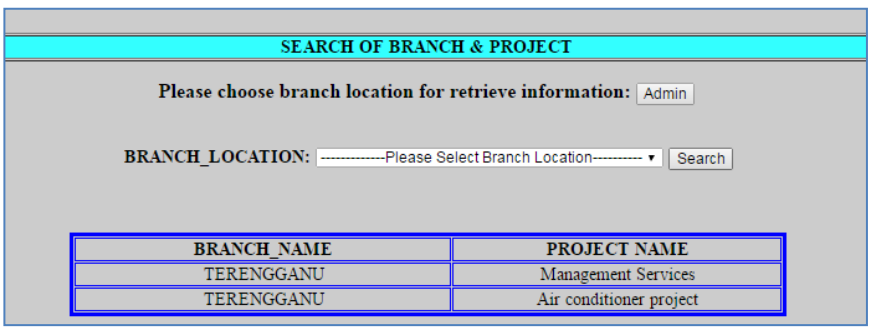


Figure 4.68: The project name based on branch location

The Figure 4.69 above show the upload form document . User must fill in the details and uploaded the document image and sbumit by click the button “Add New Document”.

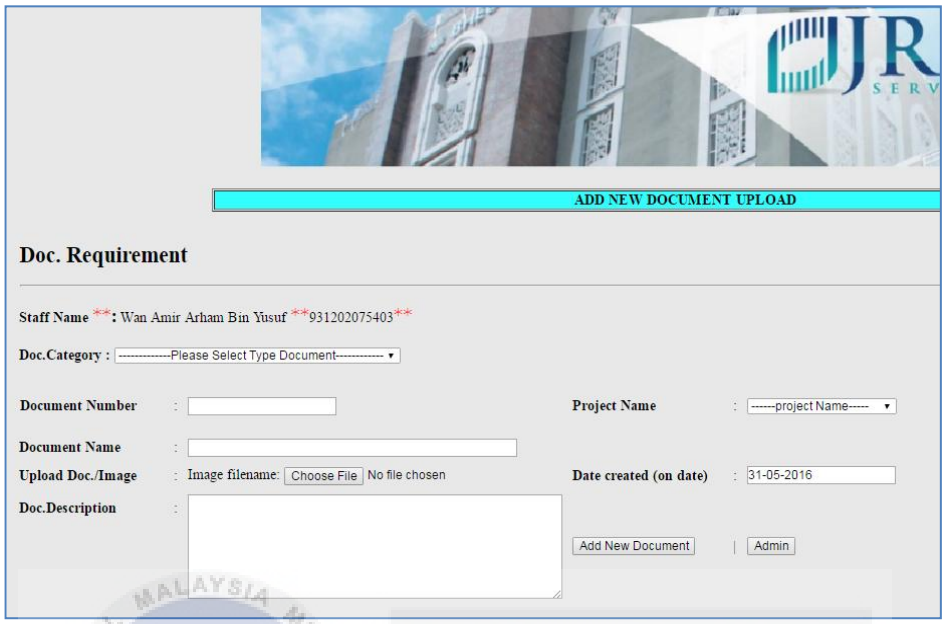


Figure 4.69: The upload form document

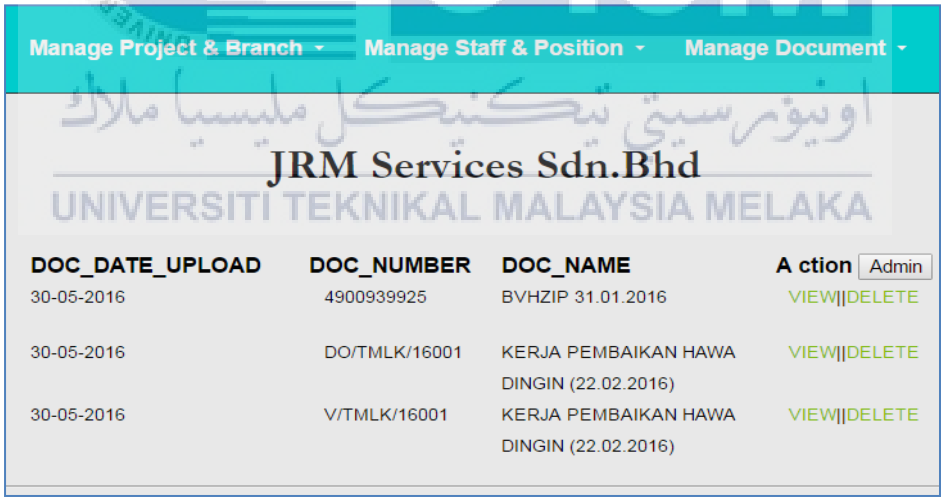
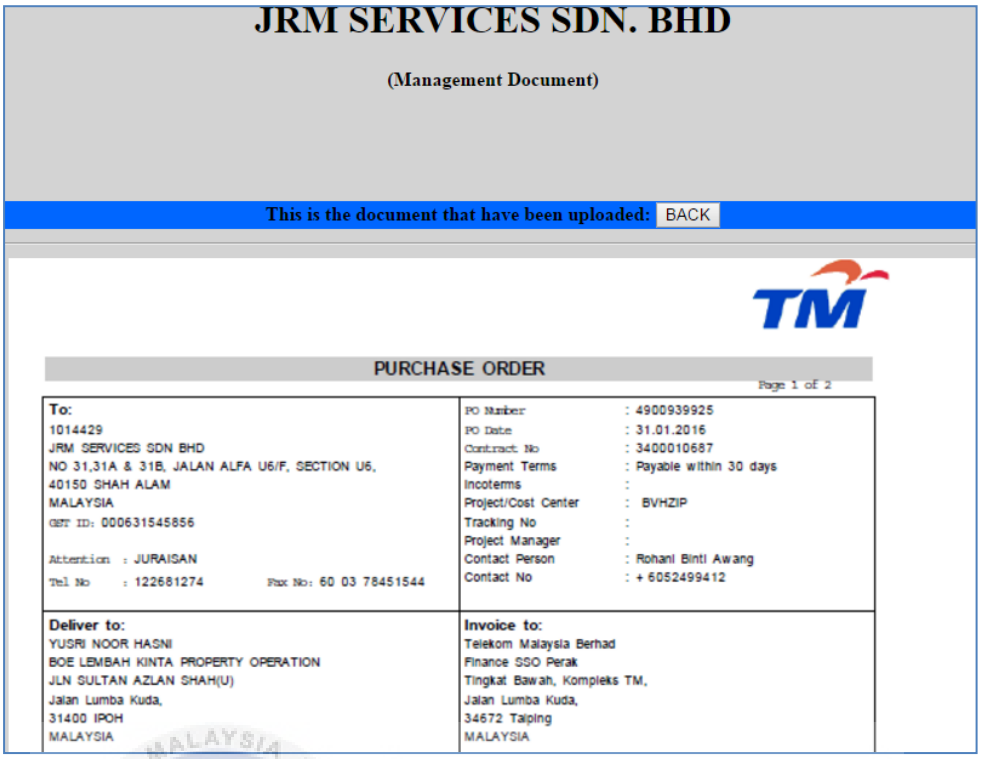


Figure 4.70: The listed upload document



JRM SERVICES SDN. BHD

(Management Document)

This is the document that have been uploaded: BACK

PURCHASE ORDER

Page 1 of 2

To: 1014429 JRM SERVICES SDN BHD NO 31,31A & 31B, JALAN ALFA U6/F, SECTION U6, 40150 SHAH ALAM MALAYSIA GST ID: 000631545856 Attention : JURAISAN Tel No : 122681274 Fax No: 60 03 78451544	PO Number : 4900939925 PO Date : 31.01.2016 Contract No : 3400010687 Payment Terms : Payable within 30 days Incoterms : Project/Cost Center : BVHZIP Tracking No : Project Manager : Contact Person : Rohani Binti Awang Contact No : + 6052499412
Deliver to: YUSRI NOOR HASNI BOE LEMBAH KINTA PROPERTY OPERATION JLN SULTAN AZLAN SHAH(U) Jalan Lumba Kuda, 31400 IPOH MALAYSIA	**Invoice to:** Telekom Malaysia Berhad Finance SSO Perak Tingkat Bawah, Kompleks TM, Jalan Lumba Kuda, 34672 Taiping MALAYSIA

Figure 4.71: The View upload document

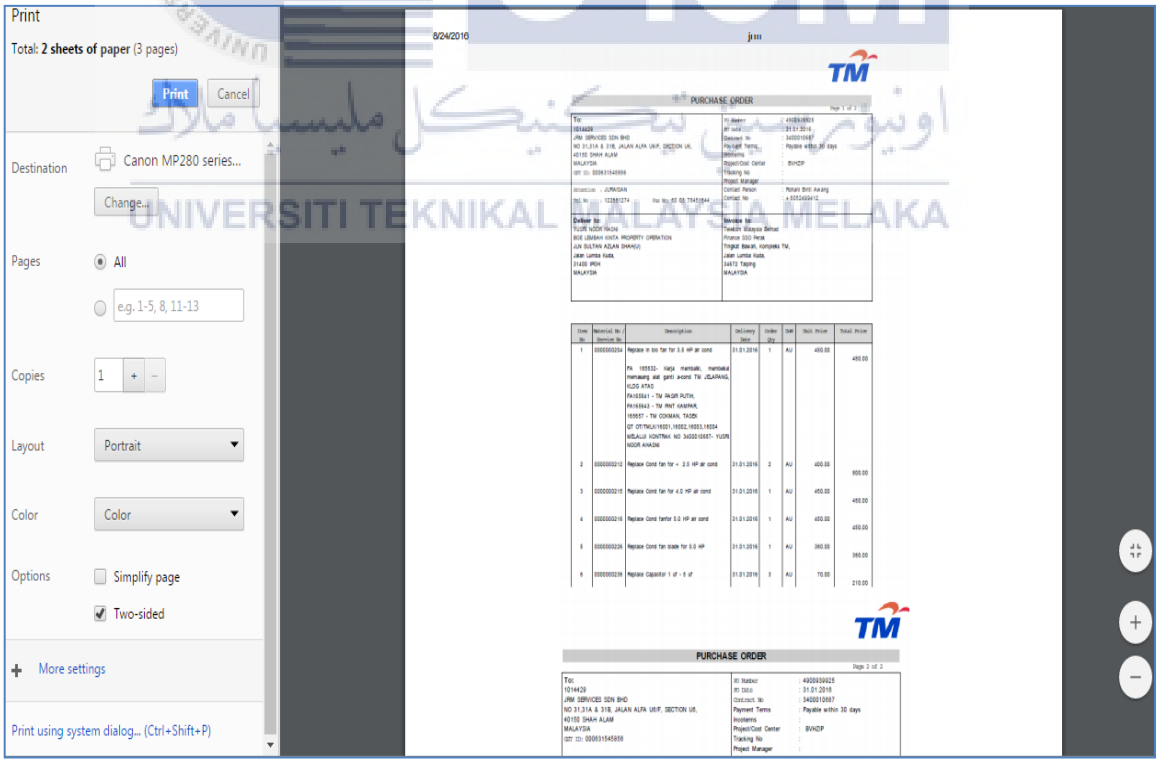


Figure 4.72: Print View of Document

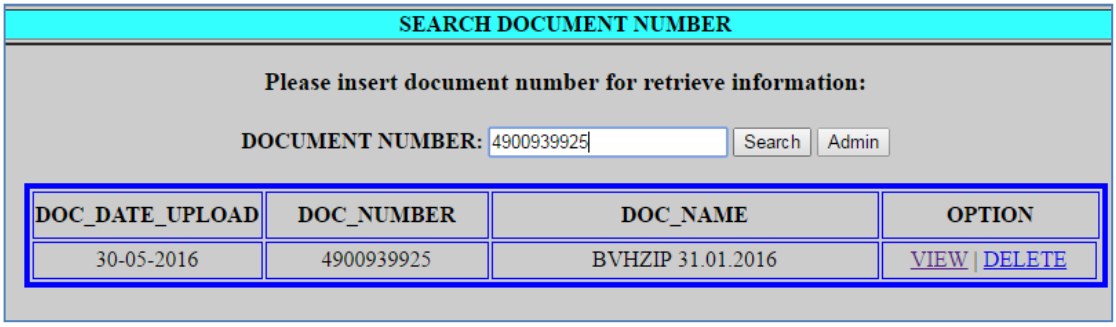


Figure 4.73: The search of document number

Figure 4.74: The details after click views the document

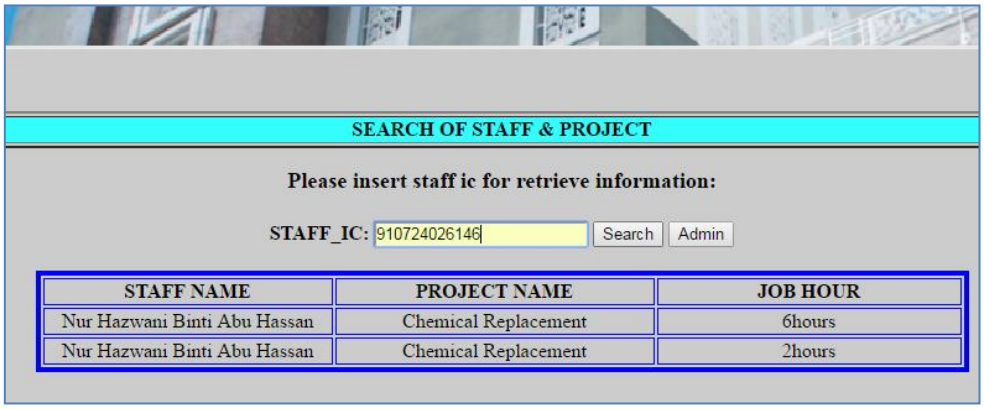


Figure 4.75 : Trace (search) project supervisor / staff who handle the project

4.4.1.8 Security Account .

```
</script>
<script type="text/javascript">
idleMax = 1;// Logout after 1 minutes of IDLE
idleTime = 0;
$(document).ready(function () {
    var idleInterval = setInterval("timerIncrement()", 60000);
    $(this).mousemove(function (e) {idleTime = 0;});
    $(this).keypress(function (e) {idleTime = 0;});
})
function timerIncrement() {
    idleTime = idleTime + 1;
    if (idleTime > idleMax) {
        alert('You have been logged out.You will now be redirected to home page.');
                document.location.href = "oralindex.html";
    }
}
</script>
```

Figure 4.76: Coding Security Logout automatic after 1 minutes login

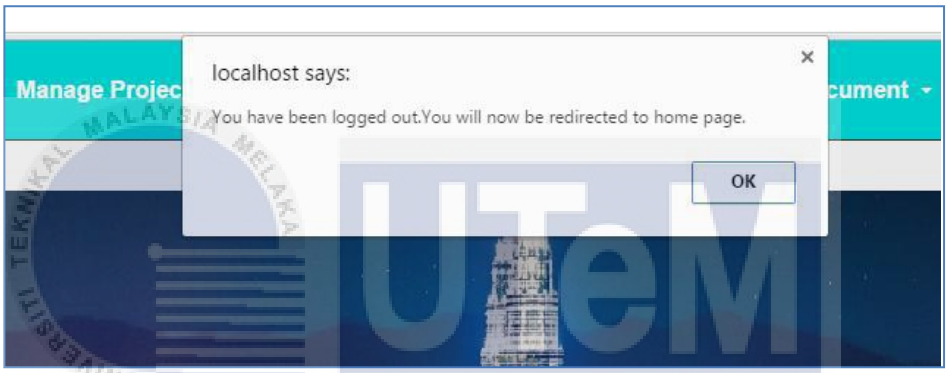


Figure 4.77: Logout automatic if account user not managed by the user

Admin can register staff during business hour which is from 8:00 a.m. until 6:00p.m. Figure 4.78 are shown on below.

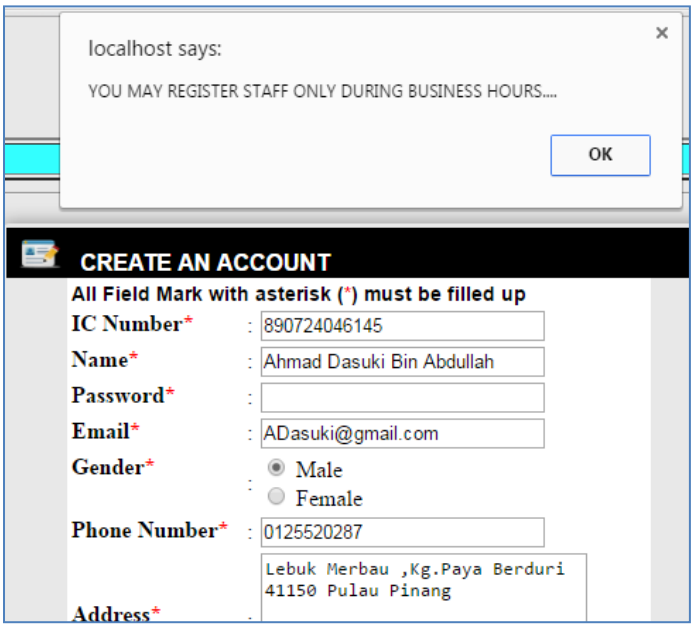


Figure 4.78: Validation to show that admin can only register during business hour.

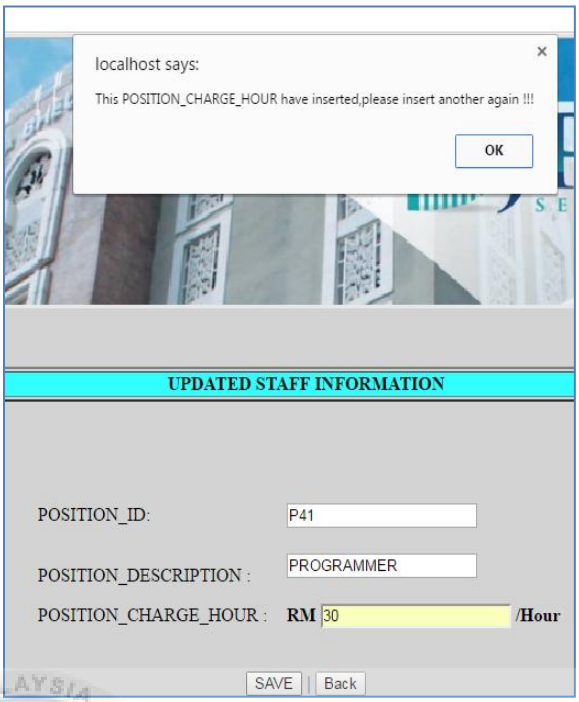


Figure 4.79: Validation Avoid Duplicate & Same Position Charge hour (RM/Hour)

Figure 4.80 below show the process staff for claim overtime salary . Staff must insert their overtime details for the claim. If they did not follow,they cannot get their claim overtime based on month and year claim.

For future; this part may include the biometric sensor thumb screen to make sure the staff come and dismiss their work from the office.

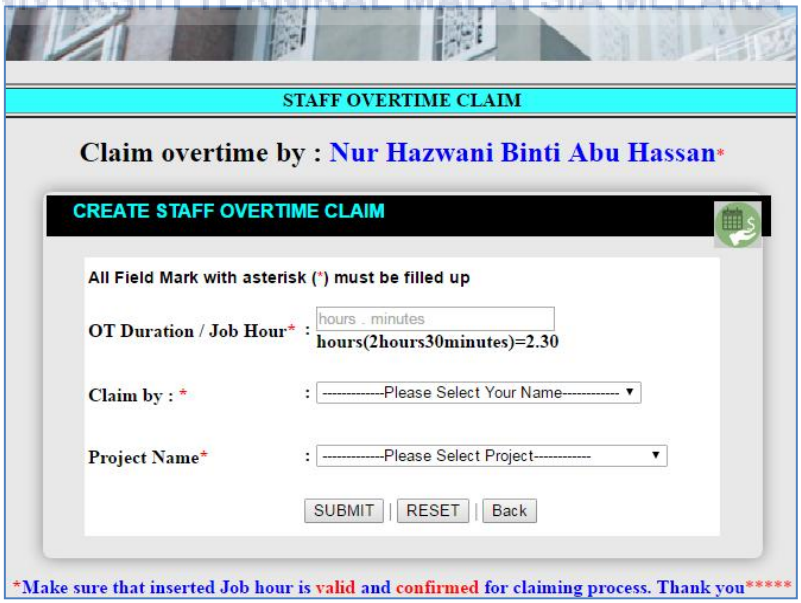


Figure 4.80: Staff Overtime Claim Form Calculation

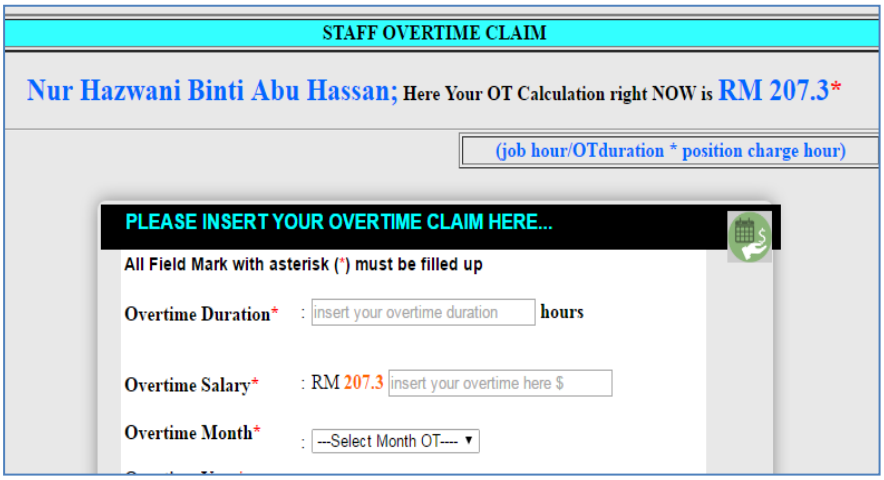


Figure 4.81: Staff Overtime Claim Registration Calculation

Figure 4.82: List of staff overtime claim from admin view

Figure 4.83 and Figure 4.84 shows the overtime checking by admin which is the manager/admin can check the overtime claim by staff .Manager searches the chosen name for view the overtime claim details

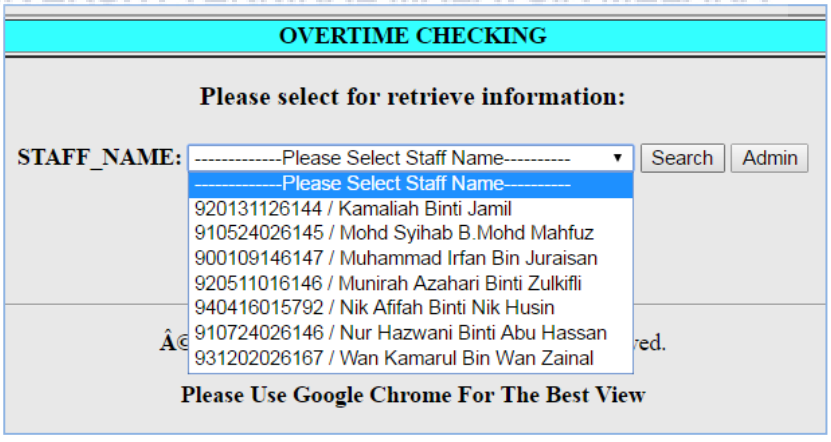


Figure 4.83: Overtime Checking

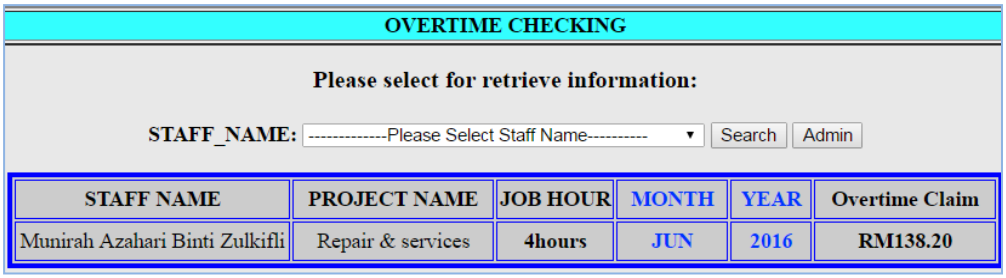


Figure 4.84: Overtime Checking Result

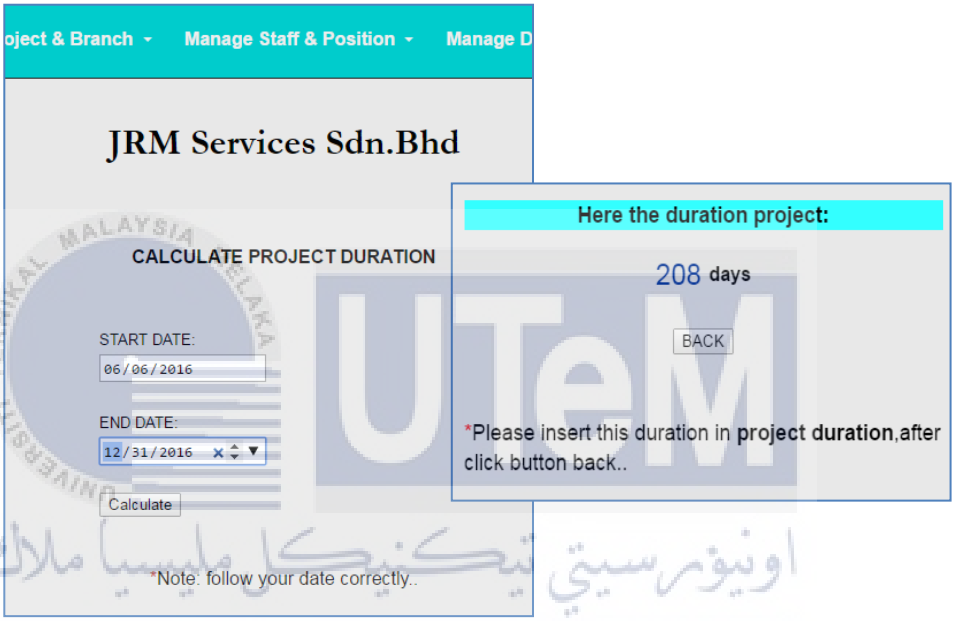
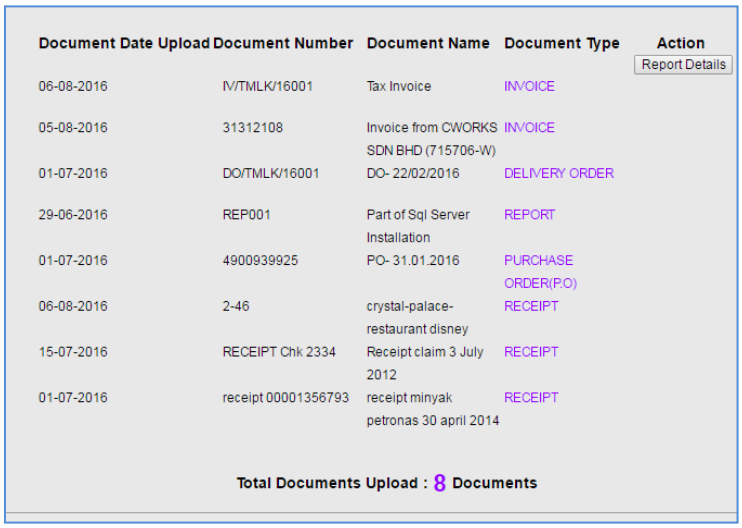


Figure 4.85 & Figure 4.86: Calculation Project Duration

This Figure 4.87 and 4.88 shows the report of document uploaded and the report total document uploaded.



Document Date Upload	Document Number	Document Name	Document Type	Action
				Report Details
06-08-2016	IV/TMLK/16001	Tax Invoice	INVOICE	
05-08-2016	31312108	Invoice from CWORKS SDN BHD (715706-W)	INVOICE	
01-07-2016	DO/TMLK/16001	DO- 22/02/2016	DELIVERY ORDER	
29-06-2016	REP001	Part of Sql Server Installation	REPORT	
01-07-2016	4900939925	PO- 31.01.2016	PURCHASE ORDER(P.O)	
06-08-2016	2-46	crystal-palace-restaurant disney	RECEIPT	
15-07-2016	RECEIPT Chk 2334	Receipt claim 3 July 2012	RECEIPT	
01-07-2016	receipt 00001356793	receipt minyak petronas 30 april 2014	RECEIPT	

Total Documents Upload : 8 Documents

Figure 4.87: List Document Upload Report

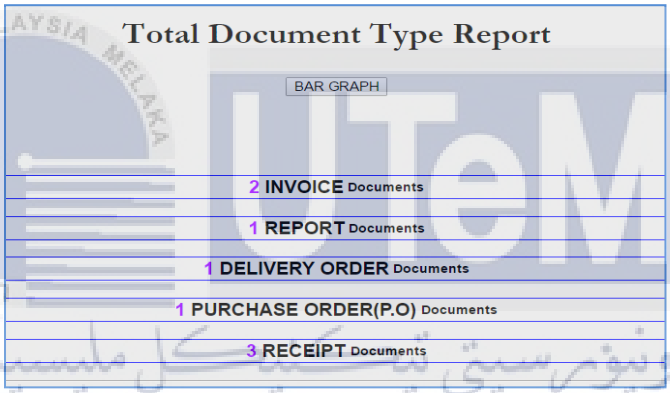


Figure 4.88 : Total Document Upload Report Details

Figure 4.89 and Figure 4.90 show bar graph report and view print bar graph report.

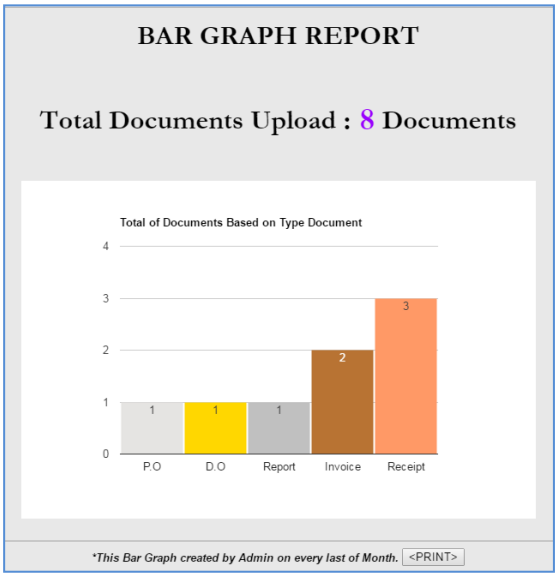


Figure4.89 :BarGraphReport

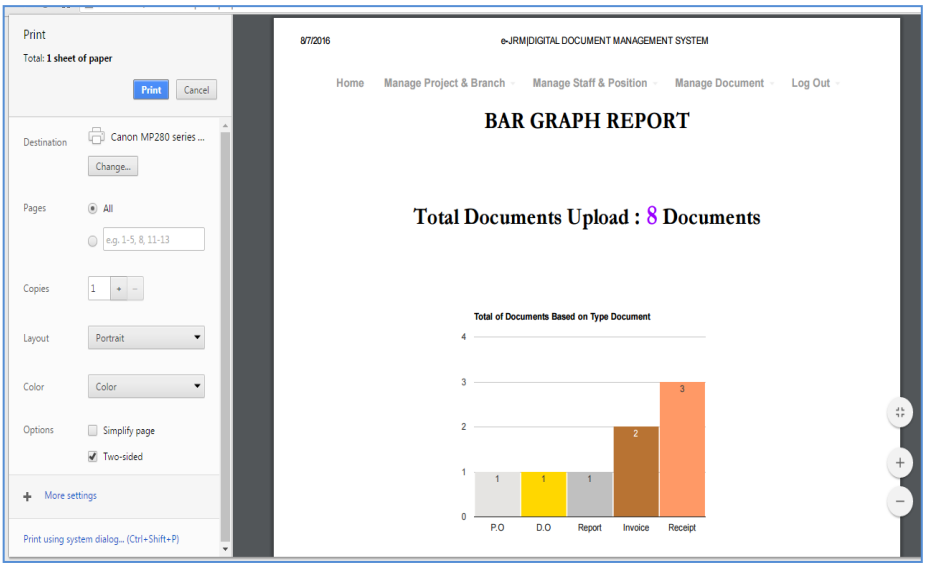


Figure 4.90 : Print View Bar Graph Report

Figure 4.91 show the pop-up came out when user click at "Lost Your Password?" tells staff to contact admin for resetting their password.

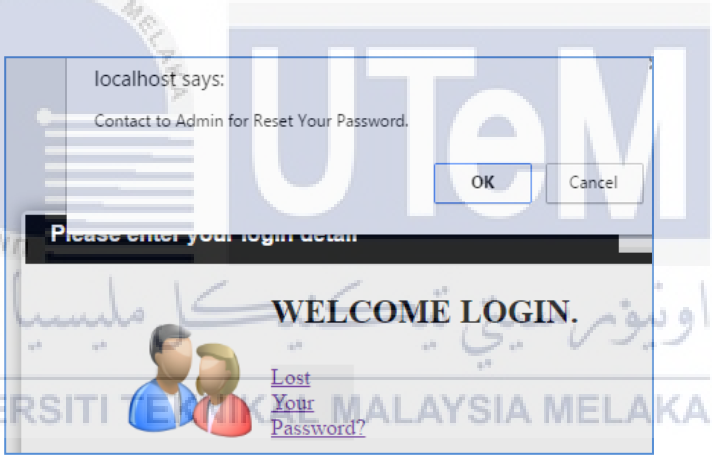


Figure 4.91: Reset Password Validation

Figure 4.92 shows the checking of identity card for making sure that the user that want to reset password is approved as staff JRM

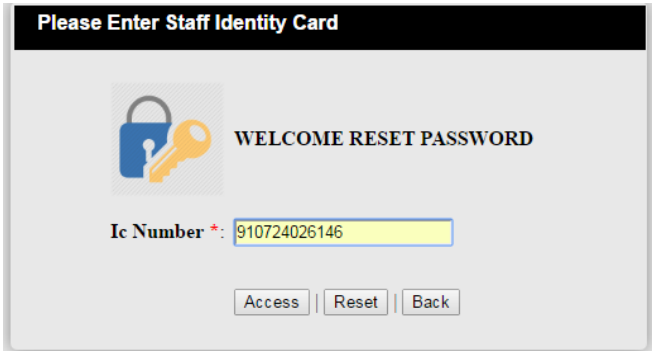


Figure 4.92: Checking Staff Identity Card for Reset Password

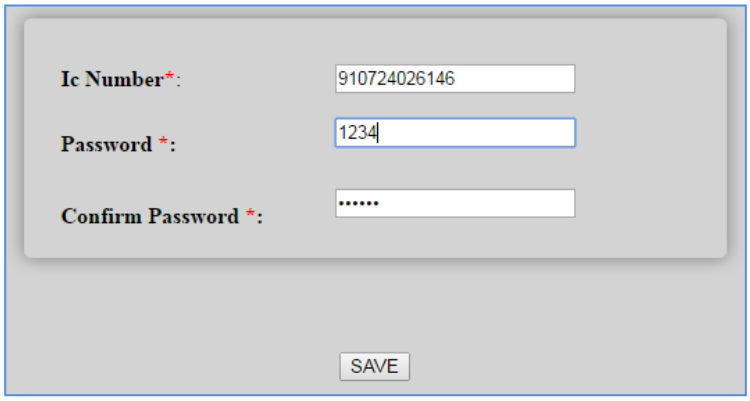


Figure 4.93: Admin resets the Password

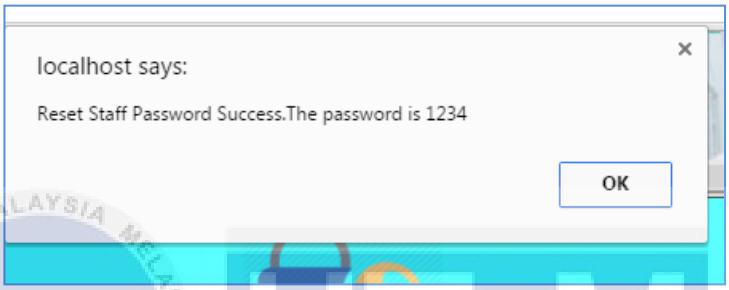


Figure 4.94: Reset Password Success Validation

Figure 4.95.show the maps and location of JRM Services Sdn.Bhd. Company based on the satellite view. There are location address, contact number and operation day of the company. The user can zoom into the map location and see the real strategic location of JRM Services Sdn. Bhd. Company.

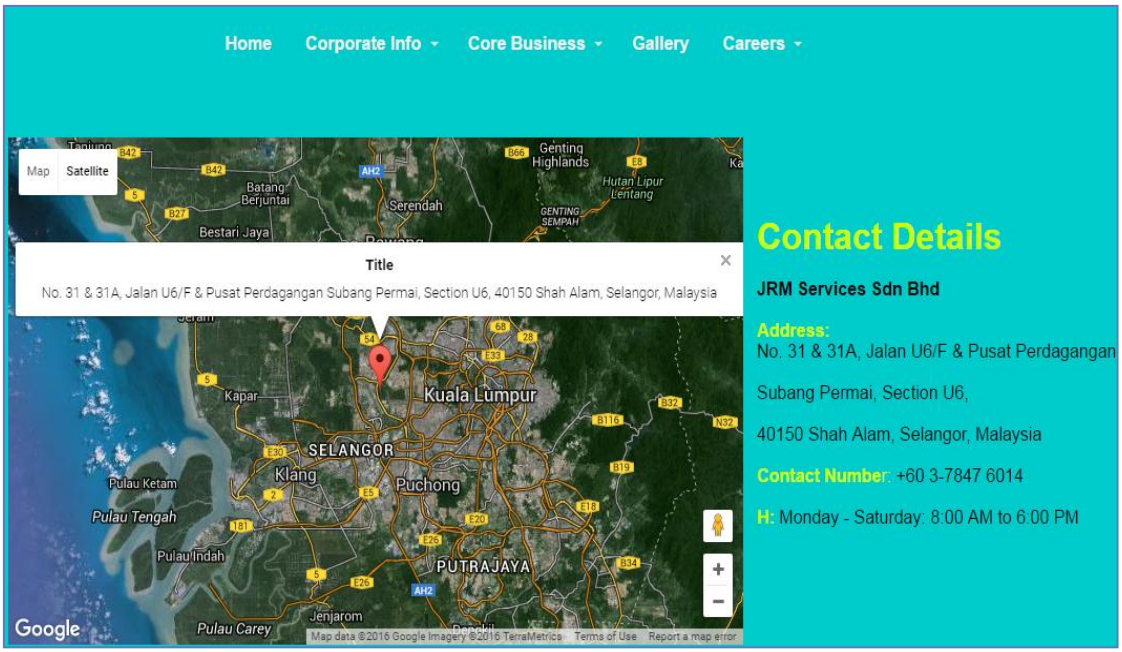


Figure 4.95: Company Location from Satellite View

APPENDIX D

(Turnitin)

APPENDIX D

The percentages similar document by using Turnitin

The digital receipt that Turnitin received report paper.