i-BANK QUESTIONS (iBQ): QUESTION GENERATOR

MUHAMAD SHAHMI BIN ZAMZURY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL : i-Bank Questions (iBQ) : Question Generator

SESI PENGAJIAN : 2015/2016

Saya MUHAMAD SHAHMI BIN ZAMZURY

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaaan 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 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: 29 Hulu Sek. Keb. Mentuan,
Jalan Sultan Omar, 16150
Kota Bharu, Kelantan.
Tarikh: 25.8.16

(TANDATANGAN PENYELIA)
(En. Yahya Bin Ibrahim)
Tarikh: 25/8/2016

CATATAN: * Tesis dimaksudkan sebagai Laporan Projek Sarjana Muda (PSM)
**Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

i-BANK QUESTIONS (iBQ) :
QUESTION GENERATOR

MUHAMAD SHAHMI BIN ZAMZURY

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

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2016

DECLARATION

I hereby declare that this project report entitled

I-BANK QUESTIONS (iBQ):

QUESTION GENERATOR

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT: _____ Date: 25.8.16

(MUHAMAD SHAHMI BIN ZAMZURY)

I hereby 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

(SIR YAHYA BIN IBRAHIM)

DEDICATION

To my father En Zamzury Bin Hassan and mother Pn Mariam Binti Yusoff

To my beloved supervisor, En. Yahya Bin Ibrahim

And to all my friends of BITD, who always help me to complete this project.

Thank you so much.

.

ACKNOWLEDGEMENT

Grateful to Allah S.W.T with His gift of His grace, we can also complete this final project of i-Bank Questions (Question Generator) successfully without any problems.

We also wish to express our appreciation and gratitude to the supervisor of this project, En Yahya Bin Ibrahim for their guidance and encouragement given during the period of project implementation.

Thank you also to all the lecturers and colleagues for your help. My gratitude also goes to all the parties involved either directly or indirectly in the success of this project.

Thank you.

ABSTRACT

i-Bank Questions (Question Generator) is a system that provides a platform to teacher and student to download question. The system can download any question base on subject, form, chapter, topic, category and level. Through this system, the teacher can download question with answers scheme. Methodology Software Development Life Cycle (SDLC) used in the development of this project is Agile method. The software involved in the development of this system is Eclipse Java – EE, Apache Tomcat Server, Sublime text editor, Adobe Photoshop CS3 and Oracle in its package, Windows 8 and Google Chrome as web browser.

ABSTRAK

i-Bank Questions (Question Generator) adalah sebuah sistem untuk guru dan pelajar untuk memuat turun soalan. Sistem ini boleh memuat turun pelbagai soalan mengikut subjek, tingkatan, bab, topik, kategori dan tahap kesukaran. Melalui sistem ini, guru boleh memuat turun soalan dengan jawapan skema. *Methodology Software Development Life Cycle (SDLC)* yang digunakan dalam pembangunan projek ini adalah kaedah *Agile*. Perisian yang terlibat dalam pembangunan sistem ini adalah *Eclipse Java – EE, Apache Tomcat Server, Sublime text editor, Adobe Photoshop CS3* dan *Oracle* sebagai sistem pengkalan data, *Windows 8* dan *Google Chrome* sebagai pelayar web.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CPU	-	**Central Processing Unit**
DBMS	-	**Database Management System**
DFD	-	**Data Flow Diagram**
ERD	-	**Entity Relationship Diagram**
GUI	-	**Graphic User Interface**
RAM	-	**Random Access Memory**
SDLC	-	**System Development Life Cycle**
UTeM	-	**Universiti Teknikal Malaysia Melaka**

CHAPTER I

INTRODUCTION

1.1 Project Background

i-Bank Questions (Question Generator) is developed to provide a platform for teacher and non-teacher to generate and download the question paper from the worldwide base on the year, subject, topic and level. It is such a waste if a good question or a good question maker are not shared or forward themselves to the worldwide. With their specialty in making a good question, all the question from any teacher are available to download for any teacher or student to make this system more interesting, each of the downloaded questions will be reward to the owner based on the most number of question downloaded and the owner of that question will be list in a special place, likes hall-of-fame, so that whole user will recognize the famous question or the most downloaded question. Besides, this system also provides search utility to ease the users on searching the specific question for download.

1.2 Problem statement

I. There is no platform for teacher and non-teacher to generate and download the question paper from the worldwide.
   - Teacher difficult to find the question with the scheme. Without using a system, they need to find manually using a book to retype and combined the question for related topic.

II. It is hard and takes a time to find and generate the specific question to download.
   - Teacher hard to find specific question base on subject, form, chapter, category and level.

III. Takes time to know what kind of question are the most downloaded and which teacher is the most download.
   - Teacher difficult to know who is the highest downloaded question or examination and to know the ranking of the teacher.

1.3 Objective

I. To develop a platform for the teacher to generate and download objective question or subjective question paper together with answer scheme.
   - Teacher can easily find and download any question in the system which is the question from other teachers that shared it. The answers scheme provided to the teacher for marking their student's examination or test.

II. To develop a platform for the teacher to generate question base on form, category, subject, topic and level.
   - Teacher can generate question base on subject, form, category, topic and level and can download or print the

question that already sort with examination format. Teachers no need to edit or sort the question and their student can easily answer the question.

III. To investigate which teacher are the most download and automatically rewarding them.

- Teacher can view the ranking for the most download so it can make other teacher compete to get the reward.

1.4 Scope

There are two major scopes involved in the development of the i-Bank Questions (Question Generator) which are the user and module scope. User scope focuses on the system's user and their roles while module scope determines the details of function that is categorized by each module.

1.4.1 User Scope

There is two users in i-Bank Questions (Question Generator) system which is teacher and student. Teacher can find any question base on subject, form, chapter, topic and level and can download or print the question with the answers scheme. Besides that, teacher can view the ranking of the most downloaded question and teacher can know which teacher is the most download. On the other hand, Student also can download any question for their exercise.

1.4.2 Modules Scope

I. Objective Question Module

The objective question module will generate the question and multiple choice answers and sort the question and answer with formal examination paper to easily print and answer directly. The answers scheme provided for teacher marking.

II. Subjective question Module

Subjective question module will generate question and the space for student answer it automatically provided below the question. The answers scheme for teacher marking is provided together with possible answers and marks so the teacher can mark the test faster.

III. Ranking and History Module

Every question downloaded will be recorded and the detail of the examination or test and the question will display in history for every teacher. The teacher can view the detail of the examination or test and can view the ranking for all teachers which are most download.

1.5 Project significance

i. Reduces Manual Processes

In order to make the examination or test, the teacher needs to find one by one question for the specific chapter or topic. The system makes teacher generate automatically any subject, topic or chapter.

ii. Improve Data Storage Techniques

This system improves data storage by using a systematic and efficient system. For example, the question in the system will be increasing from other teacher uploaded it.

iii. Save Time

This system will generate any question with the answers scheme. Teachers no need to find the question and retype the scheme and sort them manually.

iv. System Management More Efficient and Effective

By using this system, teacher easily to find the question for their student without using any revision book or manually create the question. The management of the examination will improve and more systematic.

1.6 Expected Output

I. Teacher can generate any question

Teacher and student can search objective and subjective question by subject, form, chapter, topic and level. Teacher no longer has to find manually from the book and retype into examination format question.

II. Teacher can download question with answers scheme

Teacher can download and print question with answer scheme that cans easily marking answers of their student.

III. Teacher can view their history and ranking of download

Teacher can view detail of their download history. They also can view the ranking for most download for all teachers. They also can easily search their ranking number.

1.7 Conclusion

i-Bank Questions (Question Generator) is an efficient system to assist teacher generate question for their examination or test. This system will save time for teacher to find the question without using any reference book because the system contains a lot of question from other teacher. The marking for the examination will become easy with the complete answer scheme provided together with possible answers and mark for each answers.

CHAPTER II

LITERATURE VIEW AND PROJECT METHODOLOGY

2.1 Introduction

In this chapter, the literature section review and past research regarding to teacher finding the question and answer to their student. The aim of this review is to find the solution and approach in developing the i-Bank Questions (Question Generator). This will be done by conducting personal communication from other experts. From the literature view, the significant evolution of a project over time will be described. Methods are used during develop this system is very important to estimate the time of the system to be delivered on the stage. Here, choose agile method because it is more flexible than waterfall.

2.2 Project methodology

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Agile methodology approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "Agile methodology" approach, the whole process of software development is divided into separate phases. Agile Methods break the product into small incremental builds. These builds are provided in iterations.

2.2.1 Database Planning

I. The user of this system is identified. Teacher and student are the main users in the i-Bank Questions (Question Generator). Teacher can manage data of the question such as question detail and examination download details. Student only can view the detail of the question or download the question for exercise.

II. It has 3 modules in i-Bank Questions (Question Generator) which are objective question, subjective question and examination or download details.

2.2.2 Requirements Collection and Analysis

I. The requirement of the system was collected and analyzed to make sure it supported by the database system. It will help to define the required data structures, the type and numbers of entities, the physical size of the database and so on.

II. Important information was collected from expert teacher from many schools and was analyze to identify the user requirement.

2.2.3 Database Design

I. The conceptual design of the i-Bank Questions (Question Generator) is made using Entity Relationship Diagram (ERD). This data modeling will be used to create the abstract database structure to be easier to understand.

II. The selection of the DBMS software is important to the information's system for a smooth operation. The end users also must be always aware of both DBMS and the database. After that, the logical design is develop by using Data Dictionary and Data Normalization. The physical design is then develop when Data scheme is produced. It can be define as a process of select the data storage and data access characteristics of the chosen database.

2.2.4 Implementation

The database i-Bank Questions (Question Generator) is built by using the Data Definition Language (DDL), Data Manipulation Language (DML) and Data Control Language (DCL). In modern relational DBMS, a new database implementation requires the creation of special storage-related constructs to address the end-user tables.

2.2.5 Testing

After the data was inserted into the database, testing and evaluation phase occurs that will test the database for performance, integrity, and concurrent access and security constraints. Some options will be considered to enhance the system if the database implementation fails to meet the system's evaluation criteria or requirement.

2.2.6 Operation Maintenance

All problems that could not predict during the testing phase can be detected after all the targeted end-users entered the operation phase. The database administrator must be prepared to perform routine maintenance activities within the database. Some of the required periodic maintenance activities included backup, recovery and maintain the database

2.3 Project Schedule and Milestones

The task planning is shown in table 2.1:

Table 2.1: Task Planning

Week	Date	Phase	Results
1 - 2	**23 Feb – 7 March**	**Planning** - Submit the proposal to PSM committee. - Collect and research information - Analysis the information - Choose the project methodology - Build the GANTT Chart.	- Student Demerit System proposal - Report of Chapter 1: INTRODUCTION - Report of Chapter 2: PROJECT METHODOLOGY AND PLANNING
3 - 4	**9 March – 21 March**	**Analysis** - Investigate current system problem. - List down the business requirement - Analyze the requirement of the system. - Create flow chart and DFD.	- Appropriate diagram such as flow chart and DFD
5 – 6	**23 March – 4 April**	**Design** - Design the system Architecture using diagram. - Create user interface design	- Appropriate diagram such as ERD and the design form

		such as navigation design and Input/output design.	
7 – 11	**6 April – 9 May**	**Implementation** - Implement the database -Implement system modules -Implement user Interface	- Appropriate interface such as user login, search page, record, etc.
12 – 14	**11 July – 30 July**	**Testing** - Data testing such as define the hardware and software that been used. - Case testing - Prepare testing report.	- Appropriate testing such as user testing to test the result either Ok or failed.

2.4 Conclusion

As the conclusion, Project methodology is very important for developers to ensure development will be run as plan. It describes the history of the database within the information system. The DBLC is composed of six major phases; Database Planning, Requirements Collection and Analysis, Database Design, Implementation, Testing and Operation Maintenance. This chapter is covered on introduction of the chapter, domain for the system, project methodology and project schedule and milestone.

CHAPTER III

ANALYSIS

3.1 Introduction

System analysis is one of the important phases in methodology to identifying the problems of the system, the technical feasibility and implementation. Analysis defines the requirements of the system depends on how the requirements will be accomplished. It is an activity that is focused on the problem domain that needs to be solved. Analysis phase is to understand the user's requirements and the problem domain. Besides that, analysis is also to show and describe the actions of the system and to develop the best system in order to meet the needs of consumers. The problems that occur should be documented and prioritized first to be solved during development phase of the system. There are few analysis techniques to identify the system requirements and understanding the problem domain which is reading or research, knowledge acquisition and logical model of the new system. The process of this analysis is requirements modeling such as functional requirement, non-functional requirements and other requirements such as hardware and software requirements that will be discuss in more detail in this chapter.

3.2 Problem analysis

To develop and implement a system successfully, it is important to follow the procedures. Objective of the project is to ensure that teacher can generate question using this system. The current system that use by teacher is cannot generate specific question. It just comes with a set of question with all the chapter or topic. Teacher need to arrange the question manually for the specific topic or chapter. If teacher need a specific topic, they need to find manually from a bundle of question. The current system also only have question from provided only. Teacher cannot add their own question or share their question to others. Student cannot use the system because it provided of teacher only, so student hard to find the question for their exercise.

3.3 The proposed improvements/solutions

The current system that teacher use to find the question is only past year and their provide question only. Teacher cannot make their own question and share their question to others.

Figure 3.1: Sasbadi question bank system

Figure 3.2: Page for choose topic

The current system only can choose the topic without specific question for example chapter, topic, category and level. Their system also only for the teacher, so student cannot use the system.

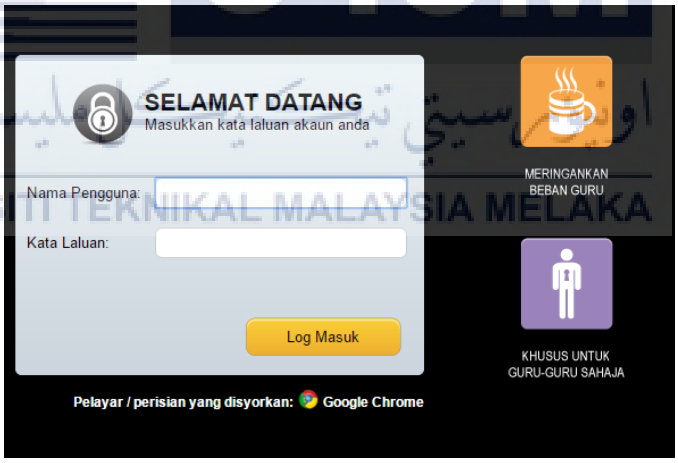


Figure 3.3: Login page

System to be develops to replace the current. This system wills easier the teacher and also student to generate the question for the base on their subject, chapter, category, level and topic. So that teacher can share their question for all students.

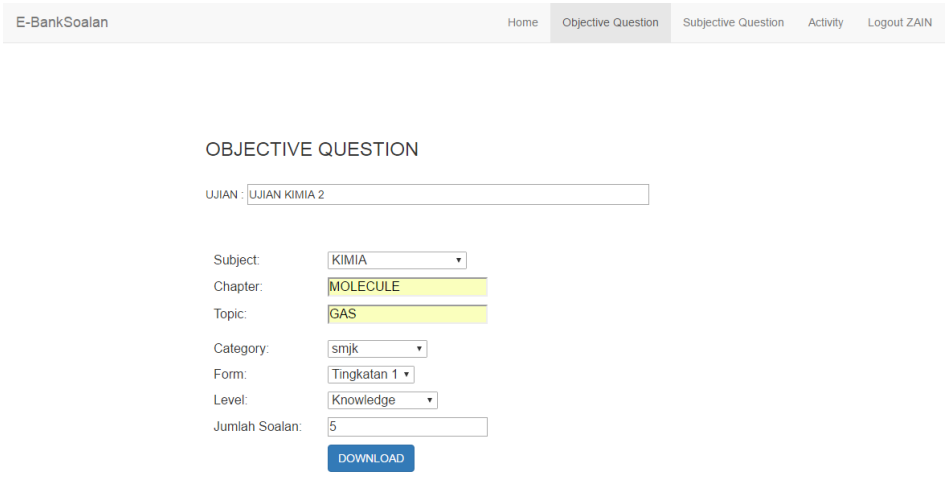


Figure 3.4 System to be develop

3.4 Requirement analysis of the to-be system

The system is for teacher to download the question and answers scheme and data was recorded in the system and student download the question without answer scheme but not recorded in the system. System needs the teacher to manage the question and give the answer scheme to their student so that is no problem for other teacher use their question to check the answer scheme. Requirement analysis is a necessary requirement in software, hardware and other aspect.

3.4.1 Functional Requirement (Process Model)

This i-Bank Questions (Question Generator) provides the information about the question and answers for the teacher and student download the question. This system about question bank for teacher stores their question and shares it with other teachers and student. Once teacher download a set of question, all the data was recorded or store into the database. Teacher can view all their download history or delete it. This system also provides the question for student. Student can download question but without the answers scheme. This system will be develop using Java EE-Eclipse, and sublime text. This software is used to design the system graphical user interface during implementation phase.

Flow Chart for the Proposed System

Figure above shows the Flow Chart and Data Flow Diagram (DFD) how the system would interact with teacher and student. Teacher need to login into the system to download the question with answers scheme and student no need to login to download the question but without answer scheme.

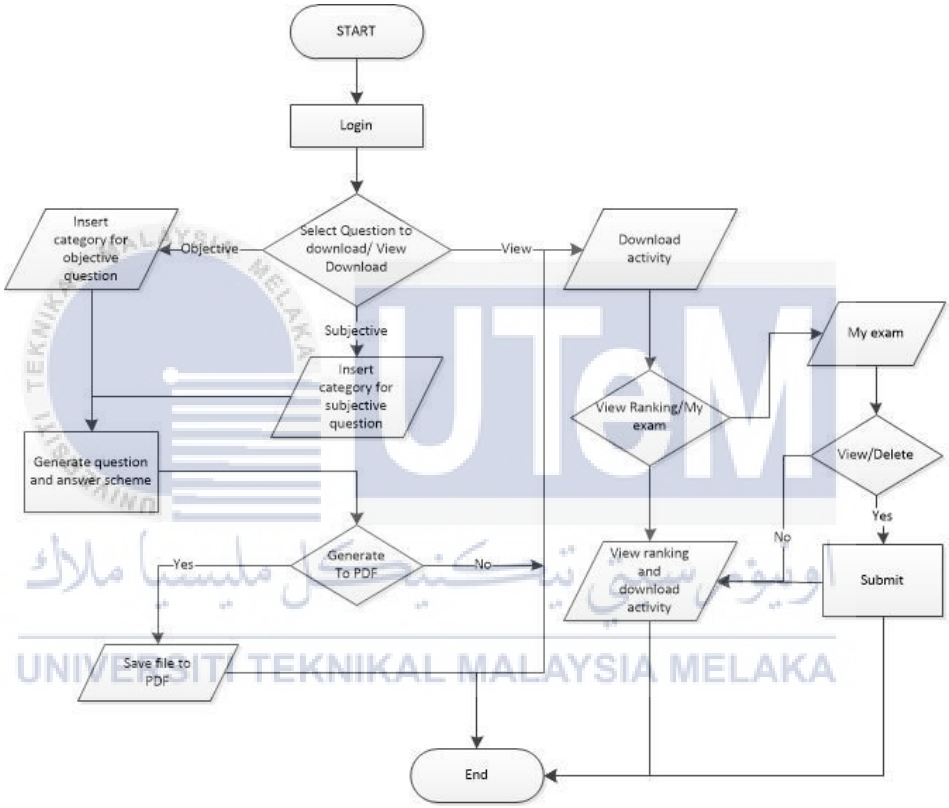


Figure 3.5: Flow chart for teacher

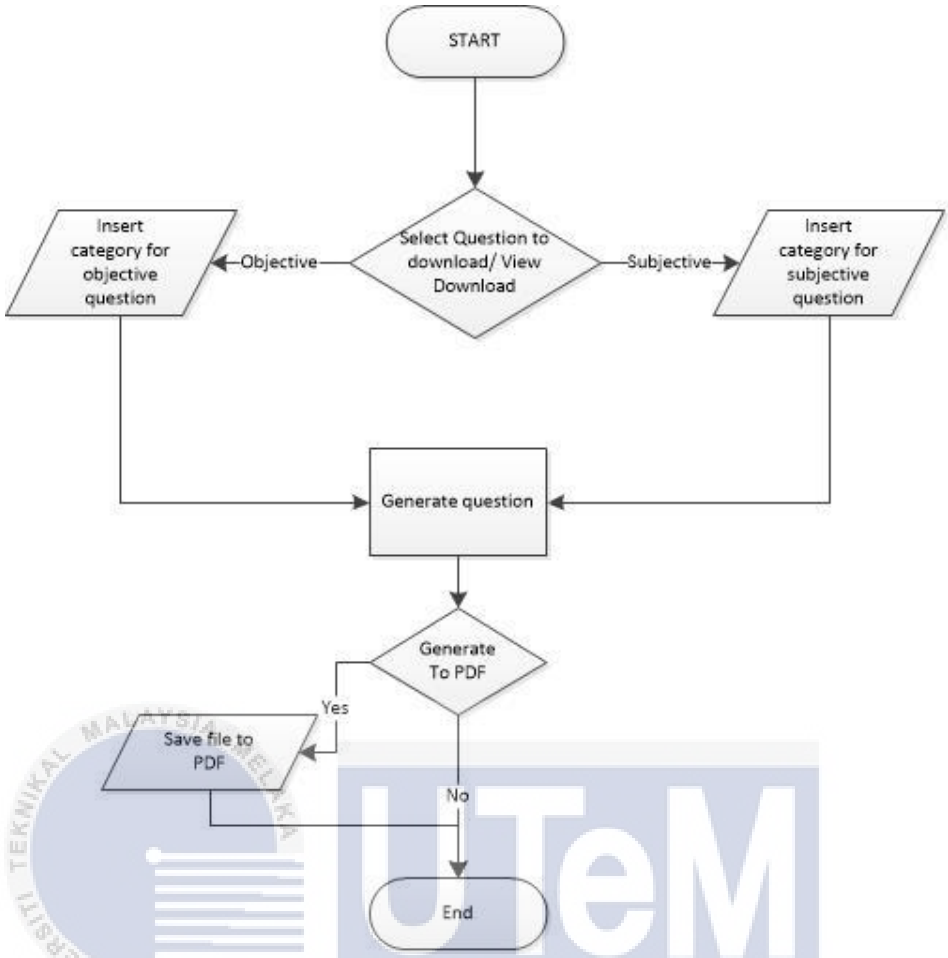


Figure 3.6: Flow chart for student

Context Diagram (DFD) of System to be developed

Figure 3.7 shows the context diagram that shows how the system boundaries, external entities that interact with the system, and the major information flows between the entities and the system.

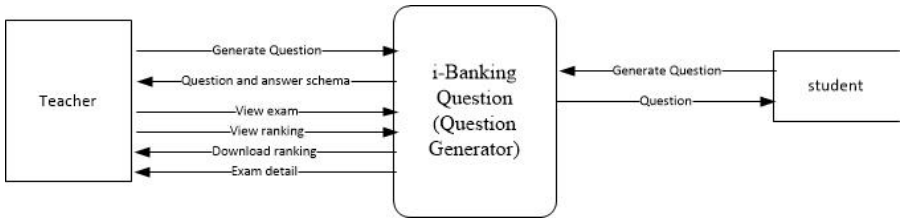


Figure 3.7: Context Diagram of system to be develop

Data Flow Diagram (DFD) Level 0

A data flow diagram (DFD) is a graphical representation the flow of data through an information system. DFD also be used for the visualization of data processing. The figure below shows the flow of the system and the operation of that system.

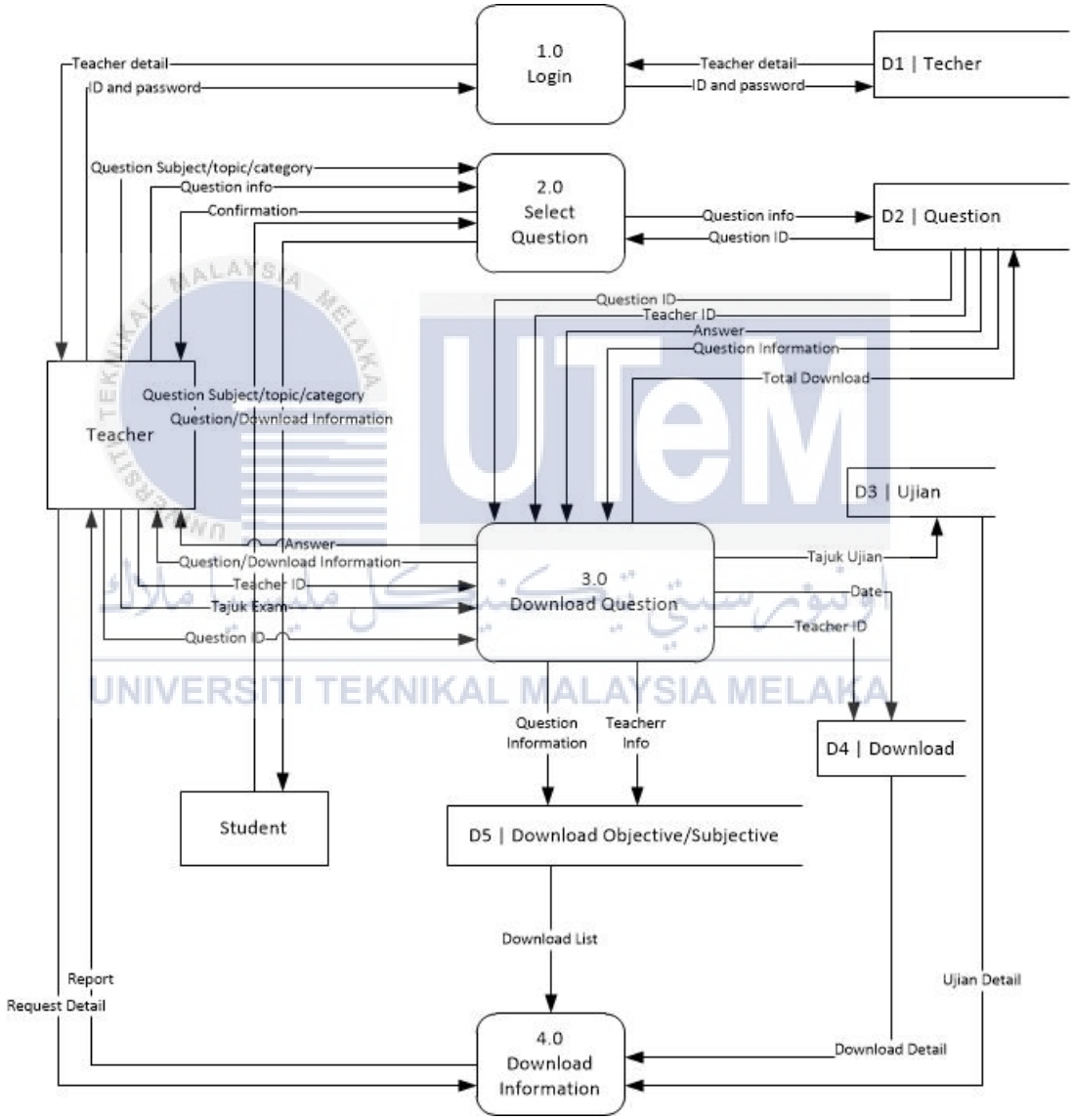


Figure 3.8: DFD Level 0 of System to be developed

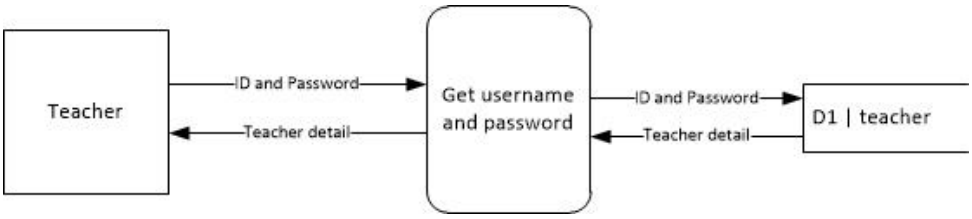


Figure 3.9: DFD Level 1 for Login Process

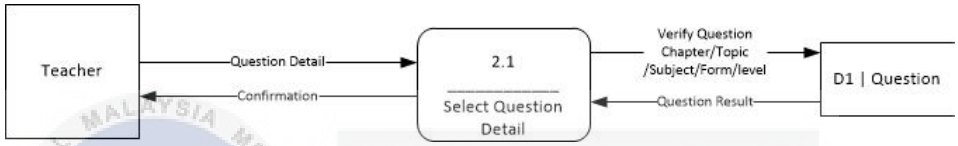


Figure 3.10: DFD Level 1 Teacher Search Question

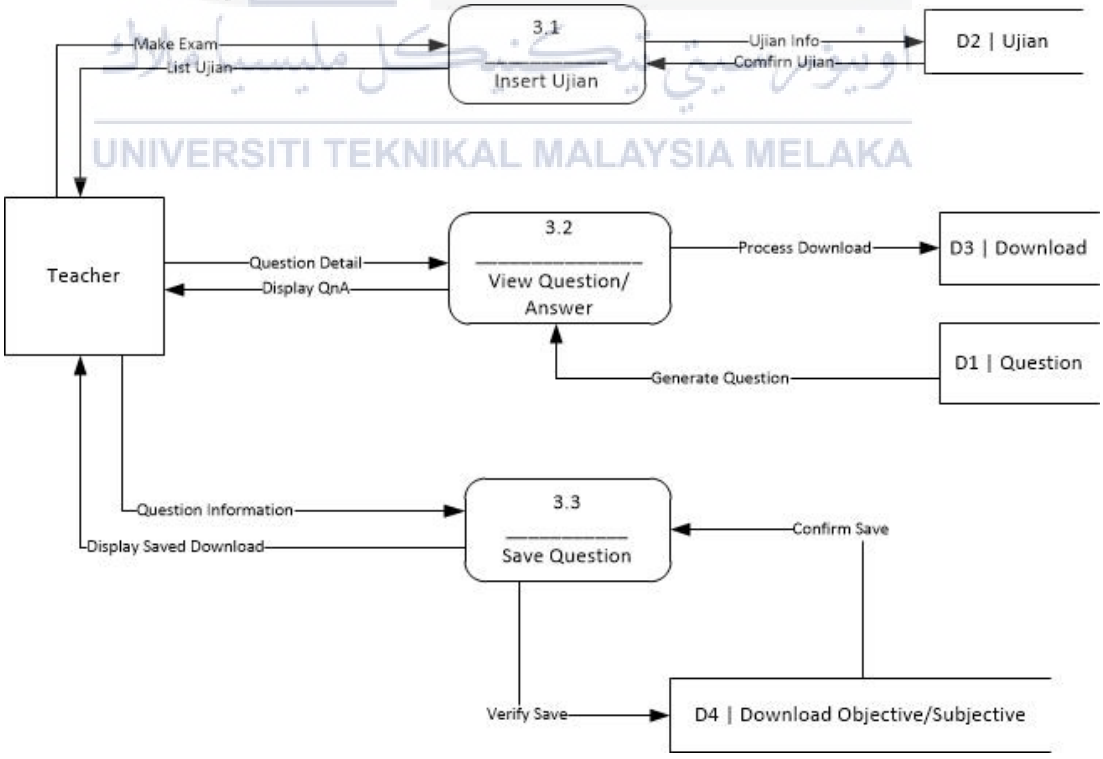


Figure 3.11: DFD Level 1 for Teacher Download Question

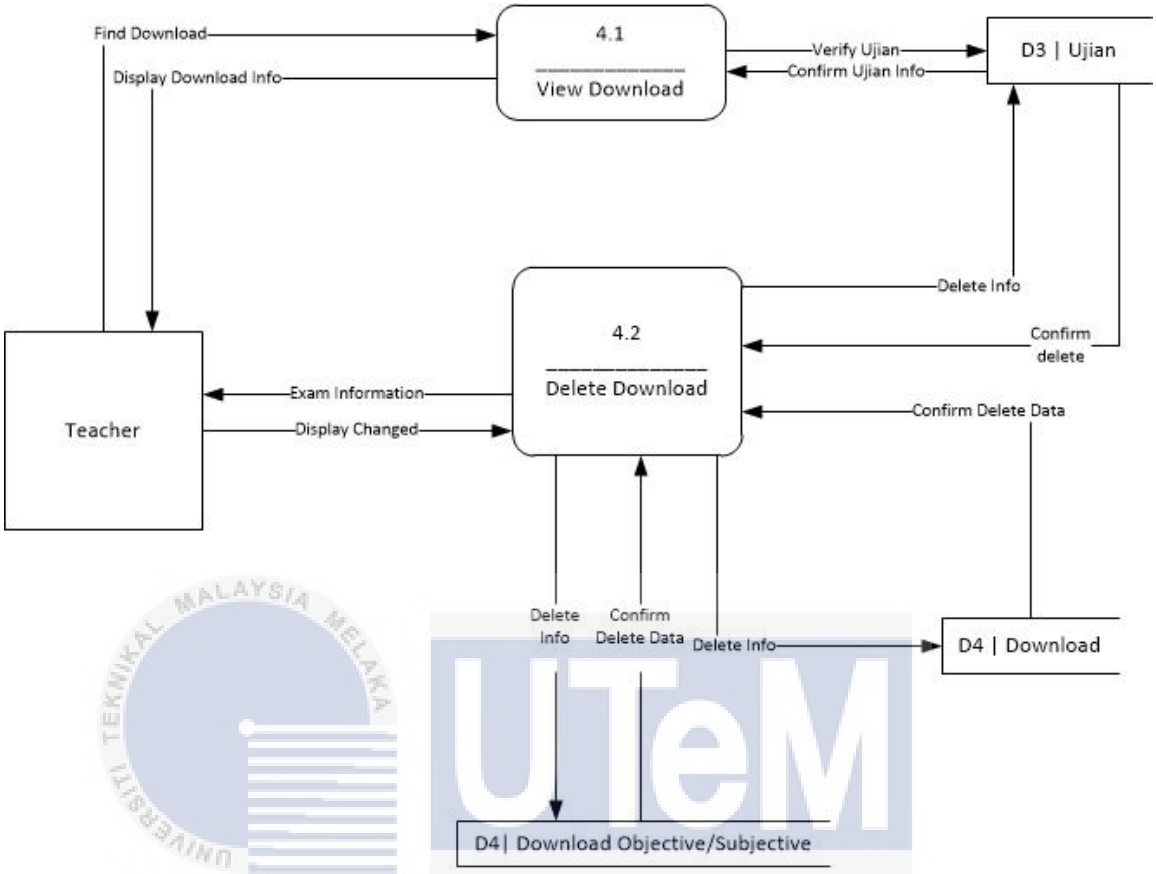


Figure 3.12: DFD Level 1 for Manage Ujian

3.4.2 Non-functional Requirement

The some non-functional requirement for this system that must consider to make sure the system will functional well. The system might be changed in the future, so the maintainability is important because it take large amount to spend on the changing. Secondly is reliability, where the system has capability of the software to maintain its performance over time Unreliable software fails frequently, and certain tasks are more sensitive to failure because they must be run at a certain time.

The third one is security. The integrity requirements determine the properties of the security system, or restrict access to its data to specific users and protect the privacy of the data entered into the software. For example, the teacher must login to make any changes to the question and answer.

3.4.3 Other Requirements

Below are defined the specific project requirement in terms of software and hardware to be develop this i-Bank Questions (Question Generator)

3.4.3.1 Software Requirement

Table 3.1: Software Requirement for Development

Software	Description
Java EE-Eclipse	Tools for Java developers creating Java EE and Web applications
Sublime Text Editor	Used to read and write the coding for the system.
Microsoft Visio 2013	Used to create or draw diagrams such as ERD for database design, Context Diagram and DFD to show the functional requirement of the system and Flow Chart of the system.
Microsoft Office Word 2013	Used for the documentation.
Microsoft Project	Used to manage the project purpose such as Gantt Chart to show the timeline or milestones for the project development

Table 3.2: Software Requirement for Server

Software	Description
Apache Tomcat v8.0 Server	Web server that implements Java EE specifications - Java Server Pages (JSP)
Microsoft Windows 8.1 Pro	Operating System as a platform for DBMS and system development installed on it. Windows 8 is new released from Microsoft and has better performance than Windows 7.
Oracle	Database to store the data for this system.

Table 3.3: Software Requirement for Client

Software	Description
Mozilla Firefox / Google Chrome / Internet Explorer / Edge	Web browser to preview this website. It is recommended for user to using latest version of it.

3.4.3.2 Hardware Requirement

These hardware requirements are very minimum requirements that can meets client and server need such as personal computer. Nowadays, perhaps the hardware for client and server is much better than these minimum requirements. The hardware requirements are shown in Table 3.4.

Table 3.4: Hardware Requirement

Hardware	Description	Server	Client
Hard Disk	Hard disk is the main storage in a computer where all the software installed on it.	Minimum 10GB free disk space	Minimum 300 MB free disk space
Memory (RAM)	Memory is defined as Random Access Memory (RAM) provides space for the computer to read and write data to the	Minimum requirement of memory is 2 GB, though 3 GB is recommended.	Minimum 512 MB of memory, through 2 GB is recommended.

	accessed by the Central Processing Unit (CPU) or processor		
Processor	Processor is the electronic component which is act as 'brain' for a computer. The higher the processing speed, the better its performance.	Minimum 2.27 GHz speed of CPU processor.	Intel Pentium Minimum 1.3 GHz speed of CPU processor.

3.5 Conclusion

In this chapter, it was discuss on the problem for the current system and requirement analysis for the system to be develop in more specific system that has the characteristics associated with the developed system. Collection of information relating to the system associated with the system requirements can be used to improve the system requirements that are related to the i-Bank Questions (Question Generator)

CHAPTER IV

DESIGN

4.1 Introduction

This section will discuss the detail of the system design. System design is the phase following the previous chapter system analysis. To develop a good system, System Architecture Design started and followed by Database Design. It will explain more detail on the conceptual design, logical design and physical design. This chapter also will discuss about the design of Graphic User Interface (GUI). All the design that are create will show on how the system works.

4.2 System Architecture Design

System architecture design is categorize by the each design or set of relation between the parts of the system. There was three level system design application which is client, application server and database server that are used to improve the performance, flexibility, control and capability. This design is relate to internal interface between the system and the sub system as well as the interface and its user.

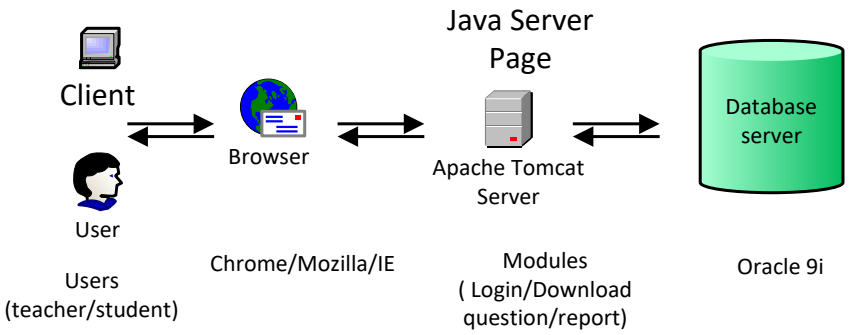


Figure 4.1: System Architecture

4.3 Database Design

Database design consisted of three designs which are Conceptual Design, Logical Design and Physical Design. The conceptual design is the process of producing the detail data model of database which is Entity Relationship Diagram (ERD). The logical design is delivered data dictionary and validate the ERD by using normalization. The physical design includes the implementation of the trigger and stored procedure.

4.3.1 Conceptual Design

Conceptual design is the process of producing a detailed data model of the database. It will show how the data can be related from one data to another. An Entity Relationship Diagram (ERD) was developed to explain this part.

4.3.1.1 Entity Relationship Diagram (ERD)

The conceptual entity relationship diagram model normally defines main reference data entity that used as the foundation for a relational database. The

purpose of the ERD is to establish structural metadata commonality for the master data entities between the set of logical entity relationship model.

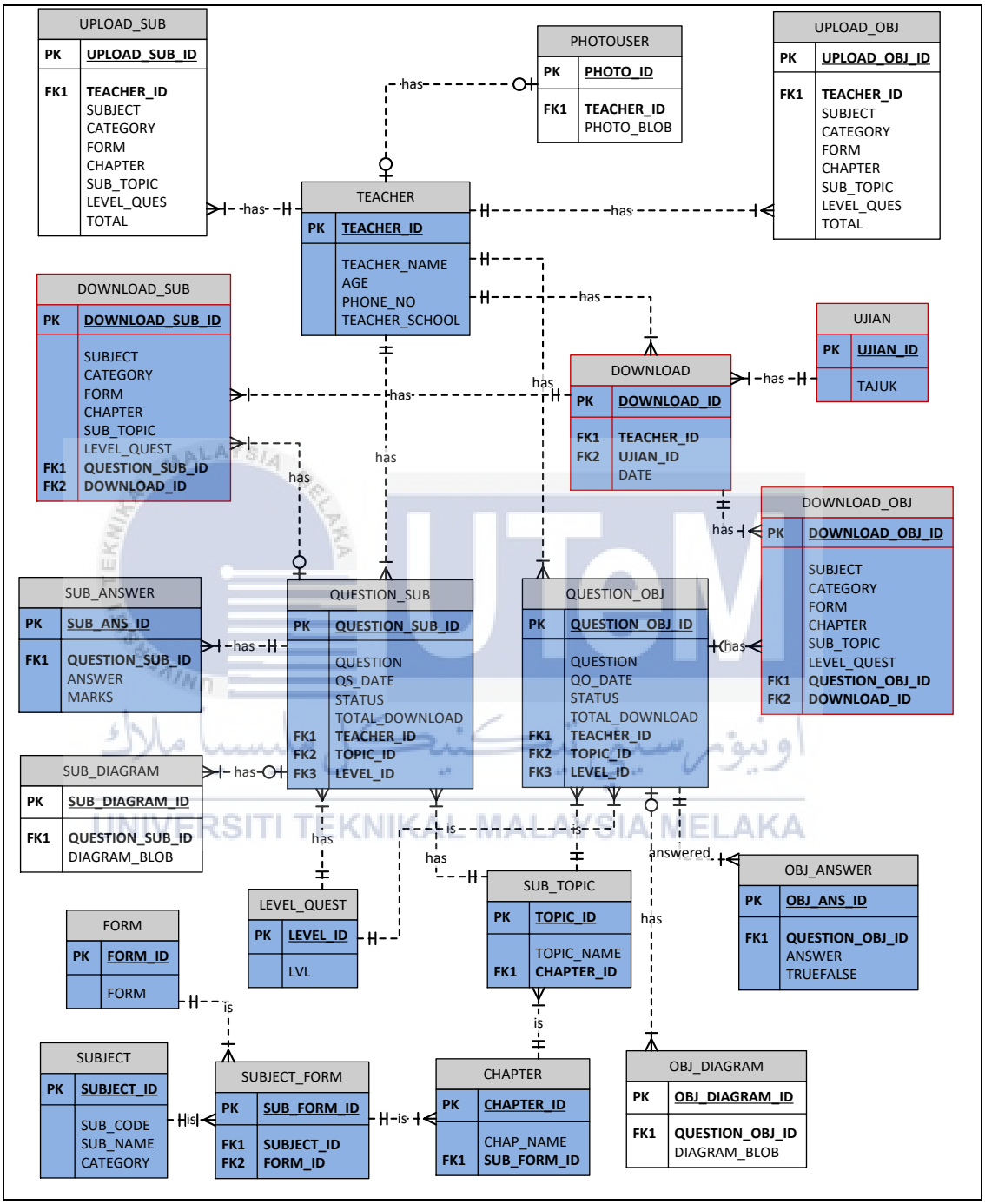


Figure 4.2: Entity Relationship Diagram (ERD)

4.3.1.2 Business Rules

A. Relationship between [TEACHER and DOWNLOAD]
   - One teacher can make many download but
   - One download is downloaded by one teacher

B. Relationship between [UJIAN] and [DOWNLOAD]
   - One examination consist of many download but
   - One download own by one examination

C. Relationship between [DOWNLOAD_OBJ] and [DOWNLOAD]
   One download can contain many download of objective question but
   One download of objective question owns one download.

D. Relationship between [DOWNLOAD_SUB] and [DOWNLOAD]
   One download can contain many download of subjective question but
   One download of subjective question owns one download.

E. Relationship between [QUESTION_OBJ] and [DOWNLOAD_OBJ]
   One objective question has zero or more than one download but
   One downloads of objective question own by one objective question

F. Relationship between [QUESTION_SUB] and [DOWNLOAD_SUB]
   One subjective question has zero or more than one download but
   One downloads of subjective question own by one subjective question

G. Relationship between [QUESTION_OBJ] and [OBJ_ANSWER]
   One objective question have many objective answers but
   One objective answer own by one objective question

H. Relationship between [QUESTION_SUB] and [SUB_ANSWER]
One subjective question have many subjective answers but
One subjective answer own by one subjective question

4.3.2 Logical Design

Logical entity relationship model describe detail of type of information that system needs. The details of each data entity are developed and the entity relationships between these data entities are established.

Teacher (teacher_id, teacher_name, teacher_school, phone_no, age)
Primary key (teacher_id)

Ujian (ujian_id, tajuk)
Primary key (ujian_id)

Download (download_id, dw_date, teacher_id, ujian_id)
Primary key (download_id)
Foreign key (teacher_id) references teacher(teacher_id),
Foreign key (ujian_id) references ujian(ujian_id)

Question_sub(question_sub_id, question, total_download, status varchar2, qs_date date, teacher_id, level_id, topic_id)
primary key (question_sub_id)
Foreign key (teacher_id) references teacher(teacher_id)
Foreign key (level_id) references level_quest(level_id)
Foreign key (topic_id) references sub_topic(topic_id)

Question_obj(question_obj_id,question,total_download,status,qo_date,teacher_id,level_id,topic_id)
primary key (question_obj_id)
foreign key (teacher_id) references teacher(teacher_id)
foreign key (level_id) references level_quest(level_id)
foreign key (topic_id) references sub_topic(topic_id)

Download_sub (download_sub_id, level_quest, sub_topic, chapter, subject, form, category, download_id, question_sub_id)
primary key (download_sub_id)
foreign key (download_id) references download(download_id)
foreign key (question_sub_id) references question_sub(question_sub_id)

Download_obj (download_ obj _id, level_quest, sub_topic, chapter, subject, form, category, download_id, question_sub_id)
primary key (download_ obj _id)
foreign key (download_id) references download(download_id)
foreign key (question_ obj _id) references question_ obj (question_ obj _id)

sub_answer (sub_ans_id, answer, mark, question_sub_id)
primary key(sub_ans_id)
foreign key (question_sub_id) references question_sub(question_sub_id)

obj_answer(obj_ans_id, answer, truefalse)
primary key(obj_ans_id)
foreign key (question_obj_id) references question_obj(question_obj_id)

Creating data dictionary to store any information or database on the system

Table 4.1 Data Dictionary of the System

Table name	**Attribute name**	**Type**	**Required**	**Pk / fk**	**FK Reference table**	**Null / not null**
teacher	teacher_id	varchar2(30)	Yes	PK		Not null
	teacher_name	varchar2(30)				Not null
	teacher_school	varchar2(30)				Not null
	phone_no	varchar2(30)				Not null
	age	varchar2(30)				Not null
Ujian	ujian_id	varchar2(30)	Yes	PK		Not null
	tajuk	varchar2(30)				Not null
Download	download_id	varchar2(30)	Yes	PK		Not null
	dw_date	date				Not null
	teacher_id	varchar2(30)		FK	FK teacher	Not null
	ujian_id	varchar2(30)		FK	FK Ujian	Not null
Question_sub	question_sub_id	varchar2(30)	Yes	PK		Not null
	question	varchar2(200)				Not null
	total_download	Number(20)				Not null
	status	varchar2(30)				Not null

	qs_date	date				Not null
	teacher_id	varchar2(30)		FK	FK teacher	Not null
	level_id			FK	FK level	Not null
	topic_id			FK	FK sub_topic	Not null
Question_obj	question_obj_id	varchar2(30)	Yes	PK		Not null
	question	varchar2(2000)				Not null
	total_download	Number(20)				Not null
	status	varchar2(30)				Not null
	qo_date	date				Not null
	teacher_id	varchar2(30)		FK	FK teacher	Not null
	level_id			FK	FK level	Not null
	topic_id			FK	FK sub_topic	Not null
Download_sub	download_sub_id	varchar2(30)	Yes	PK		Not null
	level_quest	varchar2(30)				Not null
	sub_topic	varchar2(30)				Not null
	chapter	varchar2(30)				Not null
	subject	varchar2(30)				Not null
	form	varchar2(30)				Not null
	category	varchar2(30)				Not null
	download_id	varchar2(30)		FK	FK download	Not null
	question_sub_id	varchar2(30)		FK	FK	Not null

					question_sub	
Download_obj	download_obj_id	varchar2(30)	Yes	PK		Not null
	level_quest	varchar2(30)				Not null
	sub_topic	varchar2(30)				Not null
	chapter	varchar2(30)				Not null
	subject	varchar2(30)				Not null
	form	varchar2(30)				Not null
	category	varchar2(30)				Not null
	download_id	varchar2(30)		FK	FK download	Not null
	question_obj_id	varchar2(30)		FK	FK question_obj	Not null
obj_answer	obj_ans_id	varchar2(30)	Yes	PK		Not null
	answer	varchar2(200)				Not null
	truefalse	varchar2(30)				Not null
	question_obj_id	varchar2(30)		FK	FK question_obj	Not null
sub_answer	sub_ans_id	varchar2(30)	Yes			Not null
	answer	varchar2(200)				Not null
	mark	NUMBER(5,2)				Not null
	question_sub_id	varchar2(30)		FK	FK question_sub	Not null

- **Query Design**

a) Data Definition Language (DDL)

Data definition language is commands that define the different structures in a database. It has a pre-defined syntax for describing data include create statements, modify and remove the database object such as table, index, view and users.

I. Create table and constraint

```
create table teacher (
   teacher_id varchar2(30) primary key not null,
    teacher_name varchar2(30),
    teacher_school varchar2(30),
    phone_no varchar2(30),
    age varchar2(30)
);
```



```
create table question_sub(
    question_sub_id varchar2(30) primary key not null,
    question varchar2(300),
    total_download int,
    status varchar2(30),
    qs_date date,
    teacher_id varchar2(30) references teacher(teacher_id),
    level_id varchar2(30) references level_quest(level_id),
    topic_id varchar2(30) references sub_topic(topic_id)
);
```

```
create table sub_answer(
        sub_ans_id varchar2(30) primary key not null,
        answer varchar2(100),
        mark number(5,2),
        question_sub_id varchar2(30) references question_sub(question_sub_id)
);

create table question_obj(
        question_obj_id varchar2(30) primary key not null,
        question varchar2(300),
        total_download int,
        status varchar2(30),
        qo_date date,
        teacher_id varchar2(30) references teacher(teacher_id),
        level_id varchar2(30) references level_quest(level_id),
        topic_id varchar2(30) references sub_topic(topic_id)
);

create table obj_answer(
        obj_ans_id varchar2(30) primary key not null,
        answer varchar2(100),
        truefalse varchar2(30),
        question_obj_id varchar2(30) references question_obj(question_obj_id)
);

create table ujian(
        ujian_id varchar2(30) primary key not null,
        tajuk varchar2(30)
);
```

```
create table download(
        download_id varchar2(30) primary key not null,
        dw_date date,
        teacher_id varchar2(30) references teacher(teacher_id),
        ujian_id varchar2(30) references ujian(ujian_id)
);

create table download_sub(
        download_sub_id varchar2(30) primary key not null,
        level_quest varchar2(30),
        sub_topic varchar2(30),
        chapter varchar2(30),
        subject varchar2(30),
        form varchar2(30),
        category varchar2(30),
        download_id varchar2(30) references download(download_id),
        question_sub_id varchar2(30) references question_sub(question_sub_id)
);

create table download_obj(
        download_obj_id varchar2(30) primary key not null,
        level_quest varchar2(30),
        sub_topic varchar2(30),
        chapter varchar2(30),
        subject varchar2(30),
        form varchar2(30),
        category varchar2(30),
        download_id varchar2(30) references download(download_id),
        question_obj_id varchar2(30) references question_obj(question_obj_id)
);
```

b) Data Manipulation Language (DML)

Data Manipulation Language is a syntax element that used for selecting, inserting, deleting and updating data in database. It was similar with other computer programming language.

i. Insert statement

```
INSERT INTO question_sub VALUES ('qs01','what are the molecule state that occur during high pressure ?',0,active,sysdate,'t01','lvl01','sb01');
```

ii. Update statement

```
UPDATE question_sub
Set status = ‘Active’
Where question_sub_id = ‘Q0032’;
```

iii. Delete statement

```
DELETE FROM ujian
WHERE ujian_id = ‘UJI023’;
```

iv. Rollback statement

```
DELETE FROM ujian
WHERE ujian_id = ‘UJI023’;
ROLLBACK;
```

v. Commit statement

```
DELETE FROM ujian
WHERE ujian_id = ‘UJI023’;
ROLLBACK;
COMMIT;
```

c) Data Achievement

i. Restricting and Sorting Data

```
SELECT * FROM download
WHERE teacher_id = ‘T005’;
```

ii. Displaying Data From Multi Table

```
SELECT q.question ,s.answer
FROM  question_obj q, obj_answer s
WHERE q.question_obj_id = s.question_obj_id;
```

iii. Aggregation Data using Group Function

```
SELECT t.teacher_name, count(d.download_id)
FROM teacher t
FULL OUTER JOIN download d on t.teacher_id = d.teacher_id
GROUP BY t.teacher_name;
```

iv. Subqueries

```
SELECT * FROM question_sub
WHERE level_id = (SELECT level_id
FROM level_quest
WHERE lvl ='knowledge')
```

4.3.3 Physical Design

Oracle 9i used to store all the data and information in i-Bank Questions (Question Generator) includes all the stored procedure and trigger.

1. Usage of Stored Procedure

i. Procedure to insert data into download table.
ii. Procedure to insert data into ujian table.
iii. Procedure to insert data into download_sub table.
iv. Procedure to insert data into download_obj table.
v. Procedure to delete data from ujian table.
vi. Procedure to login.
vii. Procedure to generate and view question.
viii. Procedure to view report.

2. Usage of trigger

i. Trigger to automatically generate ID for ujian table, download table, download_sub table and download_obj table. Current date automatically to insert into ujian table as sysdate. Each time new data inserted, all ID and date will insert automatically.

ii. Trigger to automatically increment total_download for question_obj and question_sub table, Each time teacher download the question, the total download of the question will plus by one.

iii. Trigger to automatically delete for download_obj and download_sub table each time data related in download table was deleted. Each time teacher delete their examination, all data about the question on download table will delete automatically.

3. Security Mechanism

i. Login

There are two user in i-Bank Questions (Question Generator) that is teacher and student, teachers must login to enter the system as teacher in order to download the question

ii. Privileges

This system can gran the privileges to the user which is teacher or student, teacher can download question with answers scheme but student only the question.

iii. Validation

All the data that will insert into the database will be validate first.

4.4 Graphical User Interface (GUI) Design

The graphical user interface design is to make sure that user easy to use the system without facing the coding environment. It defined the functionality of the system in order the user use the system easily.

4.4.1 Navigational Design

The navigation design is to describe the flow and transaction that involve in the i-Bank Questions (Question Generator) system.

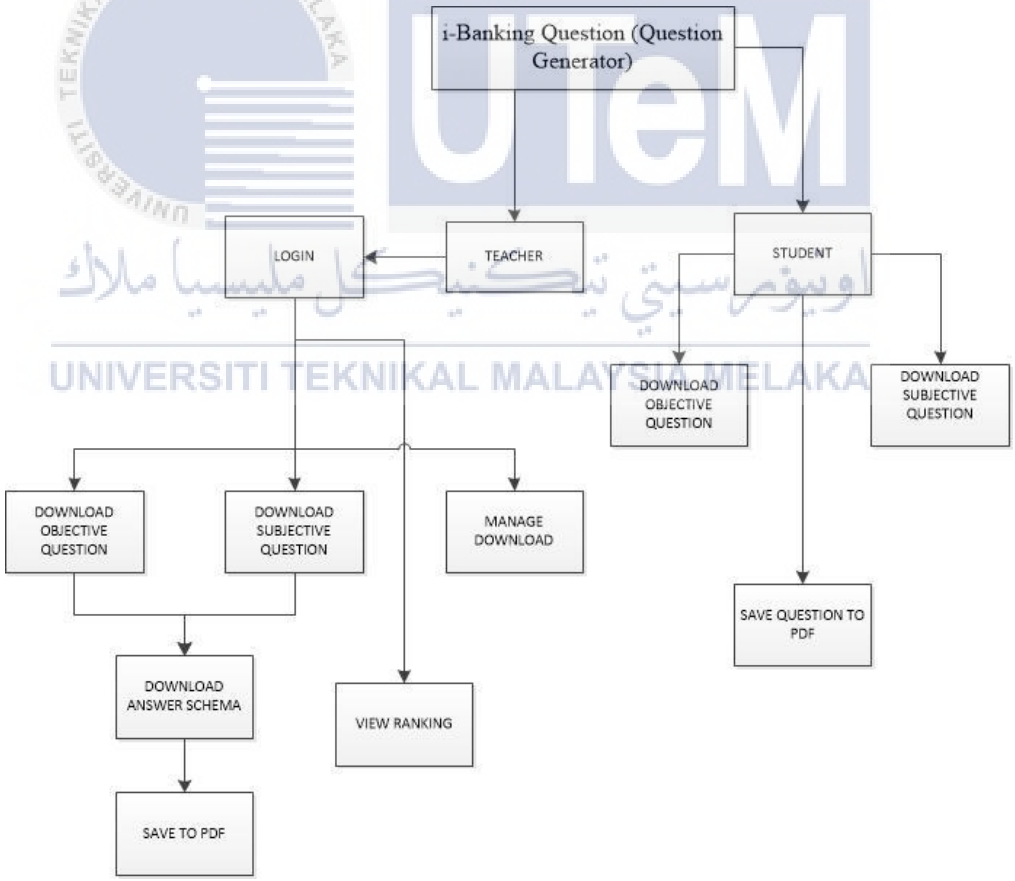


Figure 4.3 Navigation Design

4.4.2 System Interface

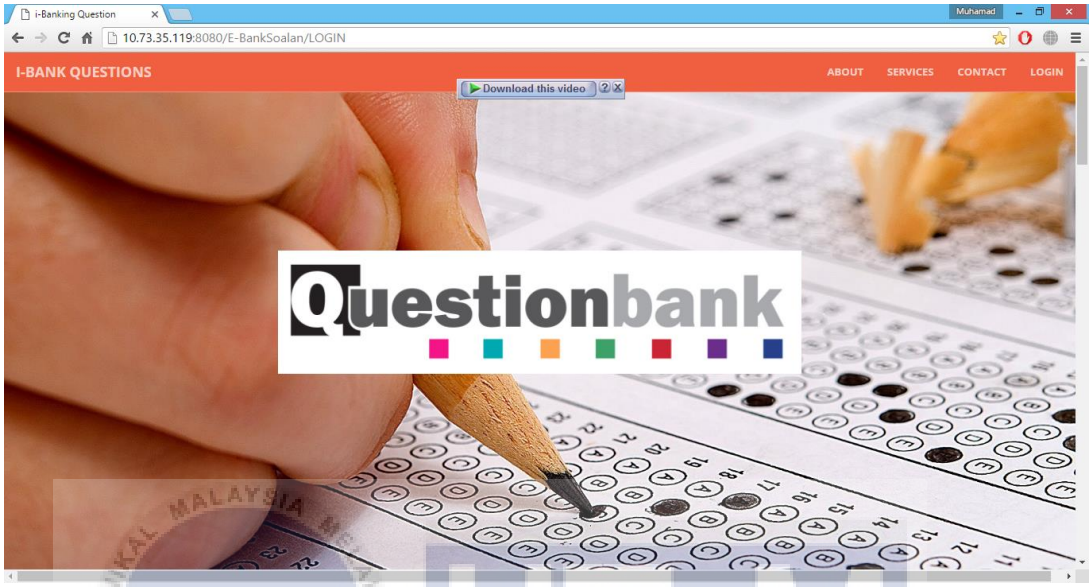


Figure 4.4: Home page i-Bank Questions (Question Generator)

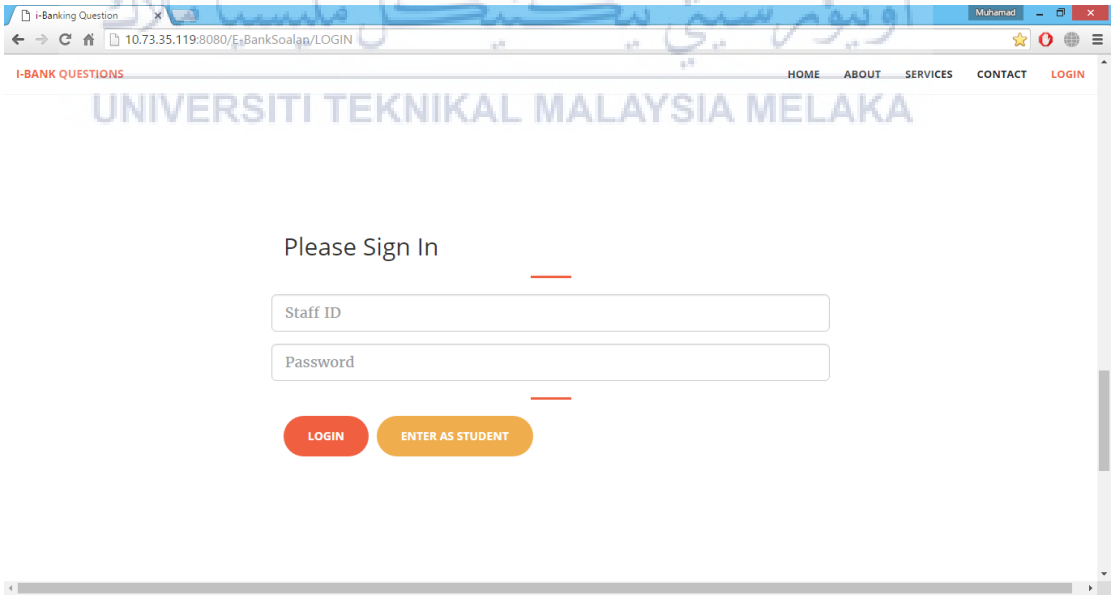


Figure 4.5: Login Page

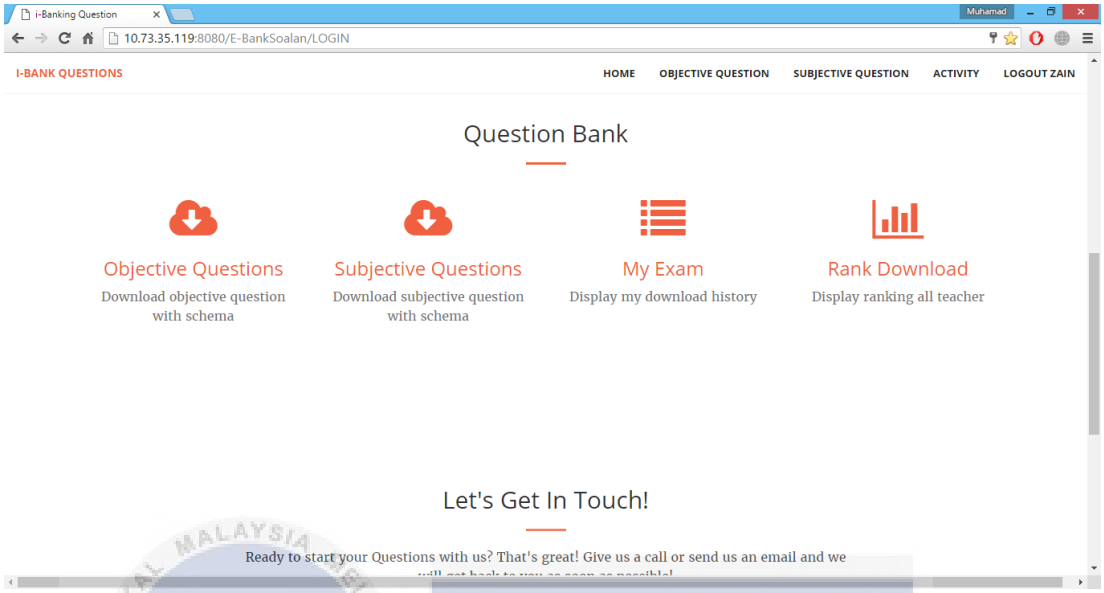


Figure 4.6: Main Page for Teacher

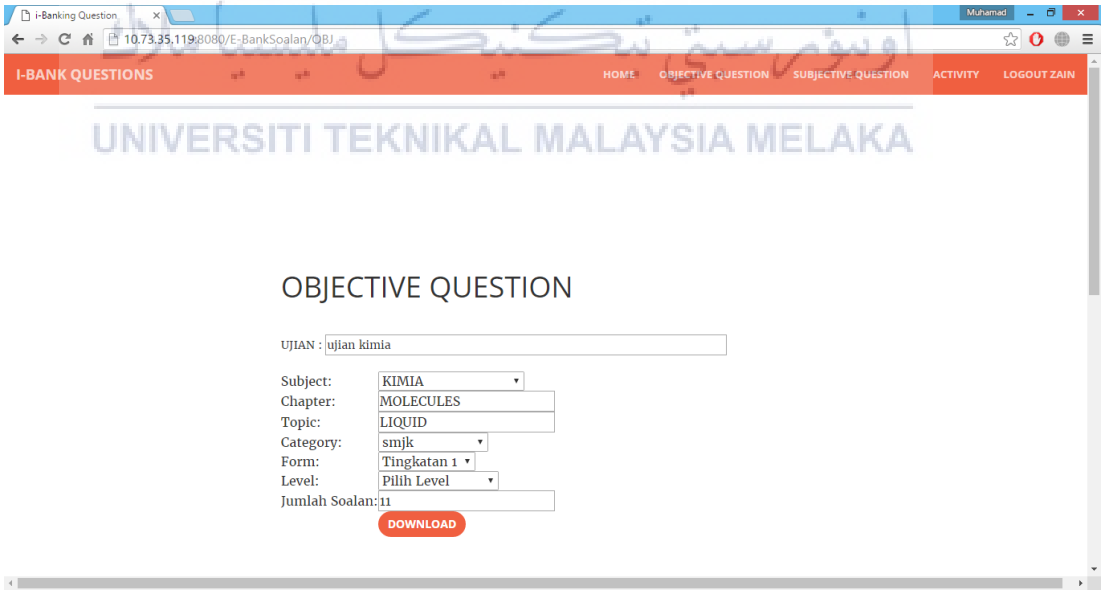


Figure 4.7: Teacher search question to download

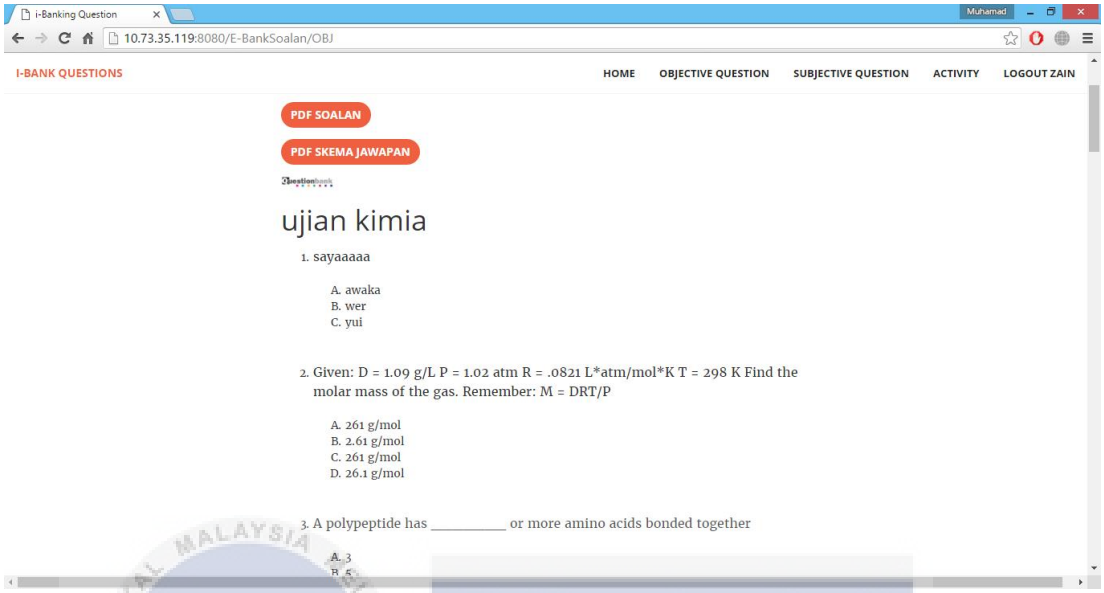


Figure 4.8: Question generated base on searching

Figure 4.9: Page of teacher's download.

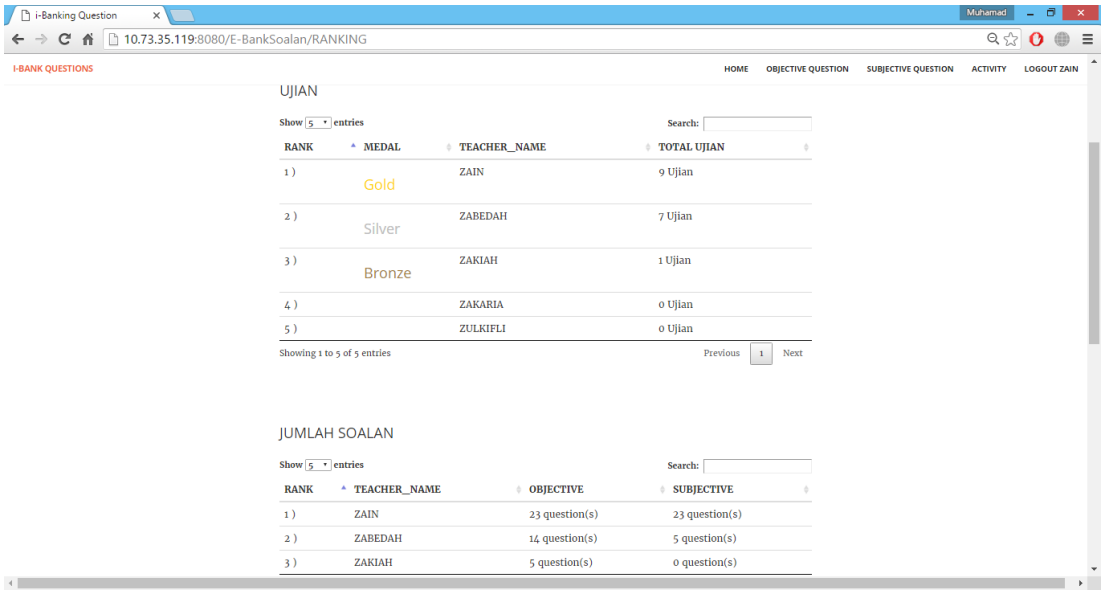


Figure 4.10: Ranking page.

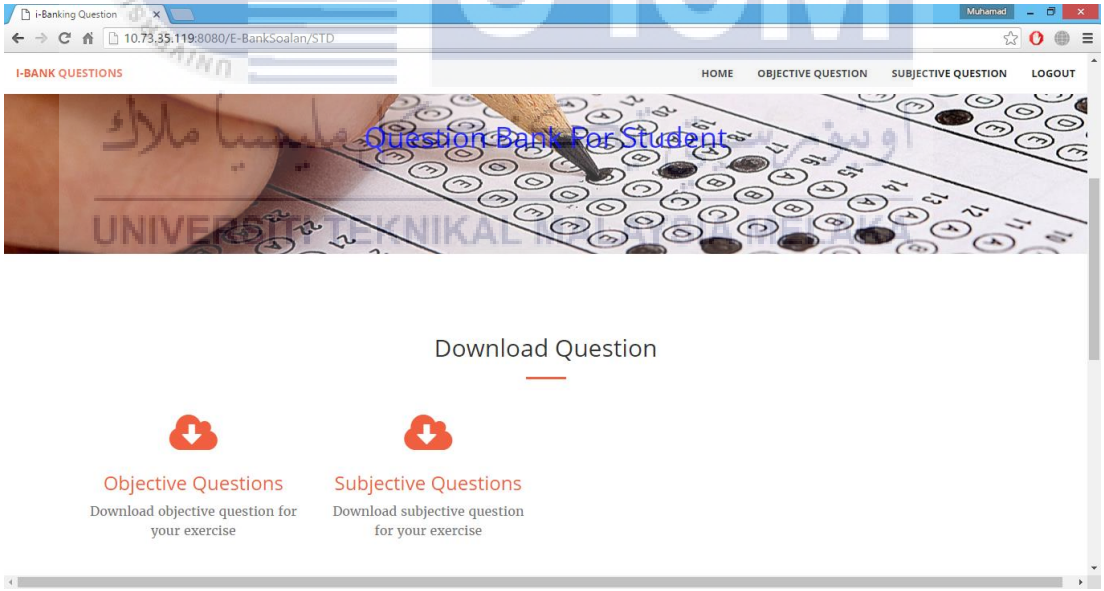


Figure 4.11: Home page for student.

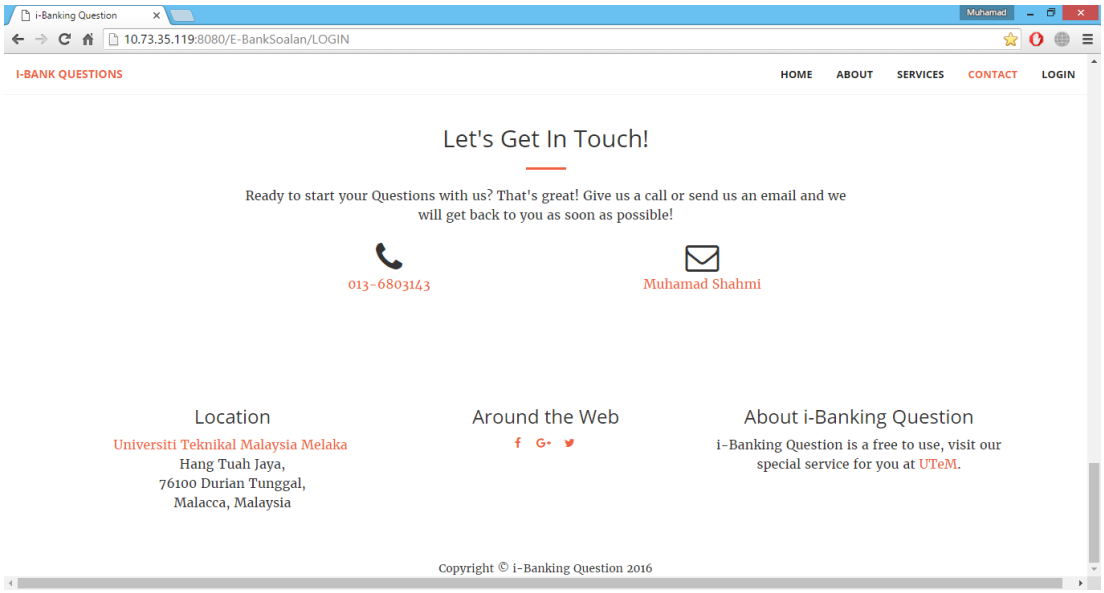


Figure 4.12: Contact page.

4.5 Conclusion

This chapter was discussing on the design that involve to complete this i-Bank Questions (Question Generator) system. System Architecture Design was discuss on the structure that facilitates the database to complete a transaction. Entity Relationship Diagram has been discussing in this chapter with the diagram and the relationship between the entities and the attributes. The next chapter will discuss the implementation for this i-Bank Questions (Question Generator) system based on this design chapter.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

This chapter explains detail about the implementation of the i-Bank Questions (Question Generator) that divided in two which is system development environment setup and database implementation. The activity involve in this implementation phase is software development which discuss the architecture of client and server software and database. This chapter also explains the configuration of the software to meet the system requirement.

5.2 System Development Environment setup

i-Bank Questions (Question Generator) are using Java application and tis run with apache tomcat. Apache Tomcat Server that act as the localhost. Oracle 11g Enterprise Edition used as the database server to store all the data. The user interface is design using Eclipse Java EE.

5.2.1 Installation step

5.2.1.1 Oracle 11g

STEP 1: Open the Oracle Database 11g installer and click next button.

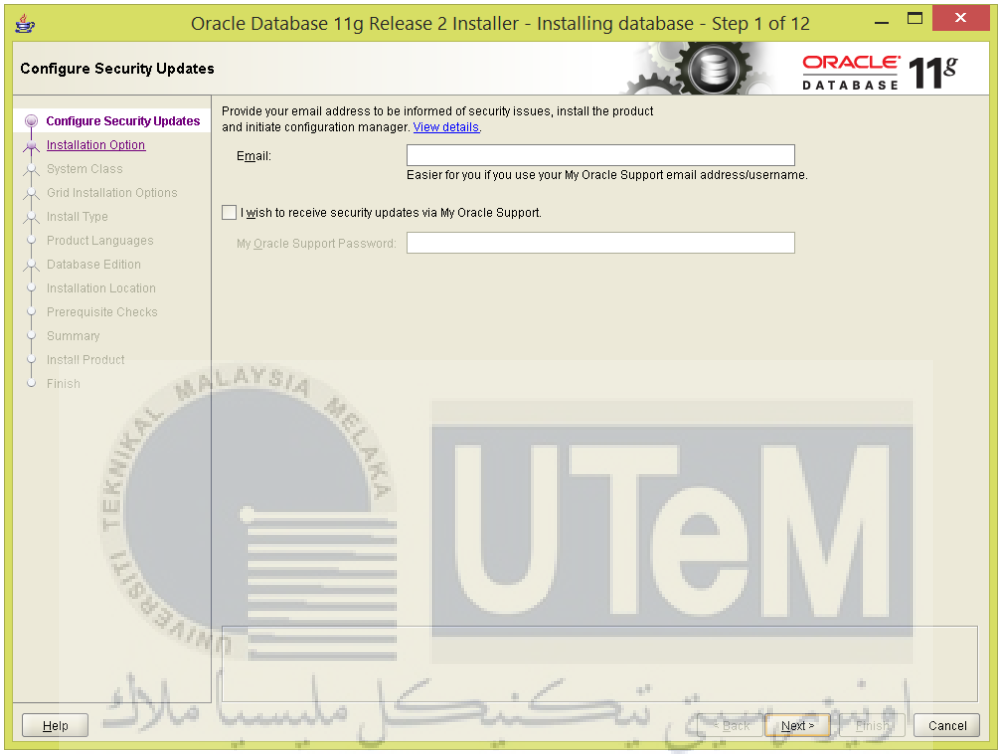


Figure 5.1: Database Installation Step 1

STEP 2: Choose Create and configure a database and click next button.

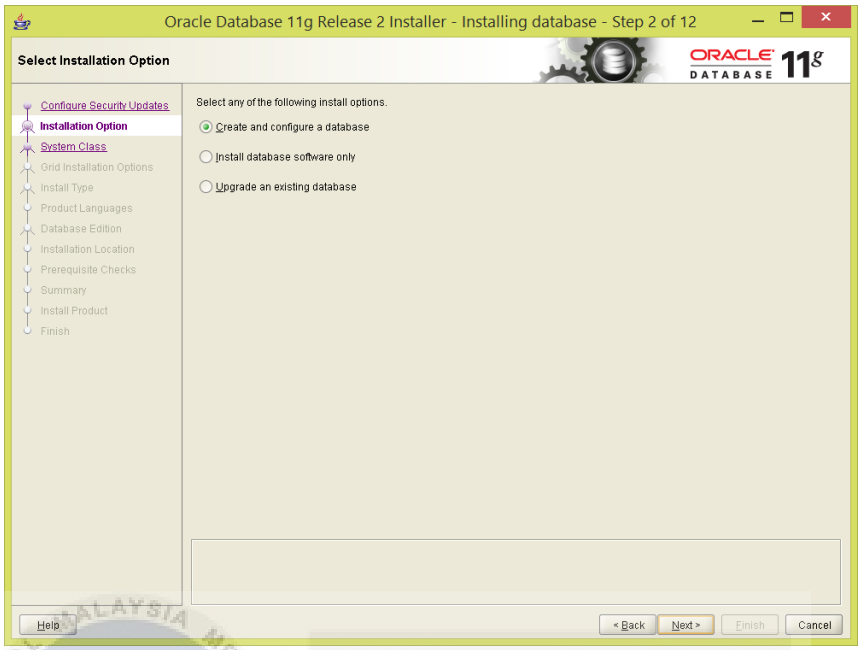


Figure 5.2: Database Installation Step 2

STEP 3: Tick Server class and click next button

Figure 5.3: Database Installation Step 3

STEP 4: Tick on Single instant database installation and click next button.

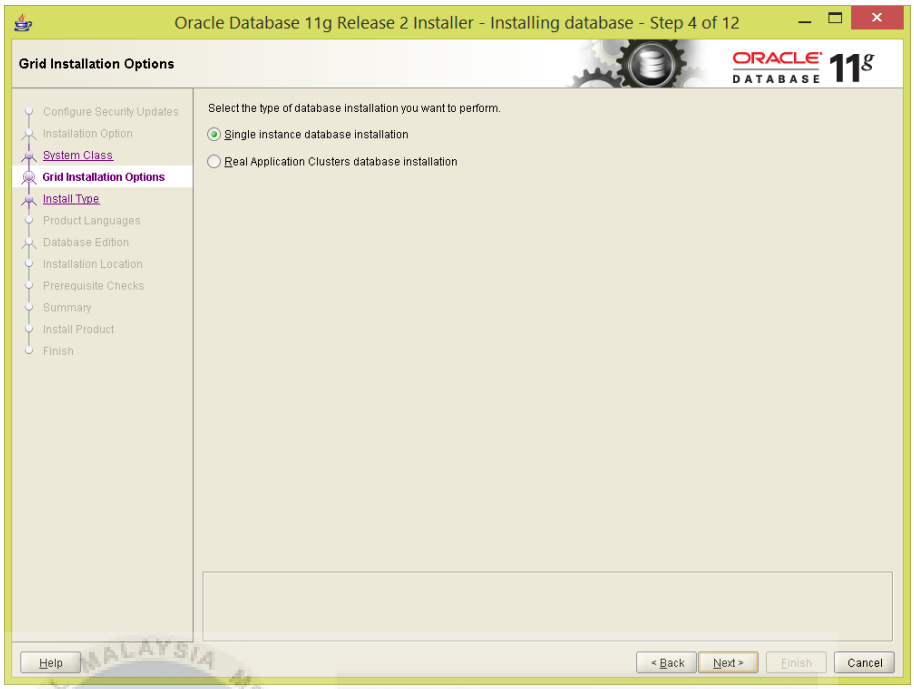


Figure 5.4: Database Installation Step 4

STEP 5: Tick Advanced install and click next button.

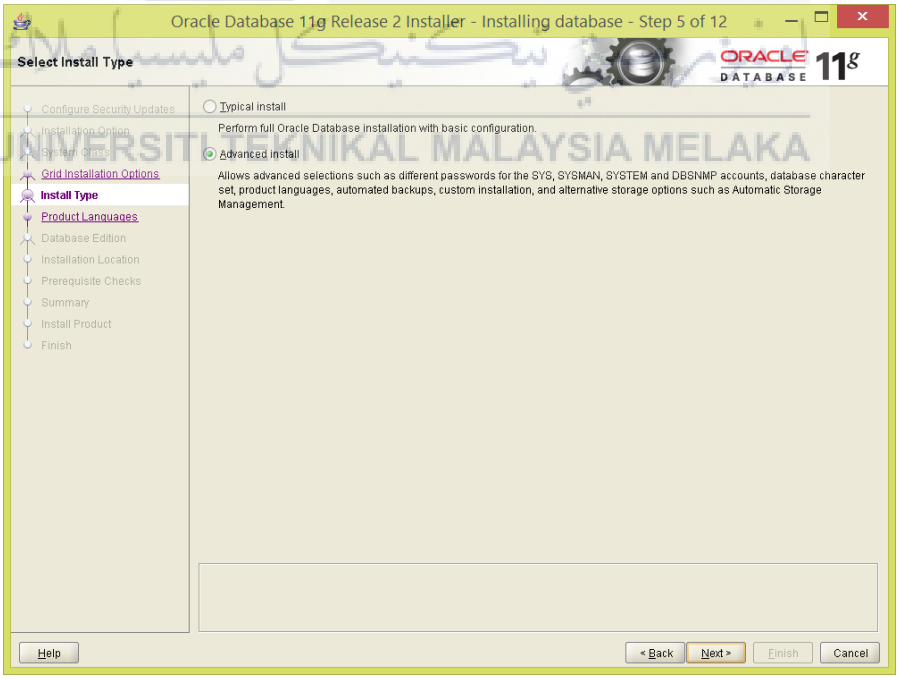


Figure 5.5: Database Installation Step 5

STEP 6: Choose English language and click next button.

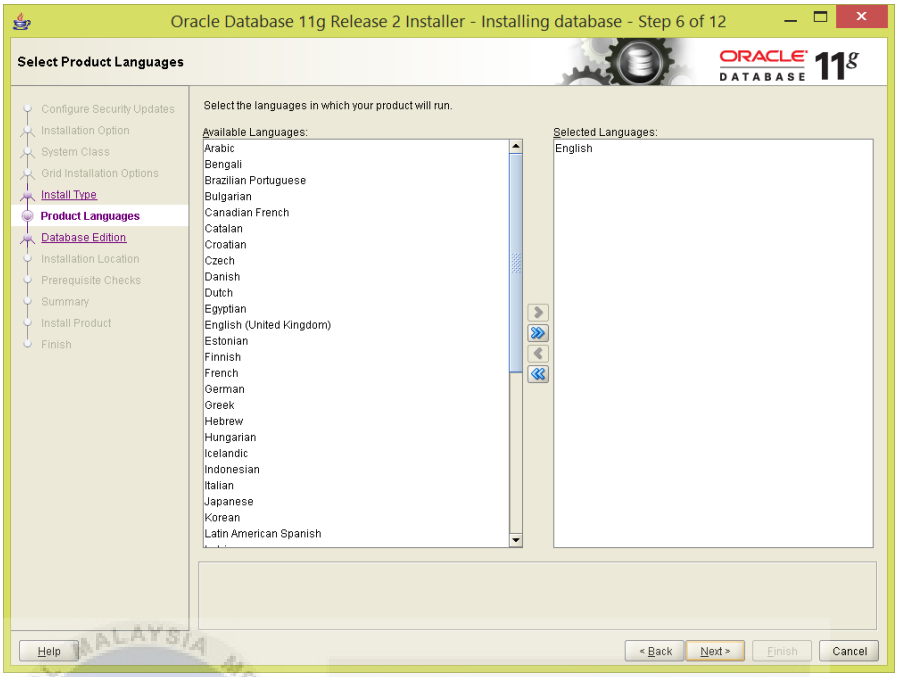


Figure 5.6: Database Installation Step 6

STEP 7: Choose the Enterprise Edition and click next button.

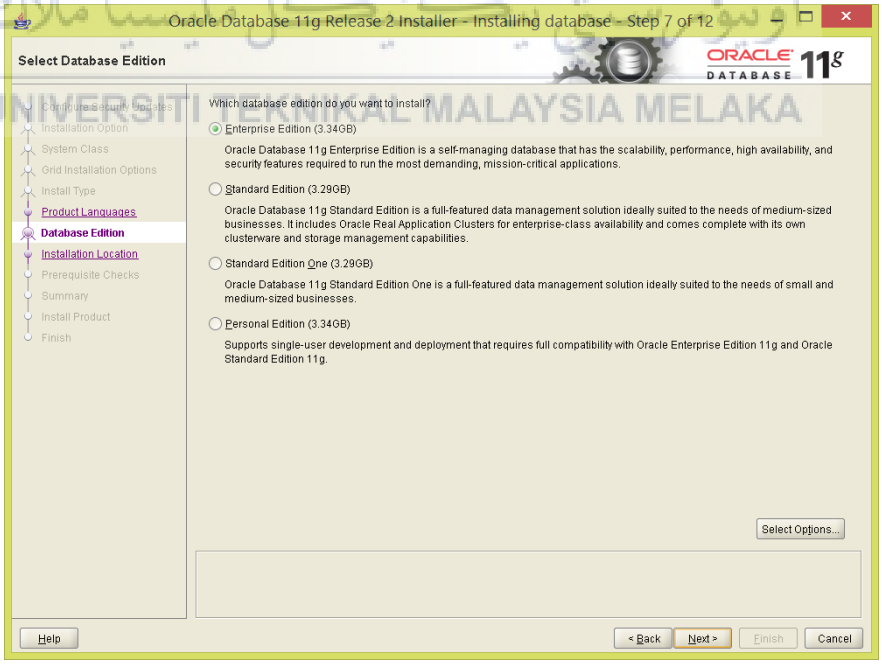


Figure 5.7: Database Installation Step 7

STEP 8: Select the location for the installation by click the Browse button and find the location in the file system and click next button.

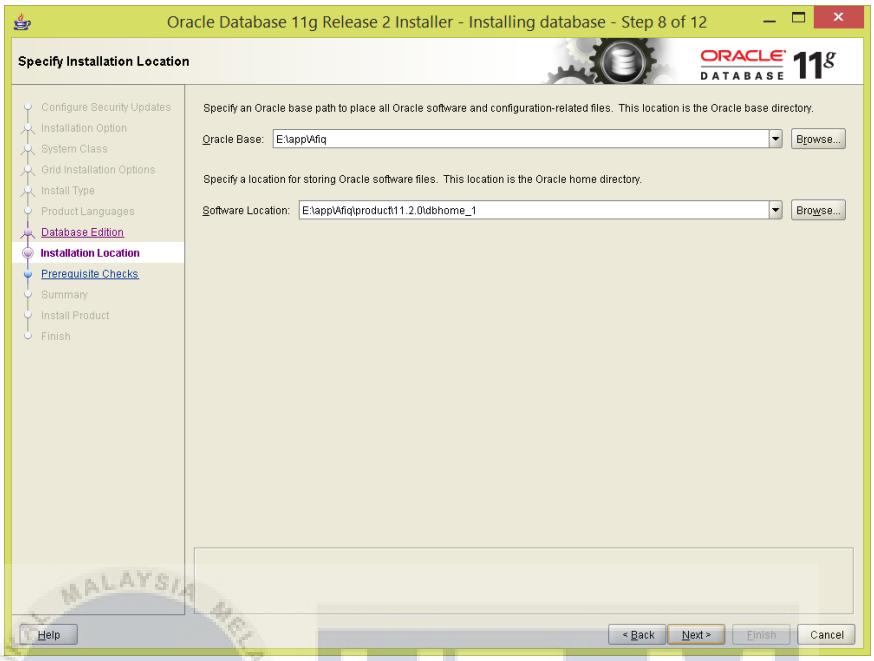


Figure 5.8: Database Installation Step 8

STEP 9: Tick on General Purpose / Transaction Processing and click next button.

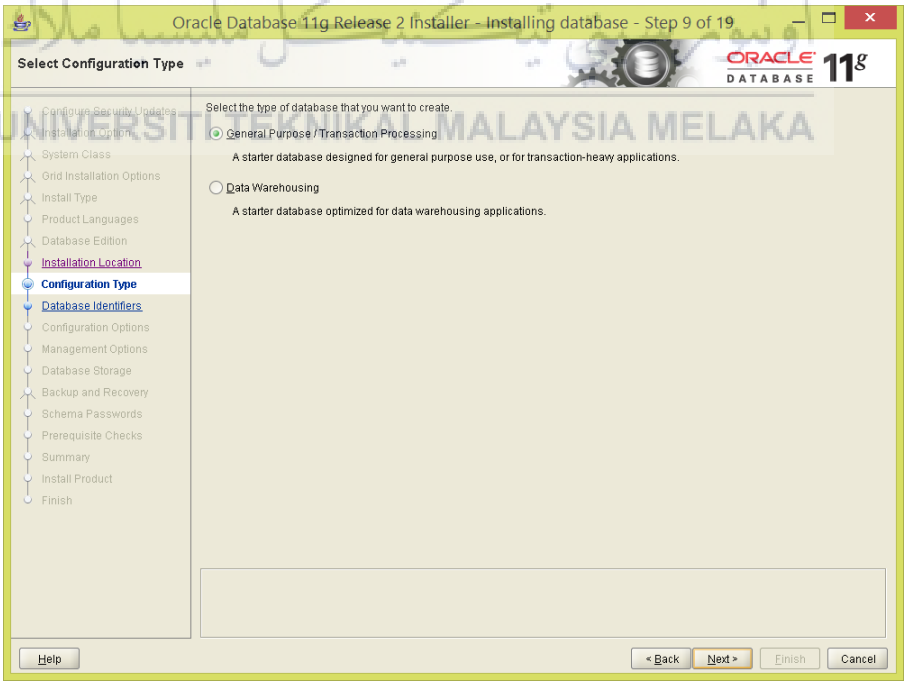


Figure 5.9: Database Installation Step 9

STEP 10: Enter the Database name and Oracle Service Identifier (SID) name and click next button.

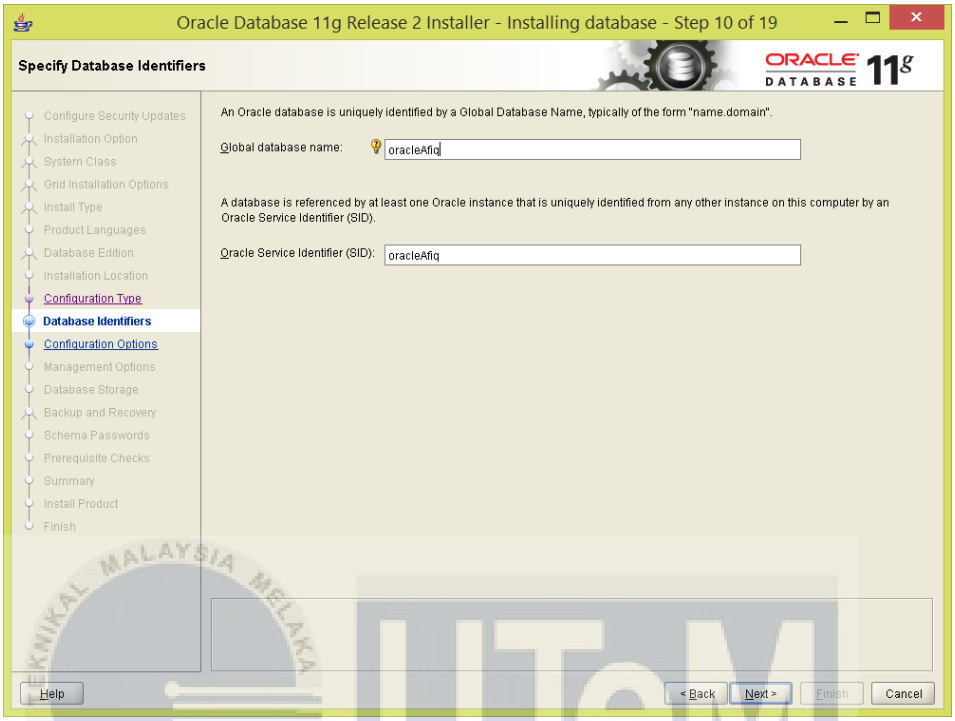


Figure 5.10: Database Installation Step 10

STEP 11: Tick on Enable Automatic Memory Management and click next button.

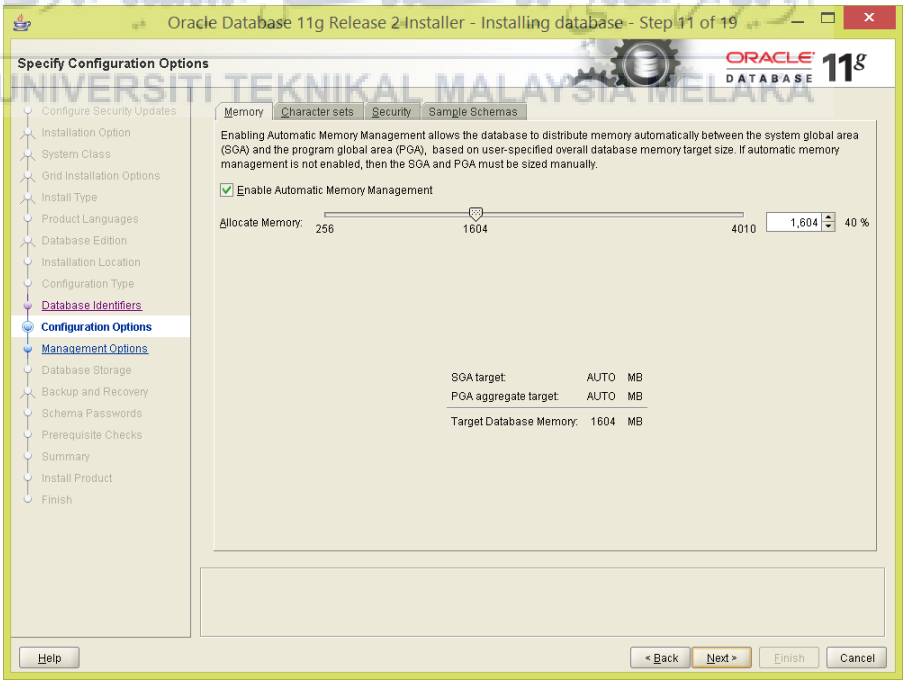


Figure 5.11: Database Installation Step 11

STEP 12: Tick on Use Database Control for Database management and click next button

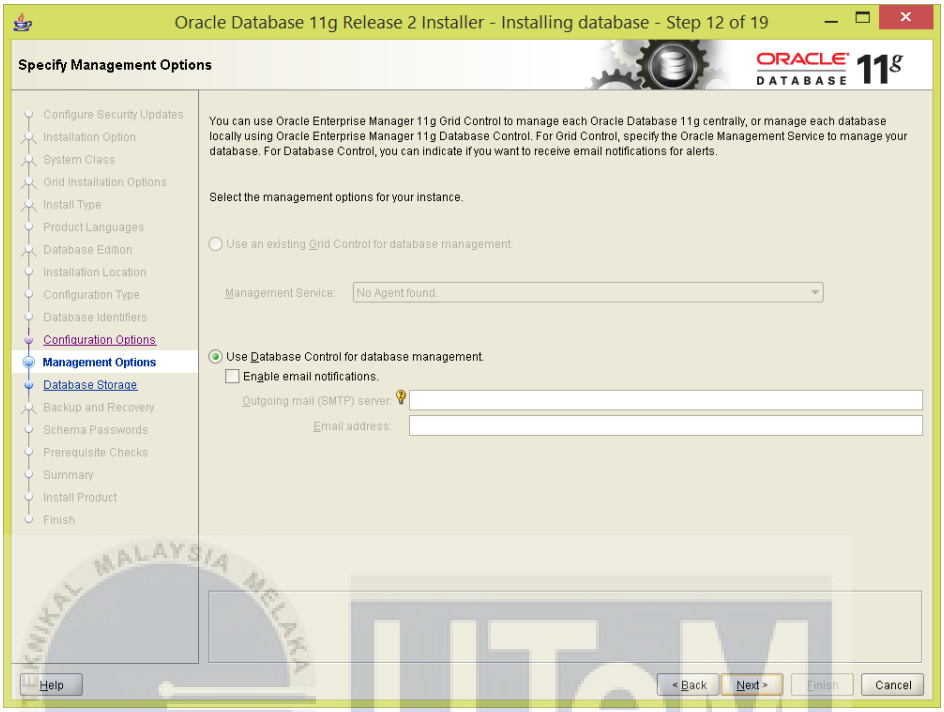


Figure 5.12: Database Installation Step 12

STEP 13: Tick on File System and click next button.

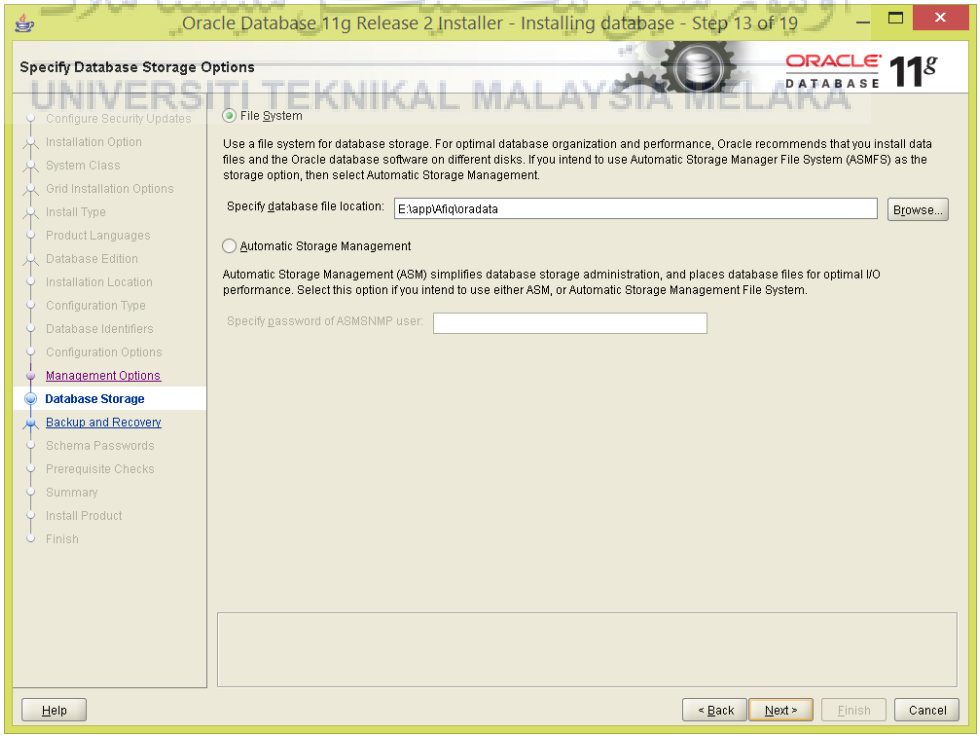


Figure 5.13: Database Installation Step 13

STEP 14: Tick on Enable automated backup and File system, then enter username and password and click next button.

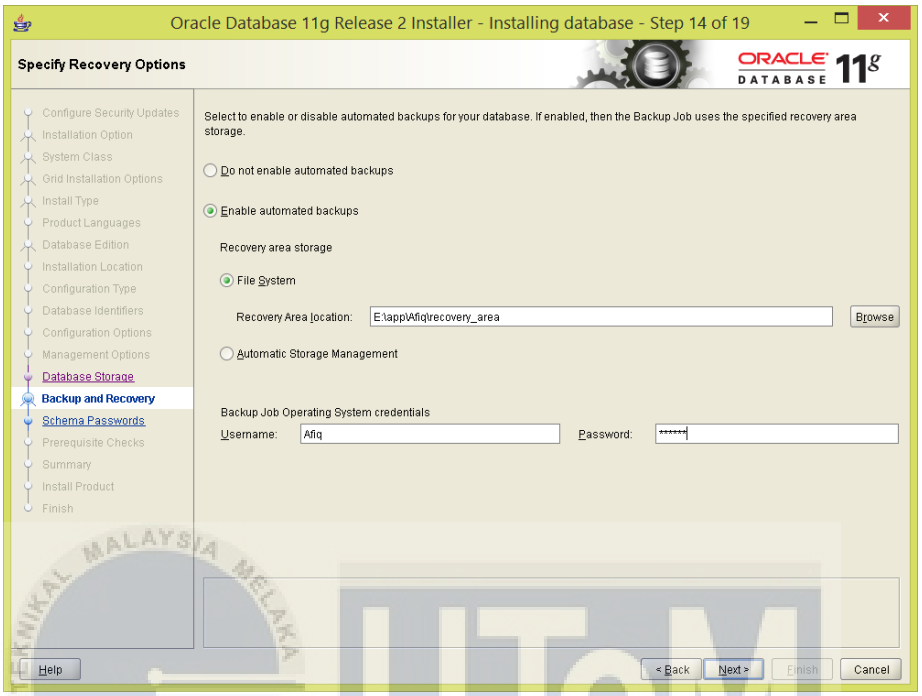


Figure 5.14: Database Installation Step 14

STEP 15: Tick on Use the same password for all account and click next button.

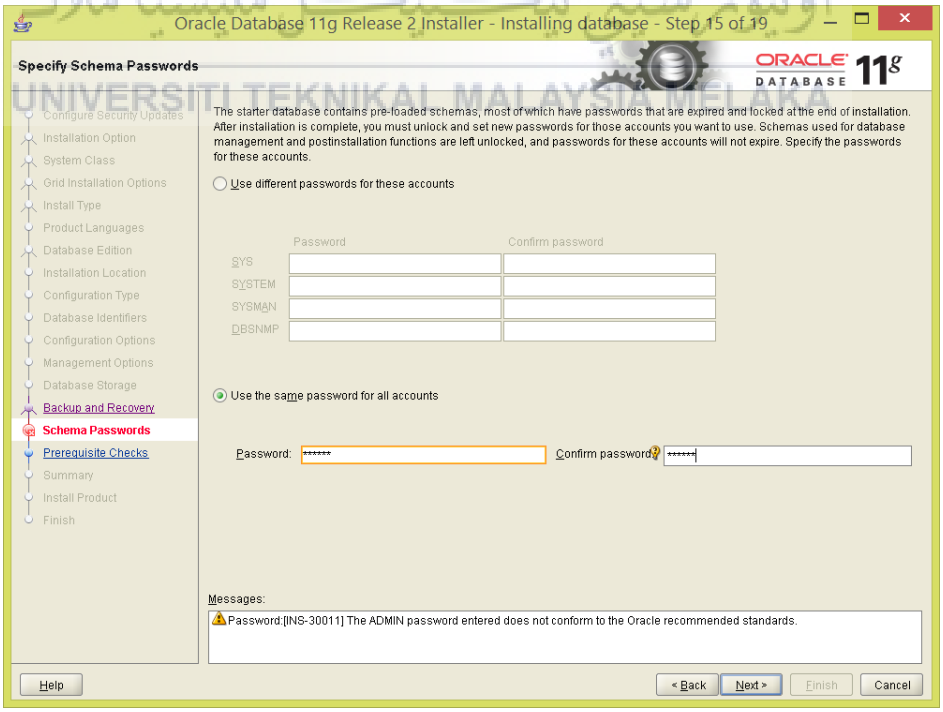


Figure 5.15: Database Installation Step 15

STEP 16: Click finish button.

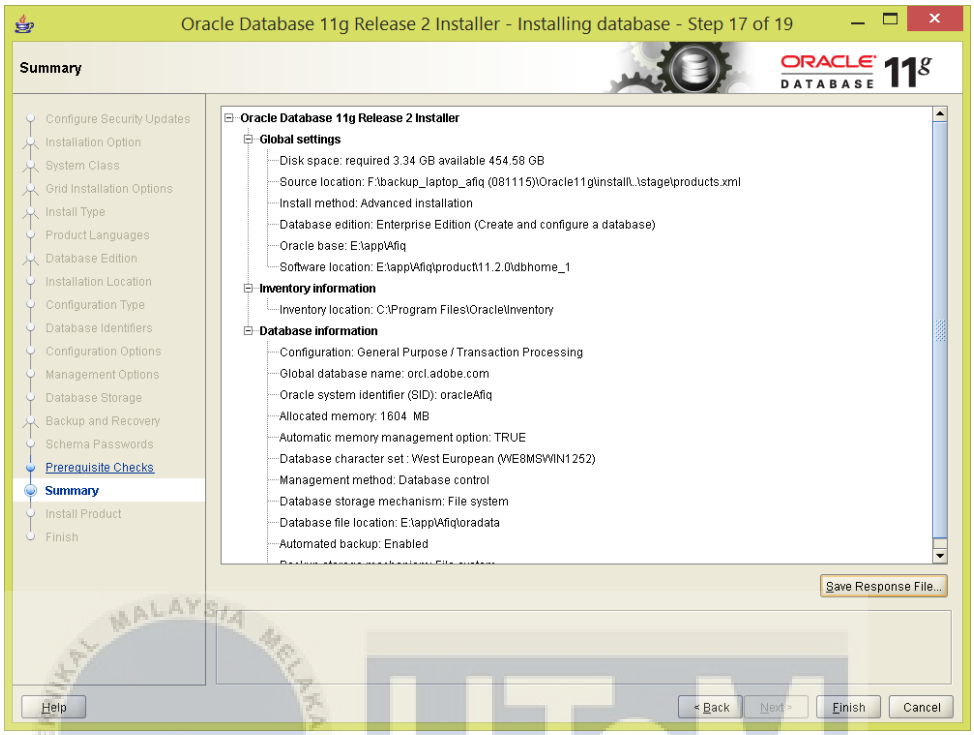


Figure 5.16: Database Installation Step 16

STEP 17: Wait until progress finish

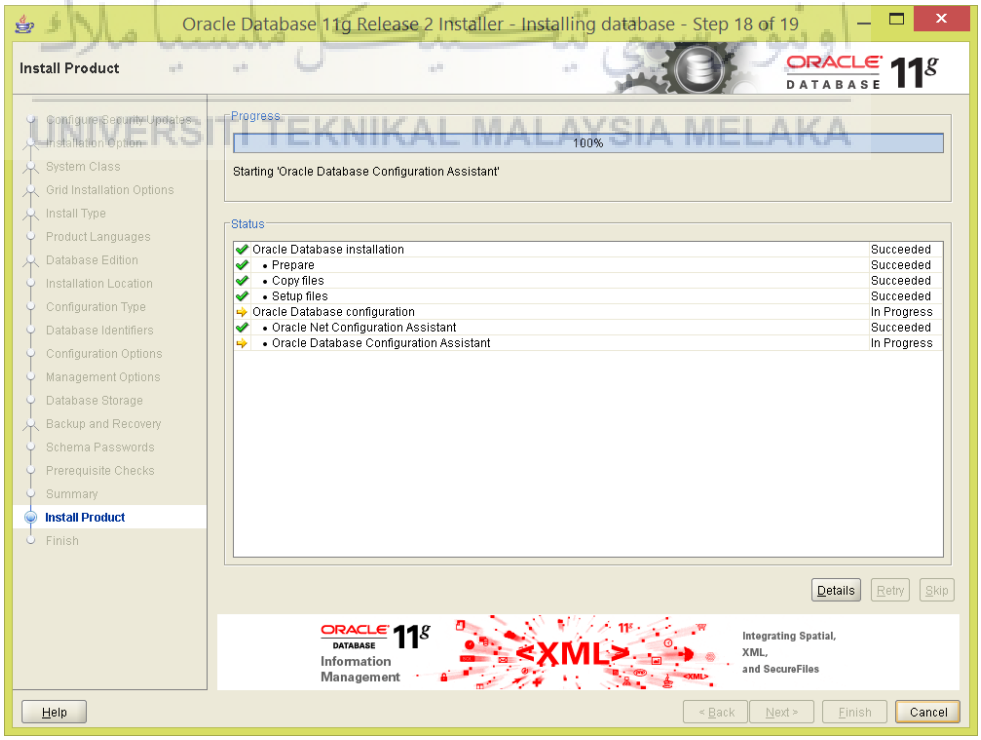


Figure 5.17: Database Installation Step 17

STEP 18: After the installation finish click Close button.

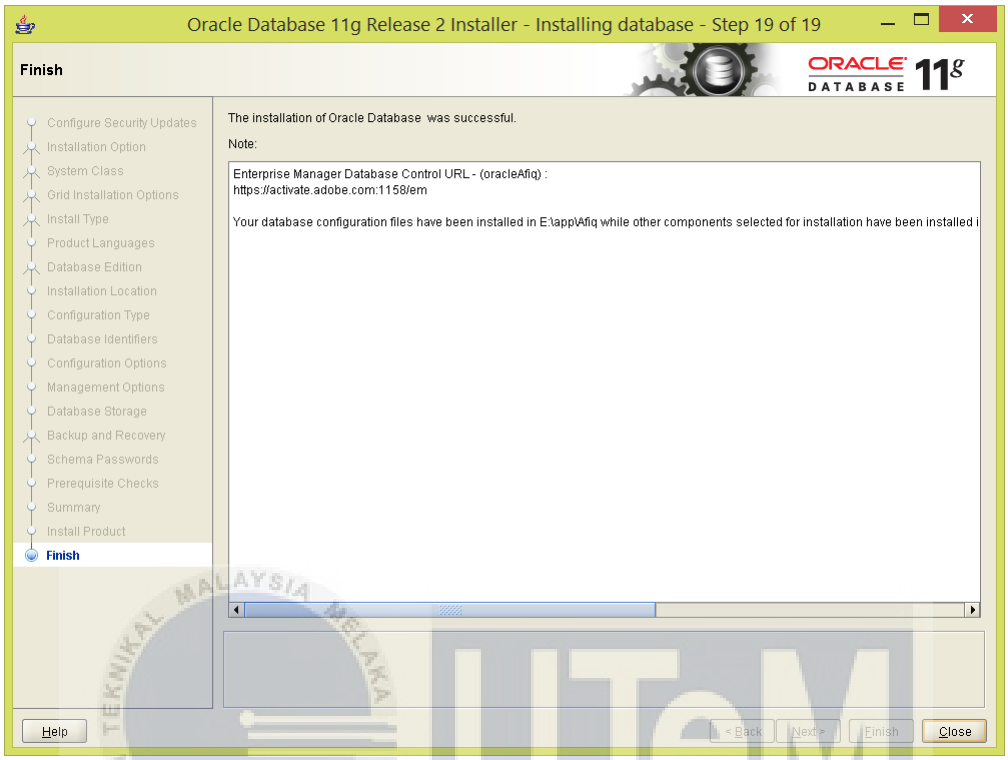


Figure 5.18: Database Installation Step 18

STEP 19: Database is ready to use and can connect as sysdba and username and password.

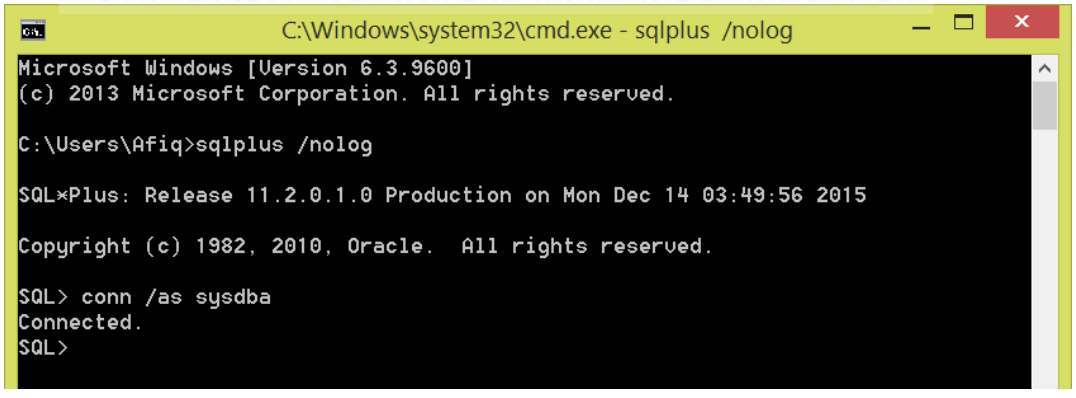


Figure 5.19: Database Installation Step 19

5.2.1.2 Apache Tomcat server and Eclipse Java EE Configuration

STEP 1: Lunch the eclipse and select workspace location in file system.

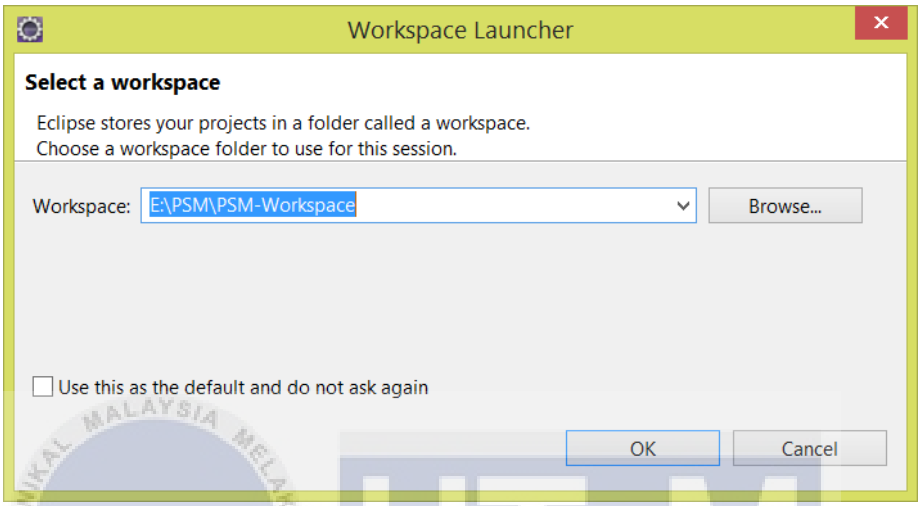


Figure 5.20: Server Installation Step 1

STEP 2: Click on File – New – Dynamic

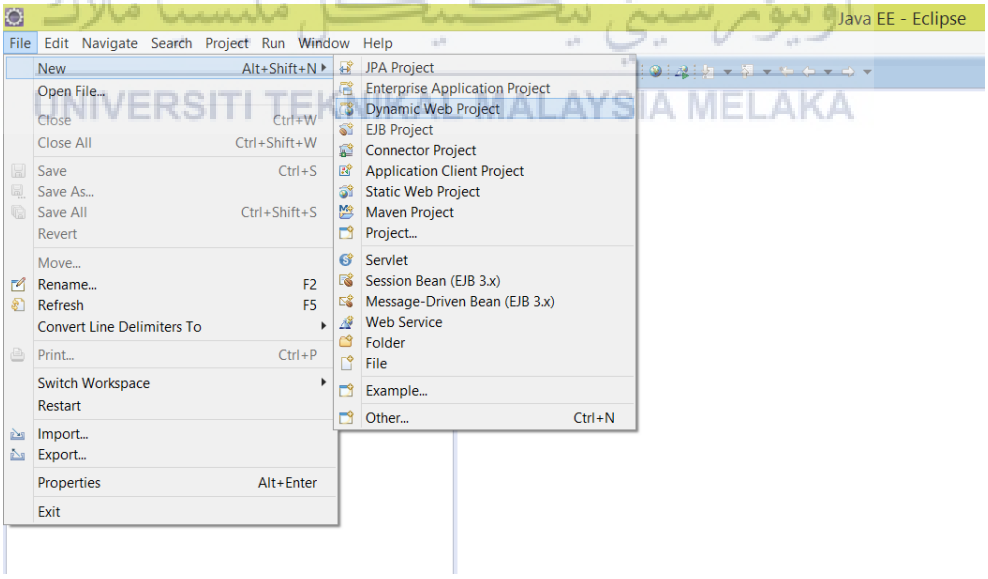


Figure 5.21: Server Installation Step 2

STEP 3: Choose Apache Tomcat v8.0 and click next button.

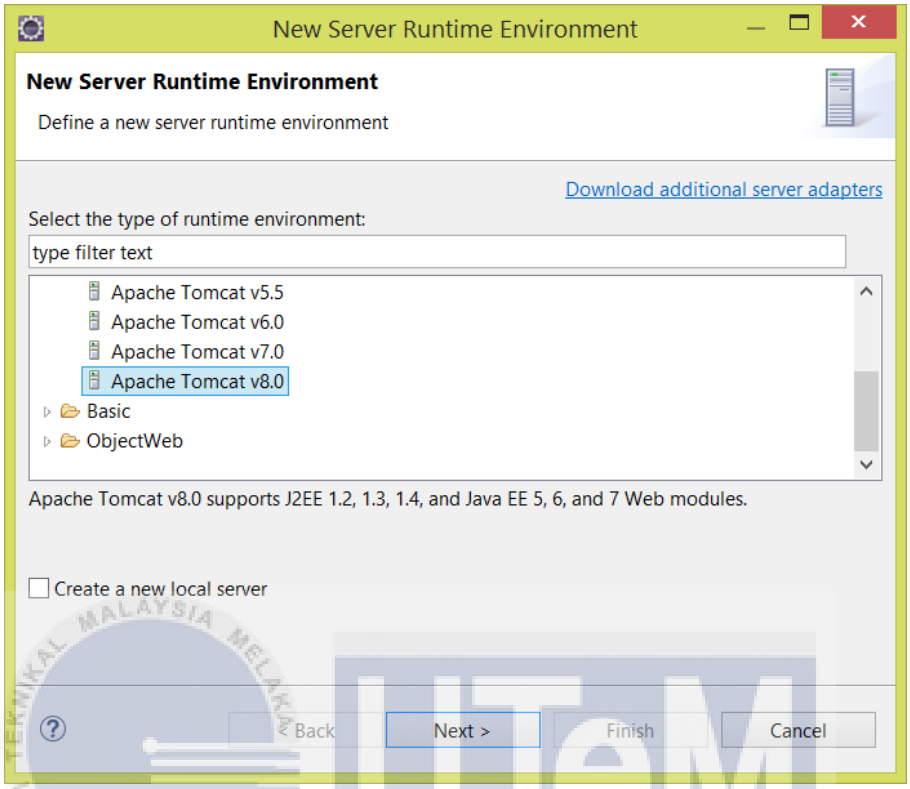


Figure 5.22: Server Installation Step 3

STEP 4: Find the apache-tomcat-8.0.27 and click on OK button.

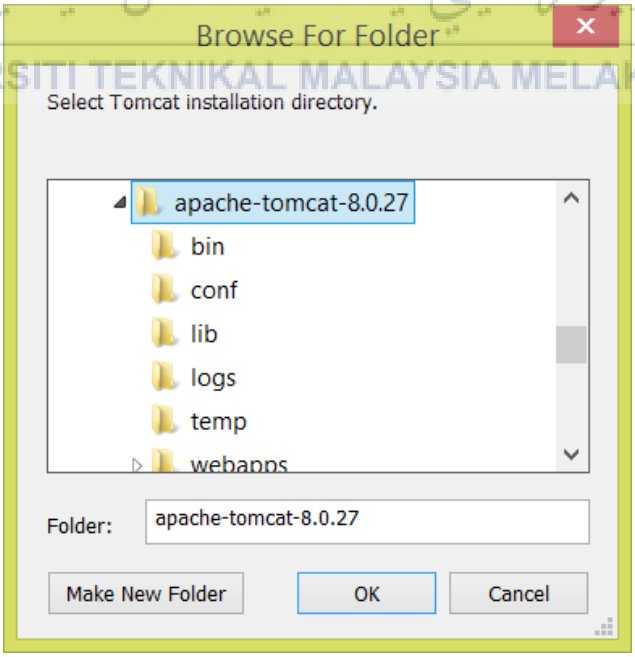


Figure 5.23: Server Installation Step 4

STEP 5: Click Finish Button

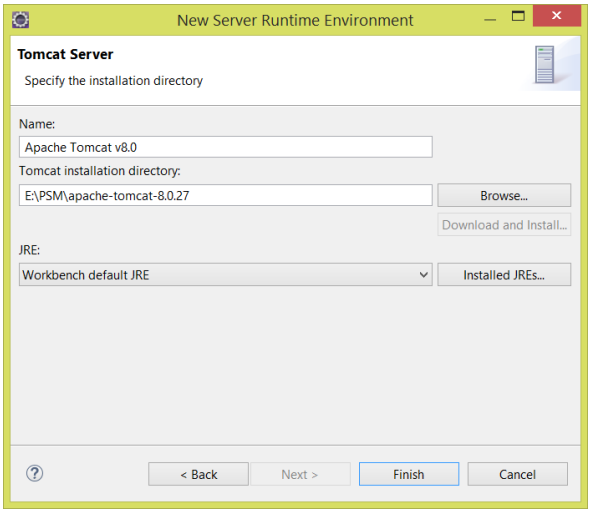


Figure 5.24: Server Installation Step 5

STEP 6: Enter the project name and tick Use default location and click on Next button

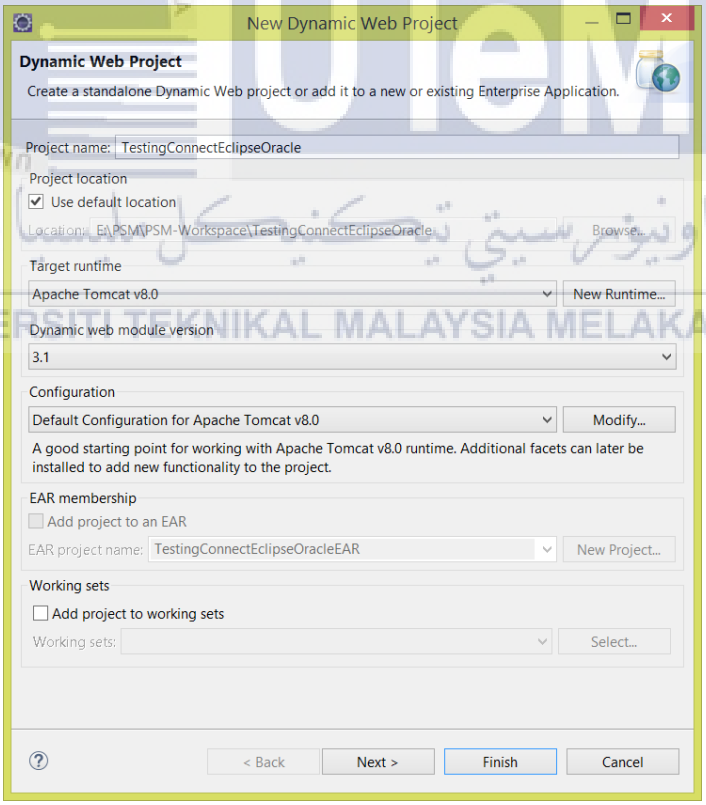


Figure 5.25: Server Installation Step 6

STEP 7: Click on FInish Button

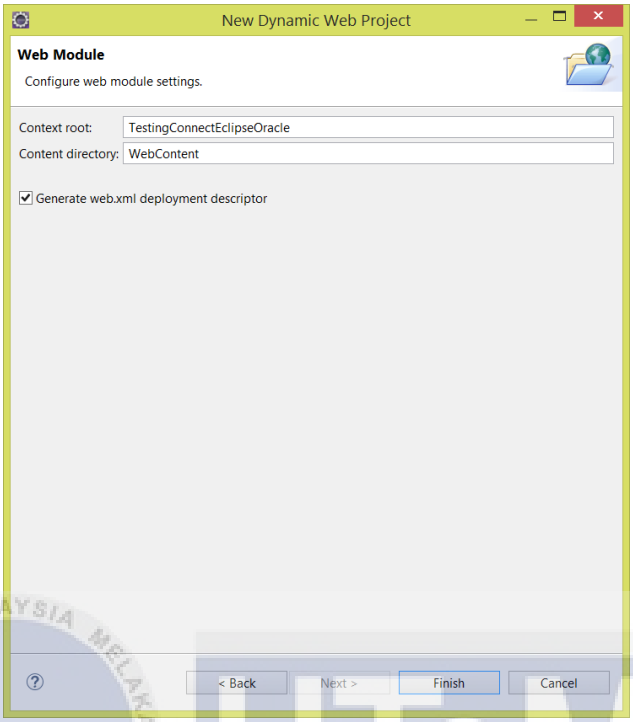


Figure 5.26: Server Installation Step 7

STEP 8: Click on Server and click the blue text to create server.

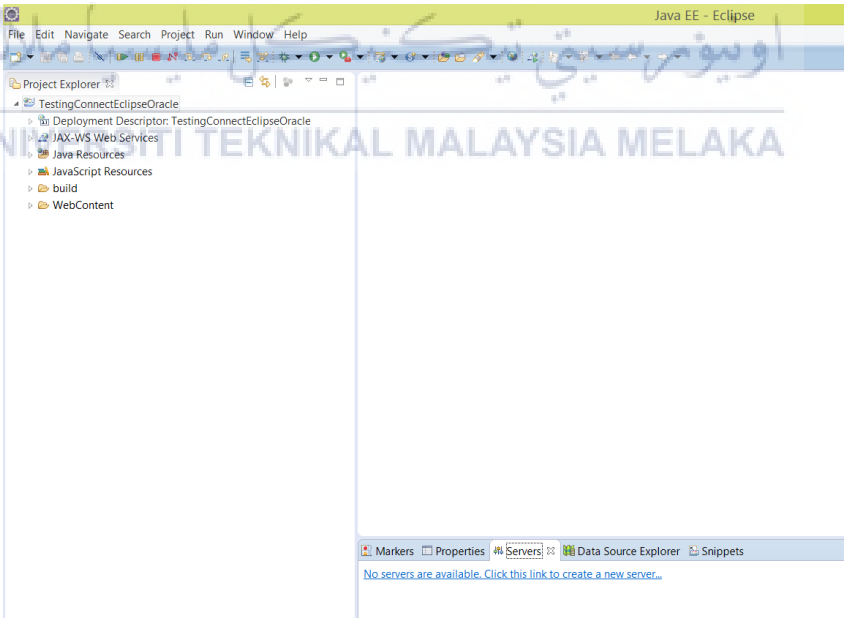


Figure 5.27: Server Installation Step 8

STEP 9: Enter the host name as localhost and Server runtime as Apache Tomcat v8.0 and click Next button

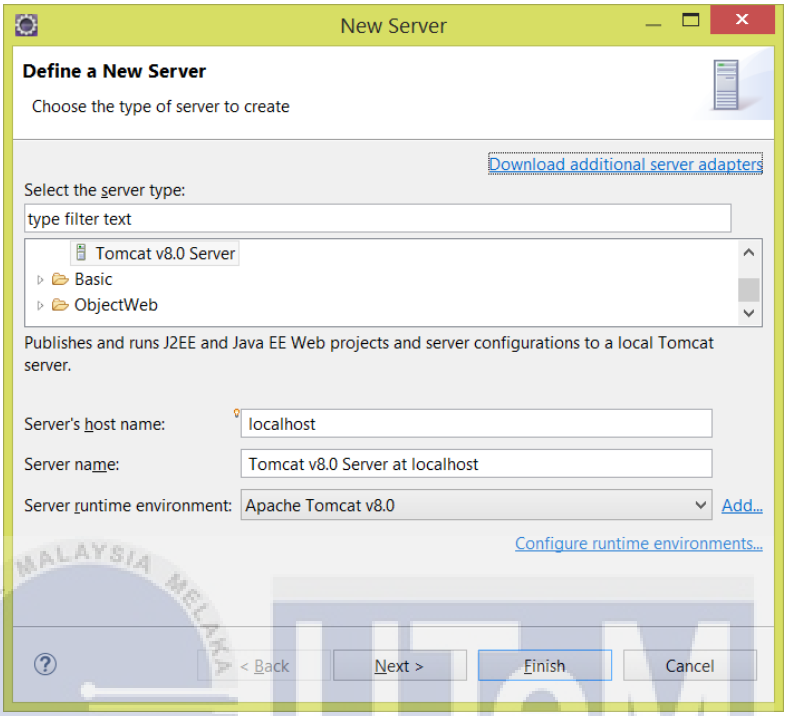


Figure 5.28: Server Installation Step 9

STEP 10: Click Add all Button and click Finish button.

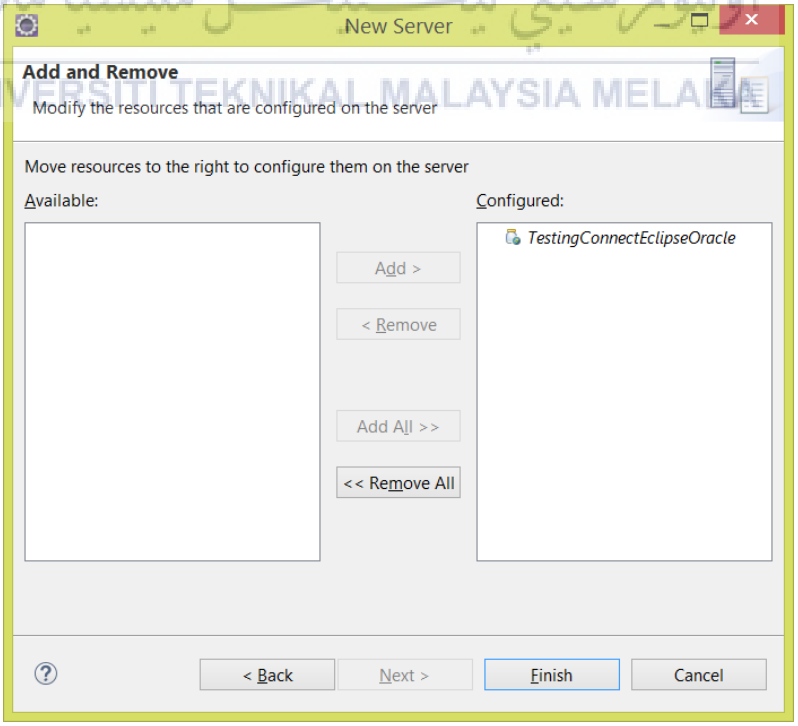


Figure 5.29: Server Installation Step 10

STEP 11: Click on server and click the green button to start the server.

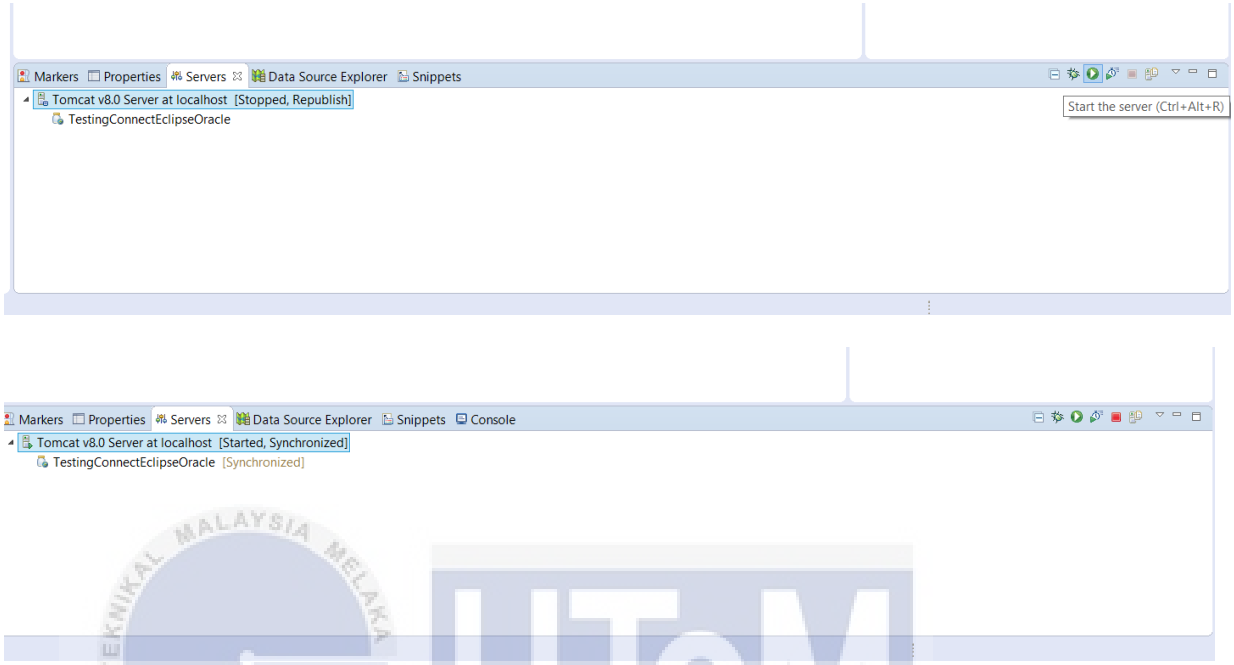


Figure 5.30: Server Installation Step 11

STEP 12: Right click on the project and Run As – Run on Server

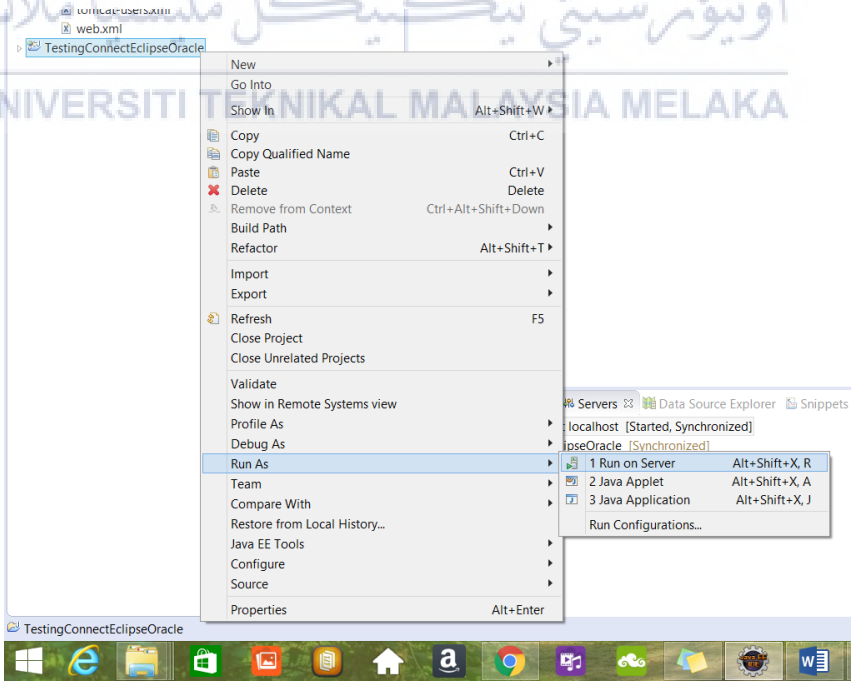


Figure 5.31: Server Installation Step 12

STEP 13: Open any web browser and go to localhost:8080 to test the server.

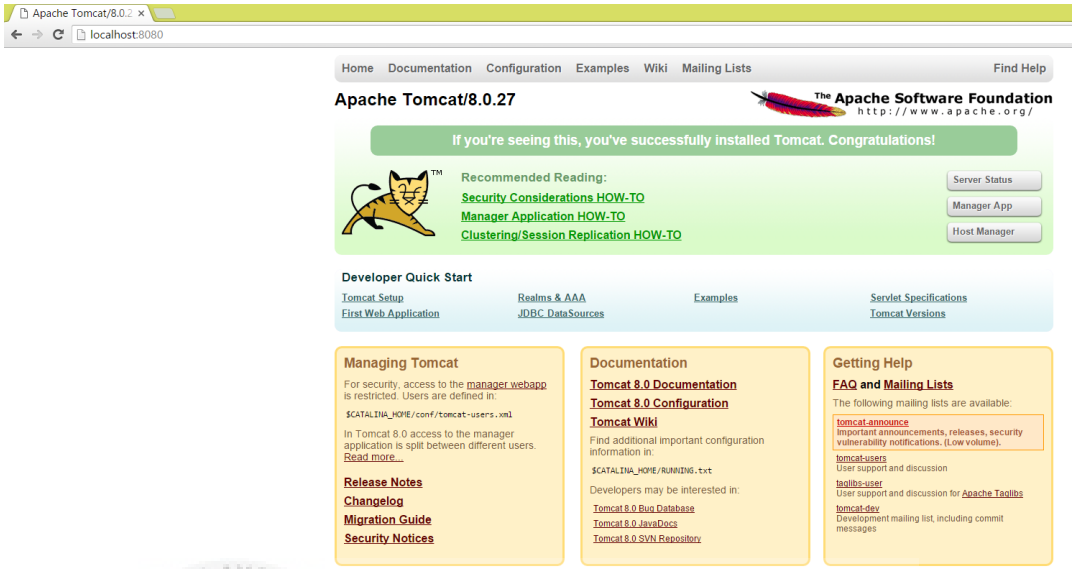


Figure 5.32: Server Installation Step 13

STEP 14: Right click the project and go to New- SQL file

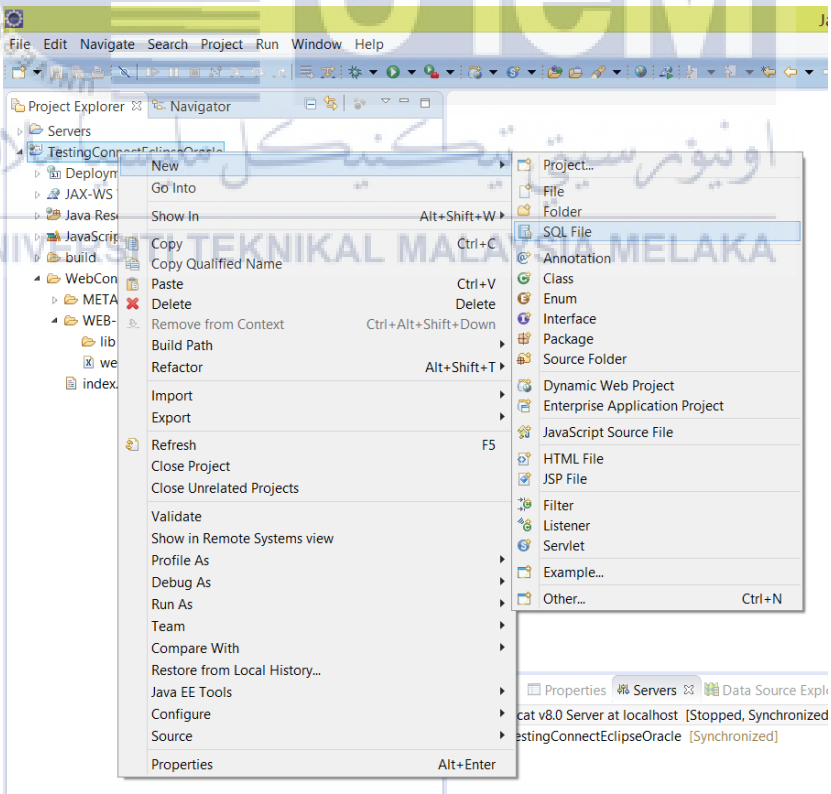


Figure 5.33: Server Installation Step 14

STEP 15: Click on Oracle and enter the name as DBOracle and click on Next button.

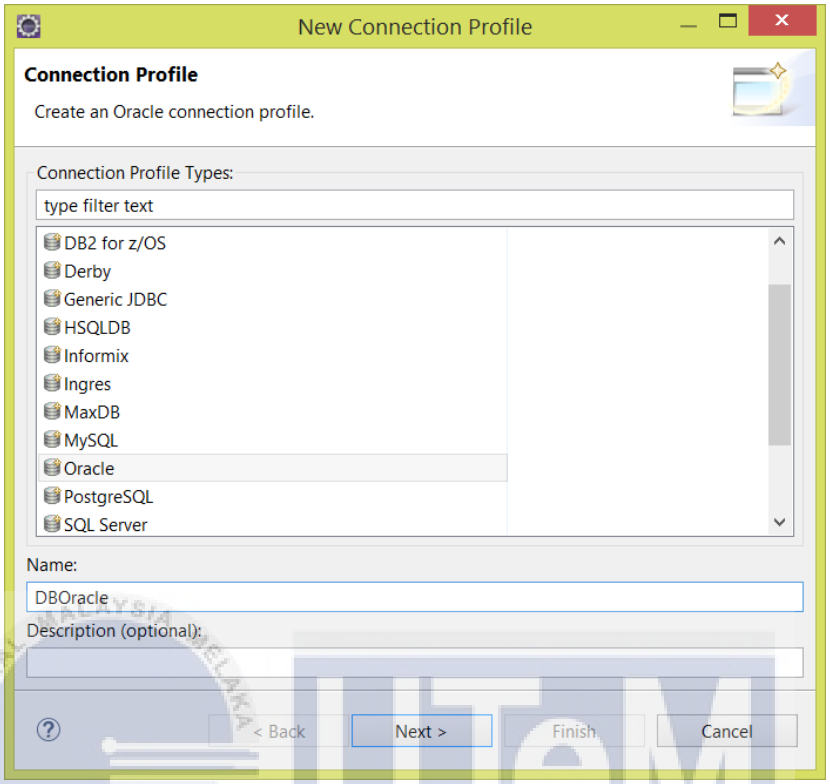


Figure 5.34: Server Installation Step 15

STEP 16: Click JAR List tab and click on Add JAR/Zip Button and find the driver in the file system and click Properties tab.

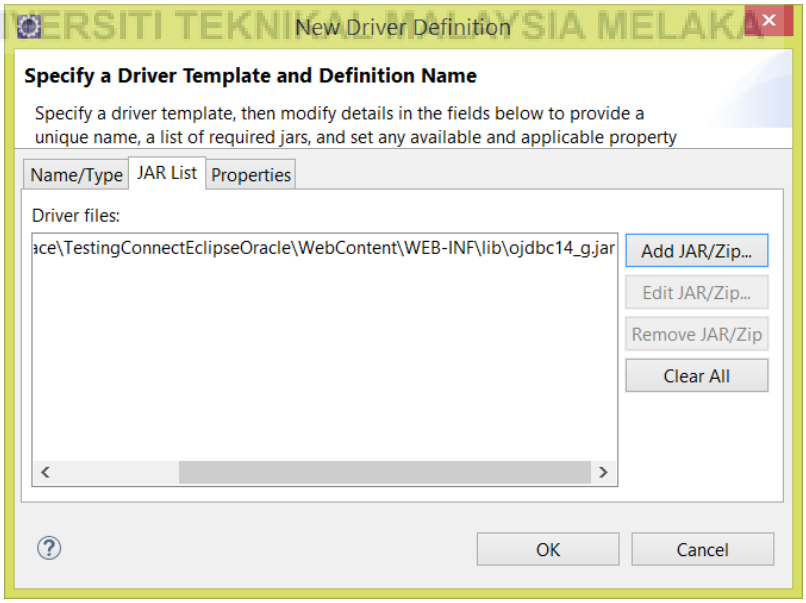


Figure 5.35: Server Installation Step 16

STEP 17: Edit the value as follow and click on OK button.

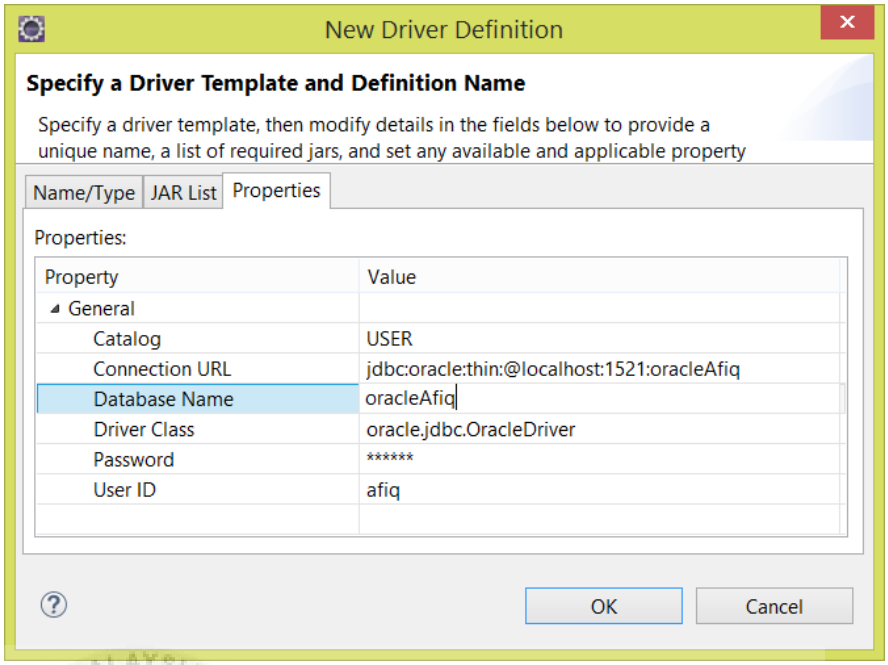


Figure 5.36: Server Installation Step 17

STEP 18: Click Test Connection button to test the connection with database.

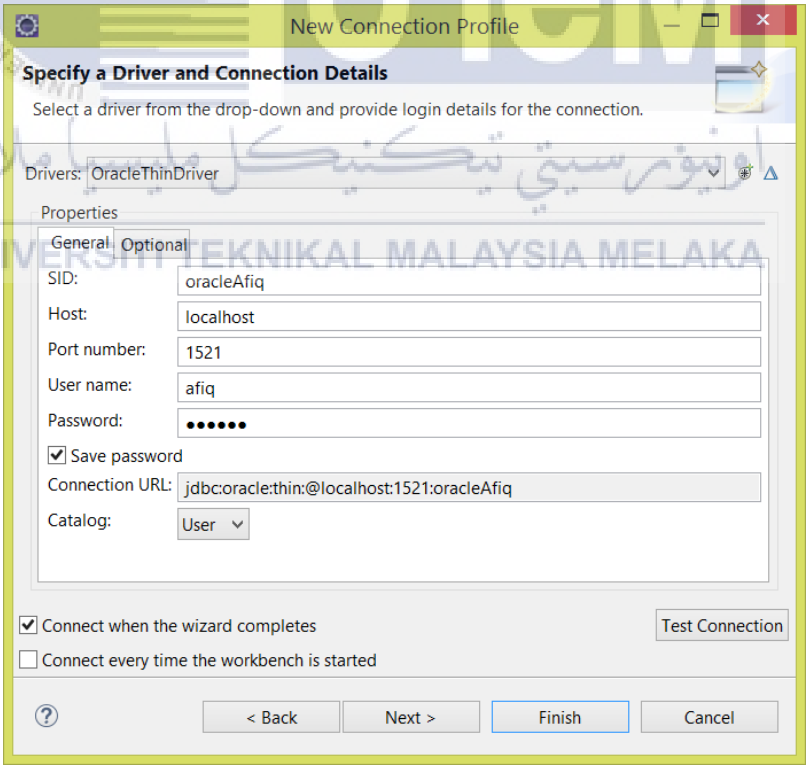


Figure 5.37: Server Installation Step 18

STEP 19: This popup will display if successful ping to the database and click Ok button to close

Figure 5.38: Server Installation Step 19

STEP 20: After all process successful, the summary of the connection will display and clock Finish to close.

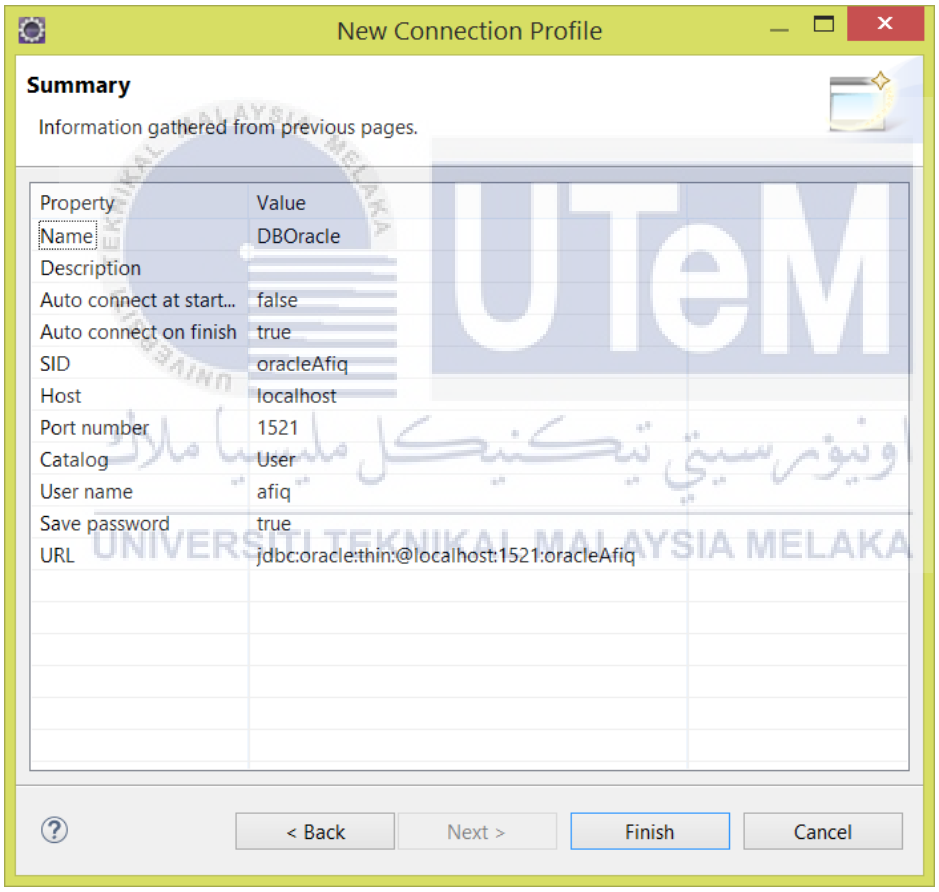


Figure 5.39: Server Installation Step 20

5.2.2 Database and database objects creation

Database for the i-Bank Questions (Question Generator) are create in the B031310227 database. The entire table that consist of table name, table column, data type, primary key, foreign key and are created in the database and using SQL statement.

```
CREATE TABLE TEACHER (
      TEACHER_ID varchar2(30) primary key NOT NULL,
       TEACHER_NAME varchar2(30),
       TEACHER_SCHOOL varchar2(30),
       PHONE_NO varchar2(30),
       AGE varchar2(30));
```

Figure 5.40: Create table Teacher

```
CREATE TABLE QUESTION_SUB(
       QUESTION_SUB_ID varchar2(30) primary key NOT NULL,
       QUESTION varchar2(300),
       TOTAL_DOWNLOAD INT,
       STATUS varchar2(30),
       QS_DATE DATE,
       TEACHER_ID varchar2(30) references TEACHER(TEACHER_ID),
       LEVEL_ID varchar2(30) references LEVEL_QUEST(LEVEL_ID),
       TOPIC_ID varchar2(30) references SUB_TOPIC(TOPIC_ID));
```

Figure 5.41: Create table Question_sub

```
CREATE TABLE QUESTION_OBJ(
       QUESTION_OBJ_ID varchar2(30) primary key NOT NULL,
       QUESTION varchar2(300),
       TOTAL_DOWNLOAD INT,
       STATUS varchar2(30),
       QO_DATE DATE,
       TEACHER_ID varchar2(30) references TEACHER(TEACHER_ID),
       LEVEL_ID varchar2(30) references LEVEL_QUEST(LEVEL_ID),
       TOPIC_ID varchar2(30) references SUB_TOPIC(TOPIC_ID));
```

Figure 5.42: Create table Question_obj

```
CREATE TABLE SUB_ANSWER
(
	SUB_ANS_ID varchar2(30) primary key NOT NULL,
	ANSWER varchar2(100),
	MARK NUMBER(5,2),
	QUESTION_SUB_ID varchar2(30) references
	QUESTION_SUB(QUESTION_SUB_ID)
);
```

Figure 5.43: Create table Sub_answer

```
CREATE TABLE OBJ_ANSWER
(
	OBJ_ANS_ID varchar2(30) primary key NOT NULL,
	ANSWER varchar2(100),
	TRUEFALSE varchar2(30),
	QUESTION_OBJ_ID varchar2(30) references
	QUESTION_OBJ(QUESTION_OBJ_ID)
);
```

Figure 5.44: Create table Obj_answer

```
CREATE TABLE UJIAN
(
	UJIAN_ID varchar2(30) primary key NOT NULL,
	TAJUK varchar2(30)
);
```

Figure 5.45: Create table Ujian


```
CREATE TABLE DOWNLOAD
(
	DOWNLOAD_ID varchar2(30) primary key NOT NULL,
	DW_DATE DATE,
	TEACHER_ID varchar2(30) references TEACHER(TEACHER_ID),
	UJIAN_ID varchar2(30) references UJIAN(UJIAN_ID)
);
```

Figure 5.46: Create table Download

```
CREATE TABLE DOWNLOAD_SUB
(
	DOWNLOAD_SUB_ID varchar2(30) primary key NOT NULL,
	LEVEL_QUEST varchar2(30),
	SUB_TOPIC varchar2(30),
	CHAPTER varchar2(30),
	SUBJECT varchar2(30),
	FORM varchar2(30),
	CATEGORY varchar2(30),
	DOWNLOAD_ID varchar2(30) references DOWNLOAD(DOWNLOAD_ID),
	QUESTION_SUB_ID varchar2(30) references
	QUESTION_SUB(QUESTION_SUB_ID)
);
```

Figure 5.47: Create table Download_sub

```
CREATE TABLE DOWNLOAD_OBJ
(
	DOWNLOAD_OBJ_ID varchar2(30) primary key NOT NULL,
	LEVEL_QUEST varchar2(30),
	SUB_TOPIC varchar2(30),
	CHAPTER varchar2(30),
	SUBJECT varchar2(30),
	FORM varchar2(30),
	CATEGORY varchar2(30),
	DOWNLOAD_ID varchar2(30) references DOWNLOAD(DOWNLOAD_ID),
	QUESTION_OBJ_ID varchar2(30) references
	QUESTION_OBJ(QUESTION_OBJ_ID)
);
```


Figure 5.48: Create table Download_obj

5.3 Database Implementation

The processes of the i-Bank Questions (Question Generator) are using stored procedure for insert, update, delete and view. The purpose of the stored procedure is to secure the source code from other user. Trigger that used in the system for automatically insert the data then meet some condition in the trigger. The triggers that involve in the system are trigger before insert, after insert and before delete.

a) Stored procedure

```
create or replace procedure loginshahmi(
view_teacher_id in teacher.teacher_id%type,
view_teachername in teacher.teacher_name%type,
view_a out int
)
is
begin
select count(*) into view_a
from teacher
where teacher_id = view_teacher_id and teacher_name=view_teachername;
end;
```

Figure 5.49: Stored Procedure for login


```
create or replace procedure insertujian(
p_tajuk in ujian.tajuk%type)
is
begin
insert into ujian (tajuk)
values (p_tajuk);
commit;
end;
```

Figure 5.50: Stored Procedure for insert

```
create or replace procedure deleteujian(p_ujian_id in ujian.ujian_id%type)
is
begin
delete ujian where ujian_id = p_ujian_id;
commit;
end;
```

Figure 5.51: Stored Procedure for delete

```
create or replace procedure ujianbyid(id varchar2,ujiancur out sys_refcursor)
as
selectu varchar2(1000);
fromu varchar2(1000);
whereu varchar2(1000);
sqlu varchar2(1000);
dateu varchar2(1000);

begin

dateu := 'dd/mon/yyyy  hh:mi:ss';
selectu := 'select to_char(d.dw_date, ''' || dateu || '''),u.tajuk,u.ujian_id ';
fromu := 'from teacher t,download d,ujian u ';
whereu := 'where t.teacher_id=d.teacher_id and d.ujian_id=u.ujian_id and
t.teacher_id = ''' || id || ''' order by d.dw_date desc';

sqlu := selectu || fromu || whereu;
open ujiancur for sqlu;
end;
/
```


Figure 5.52: Stored Procedure for view

b) Trigger

```
create or replace trigger delete_ujian_trig
before delete on ujian
for each row
begin
delete from download where ujian_id = :old.ujian_id;
end;
```

Figure 5.53: Trigger before delete

```
create or replace trigger trig_download_sub
before insert on download_sub
for each row

declare

pk_dwl number;
a varchar2(50);
dlevel_quest varchar2(50);
dsub_topic varchar2(50);
dchapter varchar2(50);
dsubject varchar2(50);
dform varchar2(50);
dcategory varchar2(50);
dwn_id varchar2(50);

begin

select download_sub_seq.nextval into pk_dwl from dual;
:new.download_sub_id :='ds'|| pk_dwl;
a:=:new.question_sub_id;

select l.lvl,t.topic_name,c.chap_name,sb.sub_name,f.form,sb.category
into dlevel_quest,dsub_topic,dchapter,dsubject,dform,dcategory
from  question_sub q, level_quest l, sub_topic t, chapter c, subject_form sf, subject
sb, form f
where q.level_id = l.level_id
and q.topic_id = t.topic_id
and t.chapter_id = c.chapter_id
and c.sub_form_id = sf.sub_form_id
and sf.subject_id = sb.subject_id
and sf.form_id = f.form_id
and q.question_sub_id=a;

:new.level_quest := dlevel_quest;
:new.sub_topic := dsub_topic;
:new.chapter := dchapter;
:new.subject := dsubject;
:new.form := dform;
:new.category := dcategory;

select max(download_id) into dwn_id from download;
:new.download_id :=dwn_id;

end;
/
```

Figure 5.54: Trigger before insert

```
create or replace trigger trigg_download_sub
after insert on download_sub
for each row
begin
update question_sub
set
total_download = nvl(total_download,0) + 1
where question_sub_id = :new.question_sub_id;
end;
```

Figure 5.55: Trigger after insert

5.4 Conclusion

In this chapter, it discuss all the activity about implementation of the system include the installation database and server. It includes the detail about creating database and database object. Next chapter will cover on testing phase, overall system will be tested to evaluate the capability of the i-Bank Questions (Question Generator).

CHAPTER VI

TESTING

6.1 Introduction

This chapter will discuss about testing phase and the results have been carried out on the i-Bank Questions (Question Generator). This phase will discuss to determine the functionality of the system either it meet the entire requirement or not. This is also to ensure the system effectively and to detect any failure or bug and any fault. This chapter will cover more about test plan, test strategy, test strategy and test result and analysis. The objective of the test is to ensure the functioning of the system, detect the error occurred and studies the risk.

6.2 Test Plan

Test plan divided in organization testing and environment testing, it's like alpha testing and beta testing. Test plan also include the testing schedule. The content is the set of test input, implementation environment and result of testing.

6.2.1 Test Organization

Test organization involving a group of people that responsibility to test the system during the testing phase. These groups come from different background which is system developer, teacher and student. System developer is the person who develops the i-Bank Questions (Question Generator). Teacher act as the main user of the system and student is second user. Their responsibilities during testing plan are as follows:

i. System developer

Responsibilities: Develop, manage and testing the system. They ensure that the system meet the entire requirement and will run smoothly before it will publish.

ii. Teacher

Responsibilities: Test the system by generates the question with answer scheme and manage their download. They will give feedback to the system developer

iii. Student

Responsibilities: Test the system by generates the question and manages their download. Their feedback can be as guide to enhance the system.

6.2.2 Test Environment

Test environment is the platform that system developer uses to develop the i-Bank Questions (Question Generator) run smoothly and successful. The environment was setup base on the following specification.

Table 6.1: Test Environment

Environment Specification	Description
Hardware	Processor : Intel Pentium RAM : 2GB Hard Disk space : 300 MB and above
Database	Oracle 11g
Server	Apache Tomcat
Server-scripting	Java

6.2.3 Test Schedule

Table 6.2: Test Schedule for Teacher

Module/ Component	Activity	Duration	Start Date	End Date
System Login	Test login into the system using their id and password	1 day	15/6/2016	15/6/2016
Searching	Testing to generate question and answer scheme	2 day	15/6/2016	16/6/2016
Management	View their download and ranking.	2 day	15/6/2016	16/6/2016

Table 6.3: Test Schedule for Student

Module/ Component	Activity	Duration	Start Date	End Date
System Login	Test login into the system as student	1 day	15/6/2016	15/6/2016
Searching	Testing to find and generate question	2 days	15/6/2016	16/6/2016
Management	View their download and ranking.	2 days	15/6/2016	16/6/2016

6.3 Test Strategy

Test strategy describes the testing approach. The objective of this testing strategy is to inform the system developer and tester about any problem regarding the testing process. There are two type of testing method that involved in testing strategy:

i. Alpha testing

Alpha testing conducted base on the testing requirement and functionality. The purpose is to improve the system to find any bug or error while users insert false data.

ii. Beta testing

Beta testing come after alpha testing, it is the methods of testing the software application to ensure the functionality safe from any faults or bugs. Beta versions probably launch to open public to increase the feedback from users.

6.3.1 Classes of tests

These are the some classes of the test that have been carried out. It is divided into security testing, error handling, output correctness test and user acceptance test.

i. Security test

Security testing can protect the system data and maintain the functionality as intended. For example before the data stored in the database, there is high probability of being stolen data. The security of data in the system will be protected by using ID and password.

ii. Error handling

Error handling testing is to ensure the i-Bank Questions (Question Generator) only accept correct input from user. The error massage will display if any error data or unfilled required data from user inserted. The system will block the data from stored into the database.

iii. Output correctness

The output correctness testing is to ensure the input and output meets the requirements that guided and related to the system.

iv. User acceptance

The purpose of user acceptance testing is to ensure the system is user friendly to use. The user interface must interest to attract user. It's also must easy to use for teacher and student.

6.3.2 Test Description

Test description is the result of test process from the data that recorded in the database of the i-Bank Questions (Question Generator). Table 6.4 shows the expected result for each system modules.

Table 6.4: Login Module

Test Case ID	Description	Testing Type	Expected Result
TC01-1	Invalid user ID and invalid password	Unit testing	'Invalid Username or Password' message will appear.
TC01-2	User ID and blank password	Unit testing	'Please fill out this field' message will appear.
TC01-3	Valid user ID and password	Unit testing	Successfully login and go to main page

Table 6.5: Download Question Module

Test Case	Description	Testing Type	Expected Result
TC02-1	All fields blank	Unit testing	'Please fill out this field' message will appear.
TC02-2	Invalid input entered	Unit testing	The result will display but will ignore the selection of the invalid input.
TC02-3	Valid input for each field	Unit testing	The result will display.

6.3.3 Test Data

Testing data will discuss about the expected result when data was entered by user. Table below shows the test data that is used in the system.

Table 6.6: Test Data for Login

Column Name	TD01-1	TD01-2	TD01-3
Test Case ID	TC01-1	TC01-2	TC01-3
User ID	Zaims		Zain
Password	T45		T03
Result Test Data	Login failed. User ID and Password not match with database.	Login failed. 'Please fill out this field' message will appear.	User login successfully.

Table 6.7: Test Data for Download Question

Column Name	TD02-1	TD02-2	TD02-3
Test Case ID	TC02-1	TC02-2	TC02-3
Ujian	Homework		Homework
Subject	Kimia	Kimia	Kimia
Chapter	Molecule	Molecule	
Topic	Gases	Gases	
Category	Smjk	Smjk	
Form	TIngkatan 1	TIngkatan 1	TIngkatan 1
Level	Knowledge	Knowledge	
Jumlah Soalan	10		10
Result Test Data	Question generated and data of download detail inserted into	'Please fill out this field' message will appear and question will not	Question generated with selected category of question and data

	database.	display and data not inserted.	of download detail inserted into database.

6.4 Test Results and Analysis

Test result is the review of the result after testing case is input to test the system. The success or failure when input data during the testing process will display and can measure the system either it can work efficiently or need to fixed or upgrade. The table below shows the test result and analysis.

Table 6.8: Test Result and Analysis for Login

Module Component: login		**Result**		
Test Case ID	**Test Data ID**	**Description**	**Pass**	**Fail**
TC01-1	TD01-1	User ID and password didn't exist	√	
TC01-2	TD01-2	User ID or password field blank	√	
TC01-3	TD01-3	Valid user ID and password	√	

Table 6.9: Test Result and Analysis for Download Question

Module Component: login		**Result**		
Test Case ID	**Test Data ID**	**Description**	**Pass**	**Fail**
TC02-1	TD02-1	All field valid data.	√	
TC02-2	TD02-2	Invalid data or format in download fields.	√	
TC02-3	TD02-3	Valid data for required field.	√	

6.5 Conclusion

As the conclusion, the testing process successful finish and meet all the requirement of the system. The testing process conducted every single part of the system to identify the errors and bugs to avoid any software defects before the system will publish to users in order to achieve objective of the project. From this test, developer can fixed any bugs and error that come up. Next chapter will cover on conclusion of the project that include strength and weakness of the project and the improvement.

CHAPTER VII

CONCLUSION

7.1 Introduction

At the end, this chapter will discuss about the observation on weaknesses and strengths and some suggestion how to improve the system. The weakness can be improved to make the system better in the future. All suggestion from outsider will take into consideration. Finally, the contribution of the project to the university or faculty and individual that will use this system.

7.2 Observation on Weaknesses and Strengths

After done developing the i-Bank Questions (Question Generator) system, some strength and weakness was observed. The weakness will take into consideration to make the system better in the future.

7.2.1 System Weaknesses

Vulnerability in the system is teacher or student cannot download back their history of the examination, just in case they deleted the PDF file. They need to download other question which is the question not same with previous one. It is because the system randomly generates the question.

7.2.2 System Strengths

The benefit of this system is teachers can share their question with other teachers and also to the student. This system also make teacher easy to find the question without need to open reference book to make the question manually. Teachers also can easily marking their student's paper because this system provides the question together with the answer scheme.

7.3 Propositions for Improvement

i-Bank Questions (Question Generator) need some improvement base on some advantage and disadvantage that found. To make the system better, it must give the teachers permission to change or remove the question when they found uninterested question and can replace with other related question.

Secondly, to make the system interesting, it must provide the online examination or quiz that student can answers directly from the page without print the question. After finish answering, the marks for the student will be calculate and display automatically.

7.4 Contribution

As the contribution to the teacher and student, this system can be an additional system for any school and very helpful which has an ability to find any question base on any subject and topic. It also will reduce of time for teacher to marking question because it comes with answers scheme and possible mark. Other than that, student can easily do some exercise by downloading question from i-Bank Questions (Question Generator).

7.5 Conclusion

As the conclusion, the i-Bank Questions (Question Generator) is successfully complete according to the objective and scope. This system was made to help teacher and student to generate any question. However, this system still needs some improvement to make it more interesting and helpful.

REFERENCES

Jeffrey A. Hoffer, Joey F. George, Joseph S. Valacich (1999). *Modern Systems Analysis & Design Second Edition.* Addison-Wesley, Retrieved March 17, 2016

Dennis, A., Wixom, B.H., Tegarden, D. (2005). *System Analysis and Design with UML Version 2.0, 2nd Ed.* John Wiley & Sons, Inc, Retrieved June 25, 2016

Sasbadi Bank Soalan Website on March 4, 2016 from http://banksoalan.sasbadionline.com/

How to sort answers and question in JSP, Retrieved June 25, 2016 http://stackoverflow.com/questions/37207232/how-can-i-make-as-proper-qa-in-jsp-from-this-table

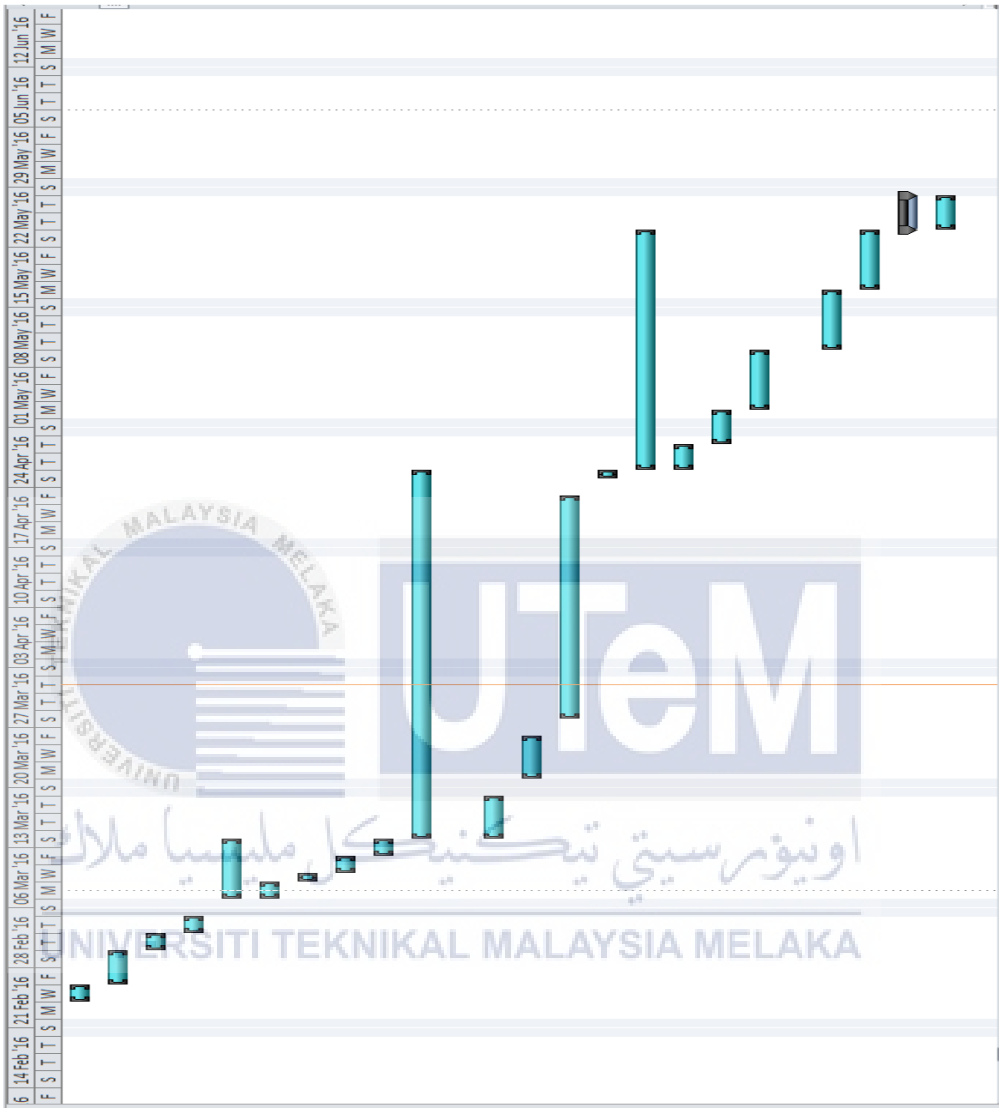
Servlet oracle database connection example, Retrieved March 30, 2016 http://stackoverflow.com/questions/28321816/servlet-oracle-database-connection-example

APPENDICES

APPENDIX A

Gantt Chart

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1		**Phase 1: Initial Study**	10 days	Mon 22-02-16	Fri 04-03-16		
2		Interview /Business Rules	2 days	Mon 22-02-16	Tue 23-02-16		
3		Identify problem and constraint	2 days	Wed 24-02-16	Thu 25-02-16		
4		Define Objective	2 days	Fri 26-02-16	Mon 29-02-16		
5		Define Scope and Boundries	2 days	Tue 01-03-16	Wed 02-03-16		
6		Write Document	2 days	Thu 03-03-16	Fri 04-03-16		
7		**Phase 2: Database Design**	6 days	Mon 07-03-16	Sun 13-03-16		
8		Create Conceptual Design	2 days	Mon 07-03-16	Tue 08-03-16		
9		DBMS Software Selection	1 day	Wed 09-03-16	Wed 09-03-16		
10		Create Logical Design	2 days	Thu 10-03-16	Fri 11-03-16		
11		Create Physical Design	2 days	Sat 12-03-16	Sun 13-03-16		
12		**Phase 3: Implementation and Loading**	31 days	Mon 14-03-16	Mon 25-04-16		
13		Install the DBMS	5 days	Mon 14-03-16	Fri 18-03-16		
14		Create the Database	5 days	Mon 21-03-16	Fri 25-03-16		
15		Load and Convert the Data	20 days	Mon 28-03-16	Fri 22-04-16		
16		Project Demo and Chapter 3	1 day	Mon 25-04-16	Mon 25-04-16		
17		**Phase 4: Testing and Evaluation**	20 days	Tue 26-04-16	Mon 23-05-16		
18		Test the Database	3 days	Tue 26-04-16	Thu 28-04-16		
19		Fine Tune the Database	2 days	Fri 29-04-16	Mon 02-05-16		
20		Evaluate the Database and Application	5 days	Tue 03-05-16	Mon 09-05-16		
21		Project Demo and Chapter4	5 days	Tue 10-05-16	Mon 16-05-16		
22		Project Demo and Report PSM	5 days	Tue 17-05-16	Mon 23-05-16		
23		**Phase 5: Operation**	**4 days**	**Tue 24-05-16**	**Fri 27-05-16**		
24		Produce the require information flow	4 days	Tue 24-05-16	Fri 27-05-16		
25		***Phase 6 : Maintanance and Evaluation***	6 days	Mon 30-05-16	Mon 06-06-16		
26		Introduce the change	1 day	Mon 30-05-16	Mon 30-05-16		
27		Make enhancement	2 days	Tue 31-05-16	Wed 01-06-16		
28		Backup and Recovery	2 days	Thu 02-06-16	Fri 03-06-16		
29		Final presentation	1 day	Mon 06-06-16	Mon 06-06-16		



APPENDIX B

TRIGGER AND STORED PROCEDURE CODING FOR I-BANK QUESTIONS (QUESTION GENERATOR)

TRIGGER BEFORE INSERT TABLE DOWNLOAD_SUB

```
create sequence download_sub_seq
start with 1
increment by 1;
create or replace trigger trig_download_sub
before insert on download_sub
for each row
declare
pk_dwl number;
a varchar2(50);
dlevel_quest varchar2(50);
dsub_topic varchar2(50);
dchapter varchar2(50);
dsubject varchar2(50);
dform varchar2(50);
dcategory varchar2(50);
dwn_id varchar2(50);
begin
select download_sub_seq.nextval into pk_dwl from dual;
:new.download_sub_id :='ds'|| pk_dwl;
a:=:new.question_sub_id;
select l.lvl,t.topic_name,c.chap_name,sb.sub_name,f.form,sb.category
into dlevel_quest, dsub_topic, dchapter, dsubject, dform, dcategory
from  question_sub q, level_quest l, sub_topic t, chapter c, subject_form sf,
subject sb, form f
where q.level_id = l.level_id
and q.topic_id = t.topic_id
and t.chapter_id = c.chapter_id
and c.sub_form_id = sf.sub_form_id
and sf.subject_id = sb.subject_id
and sf.form_id = f.form_id
and q.question_sub_id=a;
:new.level_quest := dlevel_quest;
:new.sub_topic := dsub_topic;
:new.chapter := dchapter;
:new.subject := dsubject;
:new.form := dform;
:new.category := dcategory;
select max(download_id) into dwn_id from download;
:new.download_id :=dwn_id;
end;
```

TRIGGER BEFORE INSERT TABLE UJIAN

```
create sequence ujian_seq
start with 1
increment by 1;

create or replace trigger trig_ujian_pk
before insert on ujian
for each row
declare
pk_dwl number;
begin
select ujian_seq.nextval into pk_dwl from dual;
:new.ujian_id :='uji'|| pk_dwl;
end;
```

TRIGGER BEFORE INSERT TABLE DOWNLOAD

```
create sequence download_seq
start with 1
increment by 1;

create or replace trigger trig_download
before insert on download
for each row
declare
pk_dwl number;
date_dwl date;
idujian varchar2(50);
begin
select download_seq.nextval into pk_dwl from dual;
:new.download_id :='dw'|| pk_dwl;
:new.dw_date:=sysdate;
select max(ujian_id) into idujian from ujian;
:new.ujian_id :=idujian;
end;
/
```

TRIGGER BEFORE DELETE TABLE UJIAN

```
create or replace trigger delete_ujian_trig
before delete on ujian
for each row
begin
delete from download where ujian_id = :old.ujian_id;
end;
```

TRIGGER BEFORE DELETE TABLE DOWLOAD

```
create or replace trigger delete_download_trig
before delete on download
for each row
begin
delete from download_obj where download_id = :old.download_id;
delete from download_sub where download_id = :old.download_id;
end;
```

TRIGGER AFTER INSERT TABLE ON DOWNLOAD_SUB

```
create or replace trigger trigg_download_sub
after insert on download_sub
for each row
begin
update question_sub
set
total_download = nvl(total_download,0) + 1
where question_sub_id = :new.question_sub_id;
end;
```

TRIGGER AFTER INSERT TABLE ON DOWNLOAD_OBJ

```
create or replace trigger trigg_download_obj
after insert on download_obj
for each row
begin
update question_obj
set
total_download = nvl(total_download,0) + 1
where question_obj_id = :new.question_obj_id;
end;
```

PROCEDURE INSERT TABLE DOWNLOAD_OBJ

```
create or replace procedure insertdownobj(
p_question_obj_id in download_obj.question_obj_id%type)
is
begin
 insert into download_obj (question_obj_id)
values (p_question_obj_id);
commit;
end;
```

PROCEDURE INSERT TABLE DOWNLOAD_SUB

```
create or replace procedure insertdownsub(
p_question_sub_id in download_sub.question_sub_id%type)
is
begin
insert into download_sub (question_sub_id)
values (p_question_sub_id);
commit;
end;
```


PROCEDURE FOR LOGIN

```
create or replace procedure loginshahmi(
view_teacher_id in teacher.teacher_id%type,
view_teachername in teacher.teacher_name%type,
view_a out int
)
is
begin
select count(*) into view_a
from teacher
where teacher_id = view_teacher_id and teacher_name=view_teachername;
commit;
end;
```

PROCEDURE FOR VIEW RANKING

```
create or replace
procedure pro_tot_objsub (p_objsubtot out sys_refcursor) as
begin
open p_objsubtot for
select t.teacher_name, count(o.download_obj_id) objektif,
count(s.download_sub_id) subjektif
from teacher t
right outer join download d on t.teacher_id = d.teacher_id
full outer join download_obj o on o.download_id = d.download_id
full outer join download_sub s on s.download_id = d.download_id
group by t.teacher_name
order by subjektif desc,subjektif desc;
end;
```

PROCEDURE FOR VIEW DOWNLOAD

```
create or replace
procedure rec (p_recordset out sys_refcursor) as
begin
open p_recordset for
select  t.teacher_name, t.teacher_school,to_char
(d.dw_date, 'dd/mon/yyyy hh:mi:ss'),u.tajuk
from teacher t,download d,ujian u
where t.teacher_id=d.teacher_id
and d.ujian_id=u.ujian_id
order by d.dw_date desc;
end;
```


PROCEDURE FOR TOTAL DOWNLOAD

```
create or replace
procedure pro_totalujian (p_ujian out sys_refcursor) as
begin
open p_ujian for
select t.teacher_name, count(d.download_id) jumlahdownload
from teacher t
full outer join download d on t.teacher_id = d.teacher_id
group by t.teacher_name;
end;
```