FTMK INTERNSHIP PLACEMENT SYSTEM

AMEERUL ZAKI BIN AZLAN RAOUS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FTMK INTERNSHIP PLACEMENT SYSTEM

AMEERUL ZAKI BIN AZLAN RAOUS

This report is submitted in partial fulfillment of the requirements for the Bachelor of [Computer Science (Software Development)] with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

FTMK INTERNSHIP PLACEMENT SYSTEM

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

STUDENT: ____________________________ Date: 10 September 2021

(AMEERUL ZAKI BIN AZLAN RAOUS)

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 (Software Development)] with Honours.

SUPERVISOR: ____________________________ Date: 10/09/2021

(MRS. ZARITA MOHD KOSNIN)

DEDICATION

This study is wholeheartedly dedicated to our beloved parents, who have been our source of inspiration and gave us strength when we thought of giving up, who continually provide their moral, spiritual, emotional, and financial support. To our brothers, relatives, mentor, friends, and classmates who shared their words of advice and encouragement to finish this project. And lastly, we dedicated this project to the Almighty God, thank you for the guidance, strength, power of mind, protection, skills, and for giving us a healthy life. All of these, we offer to you.

ACKNOWLEDGEMENTS

Alhamdulillah, praises and thanks to Allah because of His Almighty and His utmost blessings, I was able to finish this project within the time duration given.

I would like to thank Mrs. Zarita Mohd Kosnin for giving assistant to complete this project successfully. I sincerely appreciated the extraordinary effort that she gave to me. Thank you for the guidance and help throughout the project development. Without her help, I would have difficulties in completing my project.

I would also like to thank my beloved parents and brothers who have been giving me moral support from a far, prayers and motivation throughout my project. Thank you all for your loves.

Lastly to my beloved friends who are encouraged, guided, and inspired me. Without their patience, understanding, support and most of love, the completion of this work would not have been possible.

ABSTRACT

FTMK Internship Placement System (FIPS) is a system to help manage students for industrial training. The industrial training period is essential for UTeM student and staff, thus the FIPS will be able to help facilitate UTeM staff during the students industrial training period. The problem statement for this project is the student face difficulty to find the organization for industrial training that aligns with their study programme. The faculty coordinator and supervisor also have difficulty to monitor and track all the student internship application status and industrial training progress. The internship students having trouble for determining the organizational criteria required to undergo industrial training. To overcome these problems, the FIPS will be developed. The objective for this project is to develop web-based system as a platform for students to search and apply industrial training by referring to the requirements given by the organization and to allow internship committee to monitor and track student's internship application status and industrial training progress. Next is to design a solution that facilitate students, internship committee and organizations related to industrial training. Finally, to test the system for efficiency and user satisfaction in terms of usability by the students, internship committee and organizations. The FIPS consist of 5 users which are the system administrator, coordinator, supervisor, student, and organization. This project is using System Development Life Cycle (SDLC). The phases of the methodology are Requirements and Analysis, System and Software Design, Development and Implementation, and System Testing. FIPS will be developed using Yii2 Framework (PHP Framework), and MySQL for the Relational Database Management System (RDBMS). The testing is done using unit testing and user acceptance testing from organization and student.

10/09/2021

ABSTRAK

FTMK Internship Placement System (FIPS) adalah sistem untuk membantu menguruskan pelajar untuk latihan industri. Tempoh latihan industri sangat penting bagi pelajar dan staf UTeM, oleh itu FIPS akan dapat membantu memudahkan kakitangan UTeM semasa tempoh latihan industri pelajar. Penyataan masalah untuk projek ini adalah pelajar menghadapi kesukaran untuk mencari organisasi latihan industri yang sesuai dengan program pengajian mereka. Penyelaras dan penyelia fakulti juga menghadapi kesukaran untuk memantau dan mengesan semua status permohonan magang pelajar dan kemajuan latihan industri. Pelajar magang menghadapi masalah untuk menentukan kriteria organisasi yang diperlukan untuk menjalani latihan industri. Untuk mengatasi masalah ini, FIPS akan dibangunkan. Objektif untuk projek ini adalah untuk mengembangkan sistem berasaskan web sebagai platform bagi pelajar untuk mencari dan menerapkan latihan industri dengan merujuk kepada keperluan yang diberikan oleh organisasi dan untuk membolehkan jawatankuasa magang memantau dan mengesan status permohonan magang pelajar dan kemajuan latihan industri. Seterusnya adalah merancang penyelesaian yang memudahkan pelajar, jawatankuasa magang dan organisasi yang berkaitan dengan latihan industri. Akhirnya, untuk menguji sistem untuk kecekapan dan kepuasan pengguna dari segi kebolehgunaan oleh pelajar, jawatankuasa magang dan organisasi. FIPS terdiri daripada 5 pengguna iaitu pentadbir sistem, penyelaras, penyelia, pelajar, dan organisasi. Projek ini menggunakan System Development Life Cycle (SDLC). Fasa-fasa metodologi tersebut adalah Keperluan dan Analisis, Reka Bentuk dan Pembangunan Sistem dan Perisian, dan Pengujian Sistem. FIPS akan dibangunkan menggunakan Yii2 Framework (PHP Framework), dan MySQL untuk Relational Database Management System (RDBMS). Pengujian dilakukan menggunakan ujian unit dan ujian penerimaan pengguna dari organisasi dan pelajar.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

FTMK	**-**	**Fakulti Teknologi Maklumat dan Komunikasi**
FIPS	**-**	**FTMK Internship Placement System**
UTeM	**-**	**Universiti Teknikal Malaysia Melaka**
HTTP	**-**	**Hypertext Transfer Protocol**
HTML	**-**	**Hypertext Markup Language**
DBMS	**-**	**Database Management System**
RDBMS	**-**	**Relation Database Management System**
PHP	**-**	**PHP: Hypertext Preprocessor**
JS	**-**	**JavaScript**
AJAX	**-**	**Asynchronous JavaScript and XML**
XML	**-**	**Extensible Markup Language**
DOM	**-**	**Document Object Model**
MiCoST	**-**	**Melaka International College of Science and Technology**
IS	**-**	**Information System**
UTM	**-**	**Universiti Teknologi Malaysia**
ITS	**-**	**Industrial Training System**
UiTM	**-**	**Universiti Teknologi MARA**

STIS	**-**	**Student Internship Training Information System**
MARA	**-**	**Majlis Amanah Rakyat**
STIS	**-**	**Student Internship Training Information System**
SDLC	**-**	**System Development Life Cycle**
ERD	**-**	**Entity Relationship Diagram**
DDL	**-**	**Data Definition Language**
AMTIS	**-**	**Advance Management Technology in Intelligence System**
UAT	**-**	**User Acceptance Testing**

LIST OF ATTACHMENTS

CHAPTER 1: INTRODUCTION

This chapter introduces the overview of project background and rationale on the project entitled FTMK Internship Placement System. In this chapter, the project background and the problem statement are discussed further on the next section which are introduction, problem statement, objective, scope, project significance, expected output, and conclusion.

1.1 Introduction

An internship or industrial training is a period of work experience offered by an organization for a limited period. The internship program provides student an opportunity to expose themselves to the real work environment and link theory with practice and further gaining relevant skills and experience in particular field. Internship program is an essential process in all universities. Before the student graduate, they need to undergo the internship program. However, management of internship placement can be differing for each university.

FTMK Internship Placement System (FIPS) is a system to help manage students for industrial training. The industrial training period is essential for UTeM student and staff, thus the FIPS will be able to help facilitate UTeM staff during the students industrial training period.

By using this system, the student can get an internship placement in a short time. The registered companies will provide an internship plans for the students to choose from. This system facilitates the process of students eligible for industrial training to find a place to undergo such industrial training. This system can be used as a platform for industrial training supervisor to keep track and monitor the student internship application status.

1.2 Problem Statement

While the study on FTMK Internship Placement System implementation, only slightly specific attention considered to measure the successful or failure. The list of problem that have been discovered are:

- Student face difficulty to find the organization for industrial training that aligns with their study programme.
- Faculty coordinator and supervisor have difficulty to monitor and track all the student internship application status and industrial training progress.
- Internship students having trouble for determining the organizational criteria required to undergo industrial training.

1.3 Objective

This section explains the objective of developing the project that should be achievable at the end of this project:

- To develop web-based system as a platform for students to search and apply industrial training by referring to the requirements given by the organization and to allow internship committee to monitor and track student's internship application status and industrial training progress.
- To design a solution that facilitate students, internship committee and organizations related to industrial training.
- To test the system for efficiency and user satisfaction in terms of usability by the students, internship committee and organizations.

1.4 Scope

The FTMK Internship Placement System is focus for use in Universiti Teknikal Malaysia Melaka (UTeM).

Modules:

i. User Authentication Module

- Authenticate user, and authorize user based on access control.

ii. Management Module

- Manage user, department, programme, state, language, and session.

iii. Announcement Module

- Post announcement.

iv. Resume Module

- Generate resume from the system.

v. Internship Application Module

- Manage internship application, apply internship, compare organization, generate offer letter, and feedback form.

vi. Organization Profile Module

- Generate organization profile from the system.

vii. Internship Recruitment Module

- Advertise organization's requirements and review student for industrial training.

viii. Supervision Module

- Manage and assign supervisor with student.
- View assigned students.

ix. Report Module

- Provide dynamic statistic report.

Target User:

i. System Administration

- To manage users in the system.

ii. Coordinator

- Manage and assign supervisor with students under the same programme.

iii. Supervisor

- Consist of lecturer representatives from the faculty that allow them to monitor and track student internship application status and industrial training progress.

iv. Student

- Search organization for internship, apply for internship, and generate resume through the system.

v. Organization

- Advertise their internship recruitment through the system.

1.5 Project Significance

Internship program is important that enables student to gain knowledge and exposure of working in the real world. This allows the student to apply what they have learned in the University to the work environment. It is very important for higher education like University or College to have a digital platform that can facilitate the internship application process. FTMK Internship Placement System is developed to facilitate the process of students eligible for industrial training to find a place to undergo such industrial training. This system can be used as a platform for Industrial training coordinator to keep track and monitor student internship application status. This will act as centralized data center for UTeM internship students. History and record will be stored in the database.

1.6 Expected Output

The expected outcome for the FTMK Internship Placement System:

- Students able to use resume template generated by the system.
- Student able to choose and apply for internship placement.
- Supervisor will be able to monitor and track their students industrial training progress.
- Provide offer letter and feedback form in the system.
- Notification through system and email regarding the industrial training status.

1.7 Conclusion

In conclusion, the FTMK Internship Placement System will help facilitate UTeM staff and students for industrial training. The system consists of 9 module which are User Authentication Module, Management Module, Announcement Module, Resume Module, Internship Application Module, Organization Profile Module, Internship Recruitment Module, Supervision Module, Report Module.

In chapter 2, literature review related to the FIPS and project methodology will be stated and applied into development of the system.

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

This chapter aim to discuss on the related literature and to study the student internship placement process to develop proper guideline in web-based application.

This chapter also discussed about the methodology of FTMK Internship Placement System. In developing a system, methodology is the main part to be referred because it consists of a set of methods, procedures, business rules, and practices used to conduct the development process. Planning is important, in order to develop a good system that can be contributed to the community.

2.2 Web Based

According to (Ishak, 2012), students were given access to precise information about academic and learning courses through the usage of the internet. In today's world, web-based systems are the most important platform or medium for all types of people. People believe that online applications make it easier for them to find information in their daily lives because of these factors. Web-based applications have become a convenient way to search for numerous pieces of information at once. According to (Hacker, 2008), enhance the availability of information and make it easier to obtain it, and a plethora of online services were enhancing productivity and operational efficiency around the world.

Web architecture consists of four elements: a web browser for the client, a web server for generating data display, an application server for evaluating business logic, and a database server for data providers. (Tao, n.d.). The figure 2.1 shows the architecture of the web application.

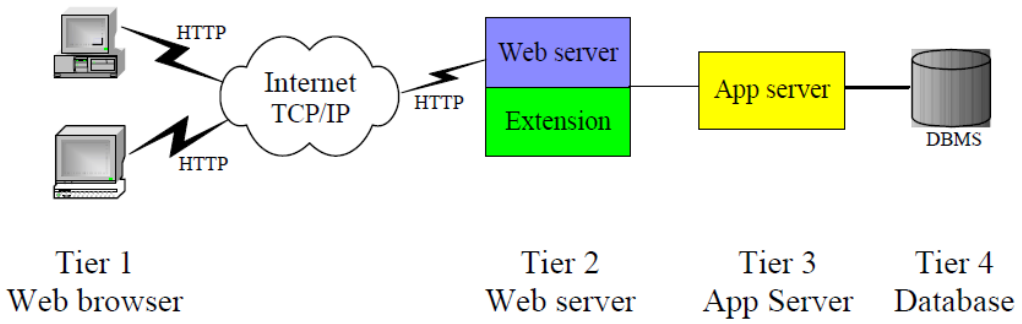


Figure 2.1: Web Architecture (Source: Introduction to Web Technologies (Tao, n.d.))

The figure 2.1 tells the flow of the architecture of the web application from tier 1 to tier 4. On the tier 1 it shows the element which is the HTTP then it goes through the Internet. Next, tier 2 which consist of Web Server pass the element to tier 3, the application server. The application server retrieves the data from database on tier 4.

Aside from that, web-based applications require a web server, which serves HTML files to web clients. Figure 2.2 depicts the web server acting as a receiver for receiving any document requests or data submissions from browsers through the TCP/IP layer using the HTTP protocol. If a client requests a static file that already exists, the file will be retrieved from a server hard disc and promptly transmitted back to the web client.

According to (Othman, 2012), before the server responds to the requested web page, the browser communicates with the web server and the browser goes to the server operating system to listen and answer to user queries.

Furthermore, the web browser is the client's graphical user interface for interacting with applications. The web browser has some basic functions that are required by most client web applications. The web browser's functions include constructing HTML markup visually and presenting documents, using HTTP protocol and HTML form to send requests to web applications, web applications also maintaining cookies on the client computer, and plug-in applications used for additional functions, such as an application that supports video files (Tao, n.d.). Figure 2.2 shows the web client and the web server.

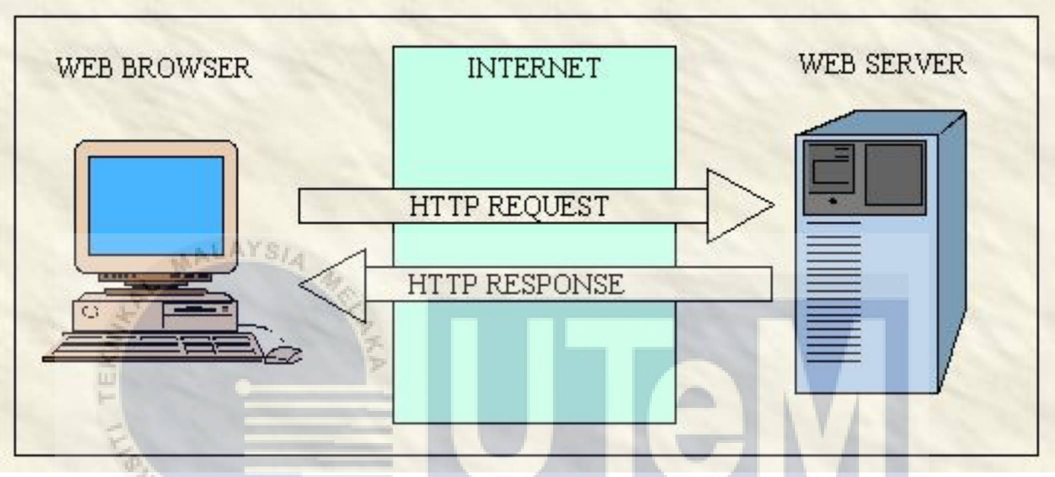


Figure 2.2: Web Client and Web Server (Source: http://www.tonymarston.net/uniface/html-from-xml-and-xsl.html (Marston, 2003))

2.2.1 Infrastructure

Three components make for a three-tier architecture: server, application, and client. Figure 2.3 depicts the execution of a three-tier architecture with the Client as the first tier, the Application as the second tier, and the Database Server as the third layer.

A network element is also included in this architecture to connect the three components. According to (Chen, 2003), the three-architecture provides better transparency in the data layer, business logic layer, and user interface layer, making systems more configurable, maintainable, resilient, and durable over the system's lifetime. Figure 2.3 shows the three-tier of architecture.

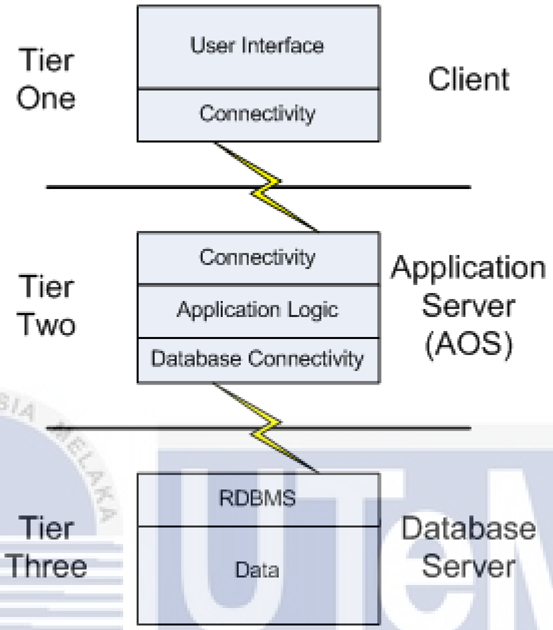


Figure 2.3: Three-tier Architecture (Source: https://msdn.microsoft.com/en-us/library/aa660629.aspx (AOS Overview [AX 2012]))

2.2.2 Database

Any organization's database is one of the most important components for storing customer data and connecting clients via a web-based system. The establishment of a database necessitated the running of a business. If the company requires a huge database storage, the developer may be required to provide a Database Management System (DBMS). A database is a type of storage that stores data received from a web application known as a database management system, or DBMS. A database management system (DBMS) can handle massive amounts of data supplied from a web application. As a result, DBMS aids in increasing the efficiency and reliability of data retrieval methods for numerous users.

Databases come in a variety of types, including MySQL, MSSQL, PostgreSQL and Oracle. This server oversees data storage and management. The servers' performance, on the other hand, is more targeted. The database must respond to requests quickly and efficiently manage and save data.

Most initiatives that require storing, retrieving, and accessing digital data have used a database management system. The database is typically used to store and manage data by dynamic websites, accounting information systems, payroll systems, and stock management systems. Because it is a free and open-source database management system, many developers are now using it to build and produce. (Bassil, 2012).

i. MySQL

MySQL is a Relation Database Management System (RDBMS) that runs as a server and has numerous databases with multiple users. Following the extraction and processing process, all customer data is stored in a MySQL database.

According to (Saikia, 2015), the developer of all open-source RDBMS systems frequently and most popularly uses MySQL. As a result, MySQL has features that can be classified as platform agnostic.

These are the characteristics that are frequently utilised by developers and can be used on a variety of platforms. In addition, it employs a multi-layered server architecture with distinct modules. MySQL is also incredibly quick and handles a wide range of data formats. Supports both fixed-length and variable-length records and uses an extremely efficient thread-based memory allocation technique.

2.2.3 Programming Language

i. PHP

PHP is a popular programming language for creating web and application frameworks. It is a dynamic object-oriented language that focuses on server-side application development. According to (Hills, Enabling PFP Software Engineering Research in Rascal, 2014), on the server-side, PHP was used by 81.8 percent of developers. According to (Hills, Static, Lightweight Includes Resolution for PHP 1, 2013), PHP is the most widely used language for server-side application development. It gives the developer a lot of flexibility because to its highly dynamic language. Object-oriented characteristics include interfaces, exceptions, and attributes, as well as other common dynamic features seen in other scripting languages. Aside from that, it might provide a unique mechanism for dealing with undefined field and method jobs. PHP framework such as Yii2 Framework provides a basic structure for streamlining the development of web apps. It can speed up the development process.

ii. JavaScript

JavaScript (JS) is a high-level programming language that widely adopted as a general-purpose programming language. It is usually found running in a web browser as interactive or dynamic content. There is various usage of JavaScript such as Asynchronous JavaScript and XML (AJAX) that is to load content without refreshing the website, changing HTML through the Document Object Model (DOM). Framework such as jQuery, a JavaScript library, may allow programming to be easier as more predefined procedures are defined within it.

2.3 Facts and Findings with Related Work

Based on findings, each university has their own internship management approaches either by manual or technology approaches. Some universities or colleges such as Melaka International College of Science and Technology (MiCoST) is still using manual approach to manage the internship program while many oversea universities already start using information system to manage their internship program. For example, the world leading university like University of Oxford already started using website services as earlier as year 2002 to manage the internship program of University of Oxford. The information system named Career Connect provides a very good platform for student to upload all their details like cover letter, resume for evaluation of employer from all over the world. This provides great opportunity for students to get internship placement offers from employer. This system helps the employer in finding potential candidates for vacancies in their organization.

2.3.1 Domain

The domain that can relate to the FTMK Internship Placement System is Information System (IS) and internship program of university. The internship process of a university involves a large amount of information sharing and transfer between organization and student. It concerns more on the management of information such as storage and retrieval of student and organization information.

2.3.2 Existing System

The existing system related to FIPS is Universiti Teknologi Malaysia (UTM) Industrial Training System (ITS), a web-based application. For UTM ITS, the student needs to obtain detailed information about the organization, and then email it to the industrial training committee for approval. Once the organization approved by the industrial training committee, the status will be changed to pending in UTM ITS. During the industrial training period, the students are required to note and update their daily activities using e-log system on every working day. Organization supervisor able to evaluate and assess the student using the online Performance Evaluation Form via UTM ITS. The assessment by faculty supervisor is also conducted using the online form in UTM ITS.

Another existing system related to FIPS is Universiti Teknologi MARA (UiTM) Student Internship Training Information System (STIS), a web-based application. The UiTM STIS have 2 logins, first is for Industrial Training Company Supervisor, second is for staff or student of UiTM. For the internship process, student need to fill up the Industrial Application Form via the UiTM STIS. The process such as email, Confirmation Letter, Training Information Form, Acceptance Letter, Logbook is done outside of the UiTM STIS. The student can print and upload the Report Duty Form in the UiTM STIS. Although, other faculty need to submit it manually by hand.

2.3.3 Comparison between Existing System with this Project

Table 2.1 shows the comparison of existing system between the FTMK Internship Placement System, UTM Industrial Training System, and UiTM Student Internship Training Information System.

Table 2.1: Comparison of Existing System

	FTMK Internship Placement System	UTM Industrial Training System	UiTM Student Internship Training Information System
Software	Web-based application	Web-based application	Web-based application
Obtain Organization Details	Organization details available in the system	Obtain detailed information of organization manually	Obtain detailed information of organization manually
Form to be Filled	Online form	Online form, and paper form	Online form, and paper form
Email Process	Automatically send the email through the system	Need to send the email manually	Need to send the email manually
Logbook	-	E-log System	-
Report	Upload through system	-	Some faculty need to submit it manually by hand

2.3.4 Technique

Requirement gathering technique is critical to the success or failure of a system development. Good requirements start with good sources of requirements. Finding high quality sources is an important task. Below will discussed several techniques that are used to gather the requirements.

Brainstorming by stating the objective of the project, then generate as many ideas as possible. After the information have been gathered, reshape, and combine the ideas.

Conducting interview with users are important when gathering requirements. It is the primary source of requirements and an important way to gather and validate their requirements. Face-to-face interview has been done with the alumni UTeM student, that have finished the industrial training, gather requirement such as the flow of internship, based on the student perspective. Interview with UTeM industrial training coordinator also done through online meeting on Microsoft Teams. This is to gather the industrial training process from the coordinator and supervisor faculty perspective.

By reading the article or journal regarding the similar or existing system that is related to FIPS also can help to gather the requirements. This can save time and avoid reinventing the system that are already on the market.

2.4 Project Methodology

The System Development Life Cycle (SDLC) is a process for planning, creating, testing, and deploying an information system. The FIPS will be developed using the waterfall model of SDLC. Waterfall methodology is a project management process that mainly used for software development. Waterfall model illustrates the software development process in a linear sequential flow. Any phase in the development process begins, the previous phase must be complete first. Figure 2.4 shows the waterfall model for the SDLC.

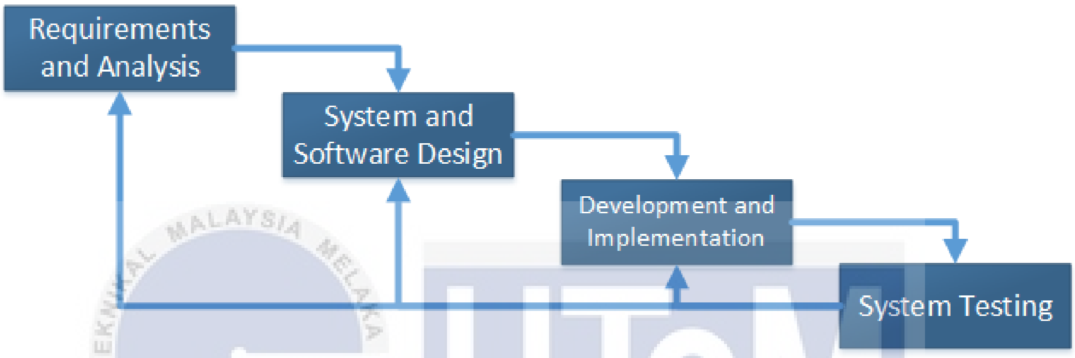

Requirements and Analysis
System and Software Design
Development and Implementation
System Testing

Figure 2.4: Waterfall Model for the SDLC

Requirements and analysis phase act as the first phase which gathered and analyzed needed information. They are statements that indicate what a system needs to do to provide a capability. Requirement gathering technique such as brainstorming, interview with the users, reading articles and journals, is done in this phase.

System and software design process acts as the second phase in the system development process. The developer team analyst the problem, choose the right System Development Life Cycle (SDLC) method to use. During this phase, the use case diagram, flowchart, wireframe, Entity Relationship Diagram (ERD) is developed for the FIPS.

Development and implementation act as the third phase in the system development process. The prototype for the system was created and shown to the stakeholder either the current develop system fulfil their needs or not.

The fourth and last element of the system development process is testing. This stage includes all aspects of the SDLC. Following the completion of the system's development phase, testing and assessment are required to confirm that the system's features perform properly. This procedure is crucial since it necessitates the verification that the built system fits all of the user's expectations. This phase assisted in identifying any errors that may have occurred in the system.

2.5 Project Requirements

This section shows about the hardware and software requirements used to develop the FIPS. Requirements such as software requirement, hardware requirement must be met before starting to develop the project and running the system. This is to ensure the optimal functionality and performance of the system.

2.5.1 Software Requirement

In developing the system, the developer used all reliable tools specified in the table below. The following table 2.2 shows the requirement details of software that would be needed in the developing process of FIPS.

Table 2.2: Software Requirement

Software	Description
Operating System	Microsoft Windows 10 Home
Software Development Tools	Laragon • Apache 2.4.46 • PHP 7.4.16 • MySQL 8.0.23 Microsoft Visual Studio Code Notepad++ DBeaver phpMyAdmin
Documentation Software	Microsoft Office 365 Education • Microsoft Word Microsoft Visio Professional 2019 Microsoft Project Professional 2019
Presentation Software	Microsoft Office 365 Education • Microsoft PowerPoint

2.5.2 Hardware Requirement

In process of FIPS development, a stable hardware is needed. The following table 2.3 shows the requirement details of hardware used in developing process of FIPS.

Table 2.3: Hardware Requirement

Hardware	Description
Laptop Specification	Lenovo Legion 5 15ARH05H • AMD Ryzen 7 4800H with Radeon Graphics up to 4.20 GHz, 8 Cores • ADATA 16 GB DDR4 RAM, 2400 MHz • NVIDIA GeForce GTX 1660 Ti, 6 GB VRAM • ADATA XPG SX8200 M.2 PCIe SSD 2 TB
Backup Storage	Google Drive GitLab
Printer	HP Laser Jet P1006
CD/DVD Burner	HP DVDRW GUE1N

2.5.3 Other Requirements

Other requirements consist of requirement other than software and hardware requirement. For meeting with the users to gather requirements, the required software is described in the table below. Table 2.4 shows the requirement details that is not specified in the hardware and software requirements.

Table 2.4: Other Requirement

Software	Description
Meeting	Microsoft 365 Education • Microsoft Teams

2.6 Project Schedule and Milestones

The project schedule and milestones stated using Gantt Chart that provide standard format for displaying project schedule information by listing project activities and their corresponding start and finish dates in a calendar format. Milestones are significant events on a project that normally have zero duration. Appendix A shows the table of Gantt Chart for developing the FIPS.

2.7 Conclusion

In conclusion, this chapter have reviewed some of the previous system that have been completed by other researchers. In developing the FIPS, waterfall model will be used for the System Development Life Cycle.

In the next chapter, which is chapter 3, it will have the further describe about the approach on the proposing a solution to the FIPS.

CHAPTER 3: ANALYSIS

3.1 Introduction

This chapter discussed about problem analysis and requirement analysis. It is to collect, analyze, validate, and define high-level needs and features of the FIPS. It is also to identify, clarifying, and organizing system requirements.

3.2 Problem Analysis

The problem analysis is to analyze the current problem of current system scenario. The UTM ITS will be used for the current system to be analyzed.

3.2.1 Business Process

The business process of UTM ITS consist of industrial training committee, student, and organization. Figure 2.5 shows the UTM ITS business process.

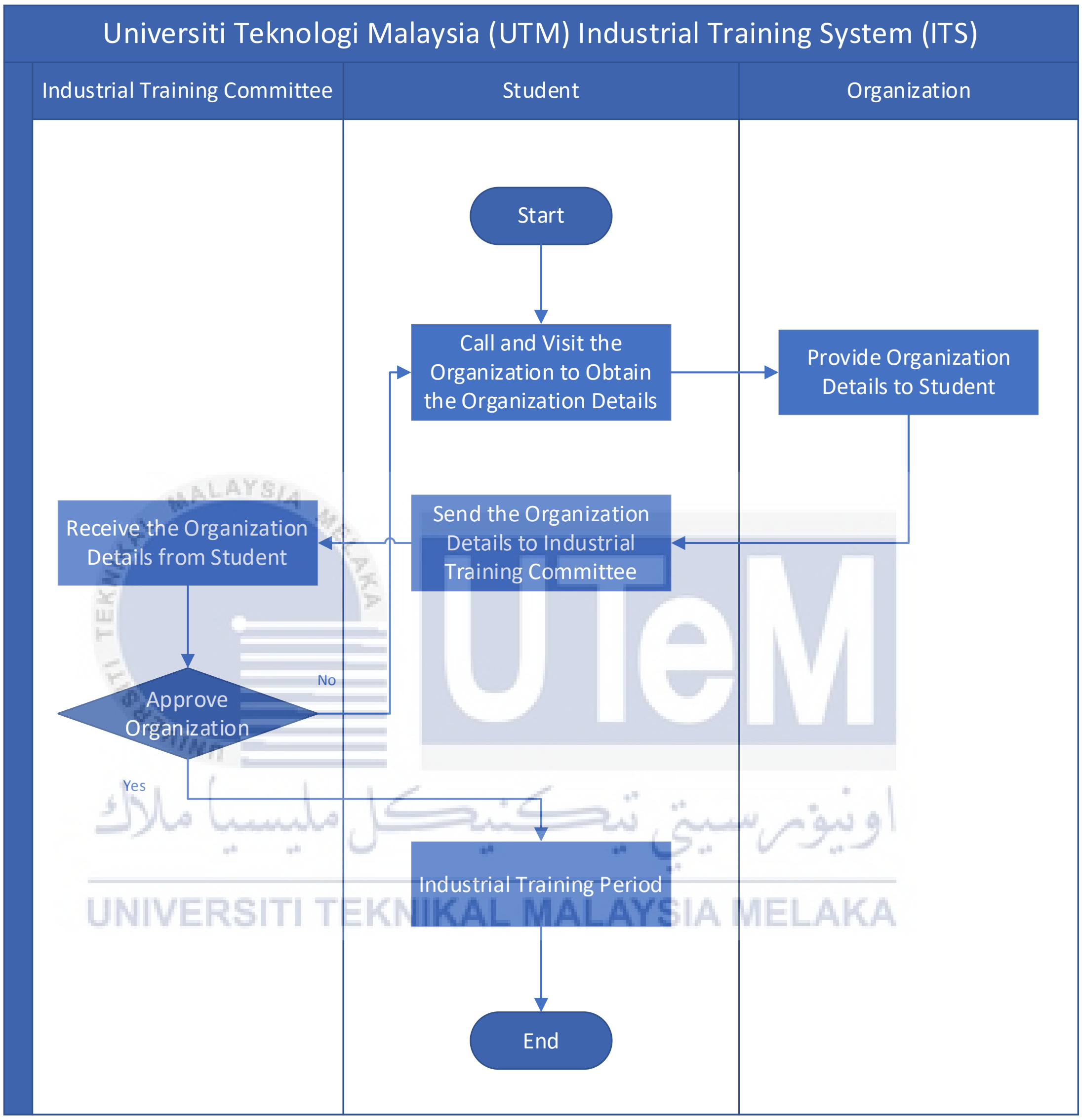

Universiti Teknologi Malaysia (UTM) Industrial Training System (ITS)
Industrial Training Committee
Student
Organization
Start
Call and Visit the Organization to Obtain the Organization Details
Provide Organization Details to Student
Send the Organization Details to Industrial Training Committee
Receive the Organization Details from Student
Approve Organization
No
Yes
Industrial Training Period
End

Figure 3.1: UTM ITS Business Process

3.2.2 Problem Decomposition Description

In the UTM ITS, the student needs to call and visit the potential organization to obtain detailed information about the organization, and then email it to the industrial training committee for approval. Once the organization approved by the industrial training committee, the status will be changed to pending in UTM ITS. During the industrial training period, the students are required to note and update their daily activities using e-log system on every working day.

The problem is divided into several parts as follows:

- Obtain organization information
- Email to the industrial training committee
- Status progress
- Logbook

3.3 Requirement Analysis

The requirement analysis phase converts the high-level explanation of the business requirements specified in the system request into a more explicit list of requirements.

Functional and non-functional requirements are the two types of requirements. A functional requirement is closely related to a procedure that the system must do and the data that it must include. Non-functional requirements refer to the system's behavioral characteristics. All of the functional and non-functional requirements that fall inside the scope of the system, which are utilized to build other analysis deliverables and lead to the new system's initial design.

3.3.1 Data Requirement

The data requirement is the data that system should be input and output. It is also described about the data that should store internally. Database name that will use to store the information of the system is ftmk_internship_placement_system. Table 3.1, Table 3.2, and Table 3.3 shows the data requirement for FIPS.

Table 3.1: Data Requirement for Personal Detail

Data	Description
id	Personal Detail ID
id_user	User ID
id_programme	Programme ID
id_state	State ID
id_attachment	Attachment ID
nric_no	NRIC No
user_no	Staff No / Matric No
name	Name
gender_no	Gender
religion	Religion
tel_no	Telephone Number
address	Address

Table 3.2: Data Requirement for Organization Detail

Data	Description
id	Personal Detail ID
id_user	User ID
id_state	State ID
id_attachment	Attachment ID
name	Name
type	Government or Private Sector
address	Address
tel_no	Contact Number
start_day	Office Start Day
end_day	Office End Day
open_hour	Office Open Hour
close_hour	Office Close Hour
job_description	Job Description

Table 3.3: Data Requirement for Internship Application

Data	Description
id	Personal Detail ID
id_personal_detail	Personal Detail ID
id_organization_detail	Organization Detail ID
application_status	State ID
application_status	The Status Process of Application
notes	Notes

3.3.2 Functional Requirement

The functional requirement is discussed about the scope of FIPS involve Administrator, Coordinator, Supervisor, Student, and Organization. Figure 3.2 shows the use case diagram for FIPS. Table 3.4, Table 3.5, Table 3.6, Table 3.7, Table 3.8, Table 3.9, Table 3.10, Table 3.11, and Table 3.12 shows the functional requirements for each module of FIPS.

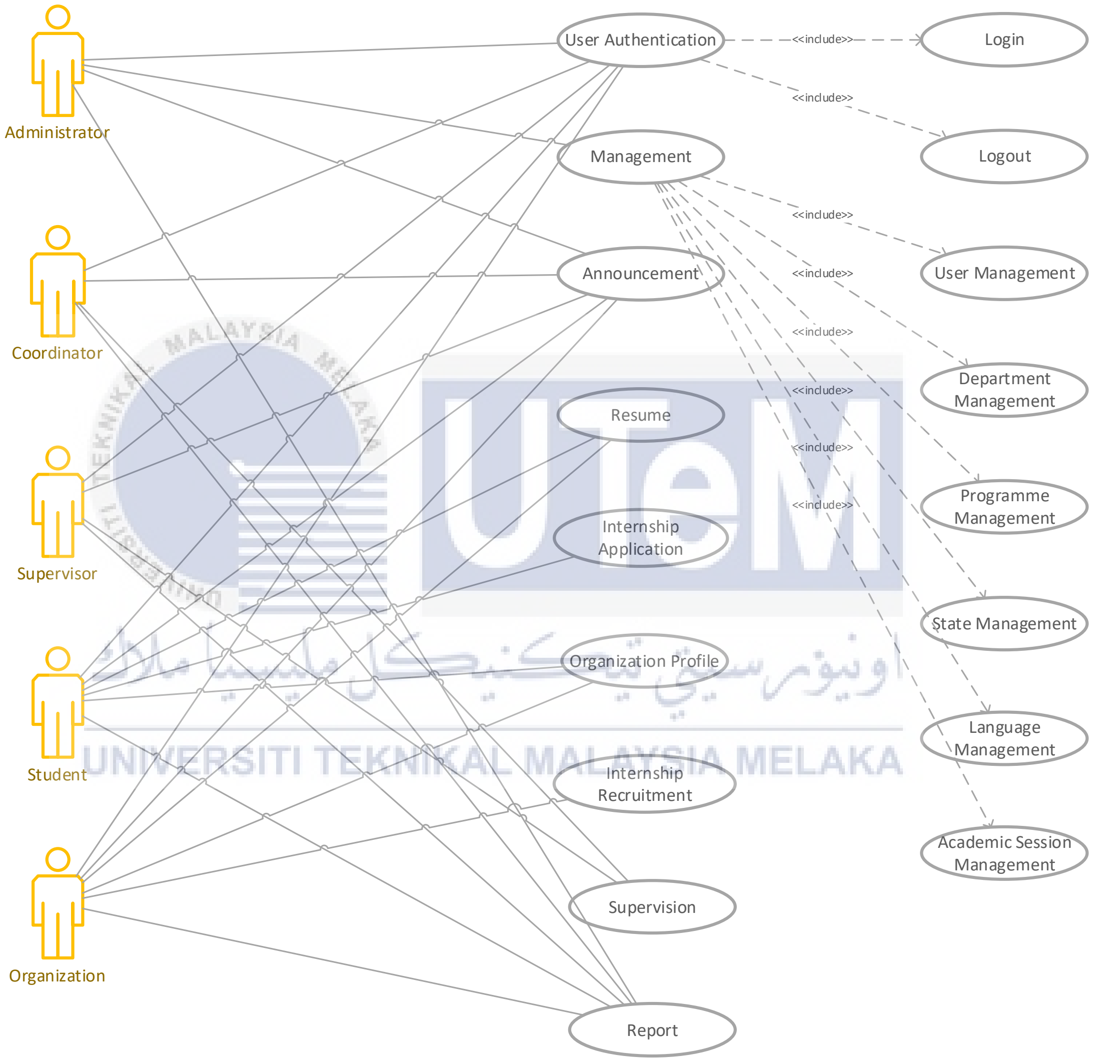


Figure 3.2: Use Case Diagram

Table 3.4: Functional Requirement for User Authentication Module

Requirement Number	Module Name	Description
FIPS-FR-101	User Authentication	When the username and password is valid, the user will be able to login to the main dashboard
FIPS-FR-102		The user can only access module based on the specified role (administrator, coordinator, supervisor, student, organization)
FIPS-FR-103		User able to logout from the system

Table 3.5: Functional Requirement for Management Module

Requirement Number	Module Name	Description
FIPS-FR-201	Management	Register user into the system
FIPS-FR-202		Update user details such as username, email, and role
FIPS-FR-203		Delete user from the system
FIPS-FR-204		Register department into the system
FIPS-FR-205		Update department details such as department code, and department name
FIPS-FR-206		Delete department from the system
FIPS-FR-207		Register programme into the system
FIPS-FR-208		Programme details such as department code, programme code, and programme name
FIPS-FR-209		Delete programme from the system
FIPS-FR-210		Register state into the system
FIPS-FR-211		Update state details such as state name and state type
FIPS-FR-212		Remove state from the system
FIPS-FR-213		Register language into the system

FIPS-FR-214		Update language details such as language name
FIPS-FR-215		Delete language from the system
FIPS-FR-216		Register academic session into the system
FIPS-FR-217		Update academic session details such as semester, start year, end year, and status
FIPS-FR-218		Remove academic session from the system

Table 3.6: Functional Requirement for Announcement Module

Requirement Number	Module Name	Description
FIPS-FR-301	Announcement	Post announcement
FIPS-FR-302		Update announcement details such as title, content, and role name
FIPS-FR-303		Show announcement based on role
FIPS-FR-304		Delete announcement from the system

Table 3.7: Functional Requirement for Resume Module

Requirement Number	Module Name	Description
FIPS-FR-401	Resume	Generate resume from the system
FIPS-FR-402		Update resume details such as profile picture, telephone number, address, skills, languages, education, work experience, and achievement
FIPS-FR-403		Add multiple skills, languages, education, work experience, and achievement

Table 3.8: Functional Requirement for Internship Application Module

Requirement Number	Module Name	Description
FIPS-FR-501	Internship Application	Show the list of organization
FIPS-FR-502		View organization profile
FIPS-FR-503		Apply internship at the selected organization

Table 3.9: Functional Requirement for Organization Profile Module

Requirement Number	Module Name	Description
FIPS-FR-601	Organization Profile	Generate organization profile from the system
FIPS-FR-602		Update organization profile details such as logo, name, type, address, state, contact number, working days (start day and end day), working hours (open hour and close hour), job description.

Table 3.10: Functional Requirement for Internship Recruitment Module

Requirement Number	Module Name	Description
FIPS-FR-701	Internship Recruitment	Show the student applicant list
FIPS-FR-702		View student resume
FIPS-FR-703		Accept student internship application
FIPS-FR-704		Reject student internship application

Table 3.11: Functional Requirement for Supervision Module

Requirement Number	Module Name	Description
FIPS-FR-801	Supervision	Create supervisor based on the selected academic session
FIPS-FR-802		Update supervisor details such as supervisor name and academic session
FIPS-FR-803		Delete supervisor
FIPS-FR-804		Assign student with supervisor

Table 3.12: Functional Requirement for Report Module

Requirement Number	Module Name	Description
FIPS-FR-801	Report	Generate report from the system

3.3.3 Non-functional Requirement

Non-functional requirements are those that define criteria that can be used to evaluate a system's functioning rather than specific behaviors. It should also be distinguished from functional requirements, which define specific behavior or functions. Table 3.13 shows the non-functional requirements for the FIPS.

Table 3.13: Non-functional Requirement

Requirement Number	Description
FIPS-NFR-101	Easy to use, by providing compact and useful information of text, appropriate button color
FIPS-NFR-102	The dashboard will load within 3 seconds after the user logins
FIPS-NFR-103	The system can accept 100 concurrent users without system hangs and system response time of 10 seconds

3.4 Conclusion

In this analysis phase, a better understanding of business process that was happening in the current system was gained. Consequently, it also helped in discovering the problems to be solved.

To solve the problem, the FIPS will be developed. The analysis of the system decomposed the system into main modules to be developed and each module was described in detail in Functional Requirements.

In the next chapter, which is chapter 4, it will further describe about system design.

CHAPTER 4: DESIGN

4.1 Introduction

The outcomes of the preliminary design analysis and the detailed design result are defined in this chapter. System architecture, user interface design, navigation design, input design, and output design are all examples of how to design a system based on studying the flow system. The components, their interfaces, and behaviours are defined by the system architecture.

4.2 High-Level Design

The project's architecture will investigate the proposed application's functional and non-functional requirements and create an overall solution architecture for the application that can manage those needs in the high-level design. The high-level design in this project will include system architecture, user interface, and database design, both conceptual and logical. Figure 4.1 shows the high-level system context view for the FIPS.

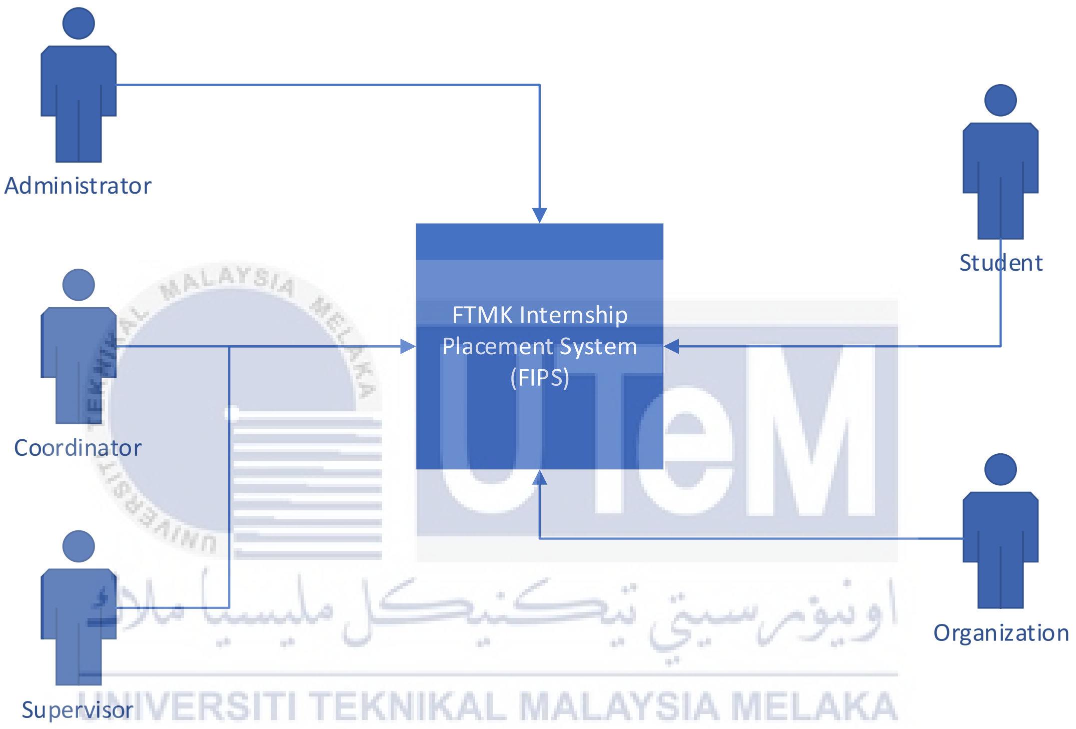


Figure 4.1: High-Level System Context View

4.2.1 System Architecture

The system architecture of this system is presented by using three-tier architecture. Three-tier architecture is a client-server architecture that consists of three layers which are, the presentation (frontend), the application logic, and the database.

The frontend is where the user sees and interacts within inside the browser. The main purpose of the frontend is to collect data from users. Next, the application logic, which is not accessible by the users, it processes and manipulates the data. Lastly, the database is where the data is stored and retrieved. Figure 4.2 shows the web application structure for FIPS.

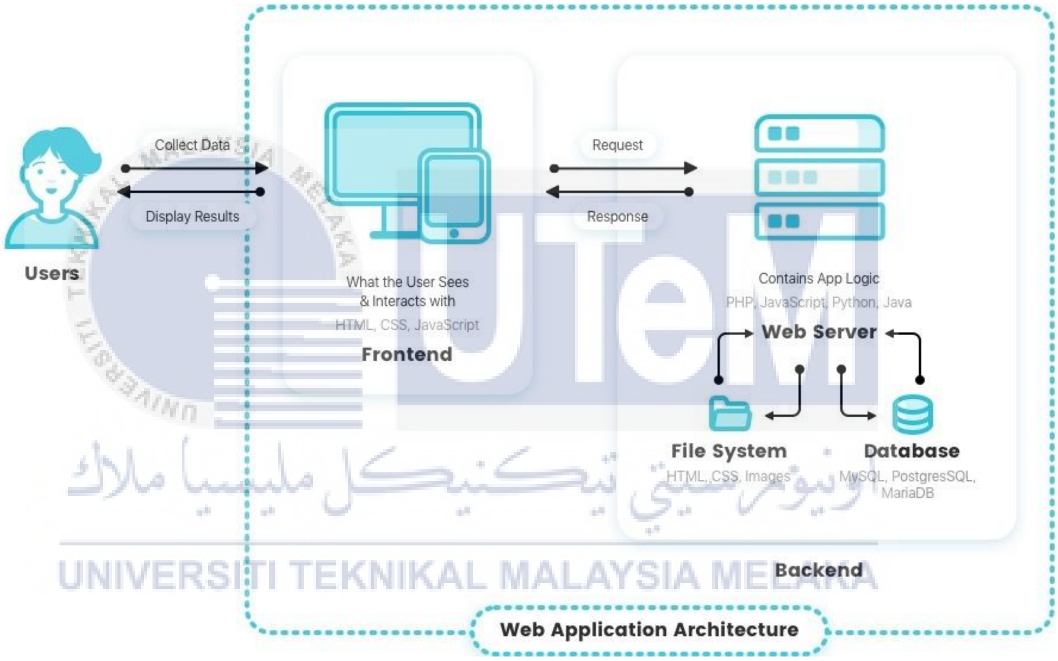


Figure 4.2: Web Application Structure (Mark Dabbs 2019)

4.2.2 User Interface Design

The user interface is the medium through which the user interacts with the system. It's crucial since the success of a system is largely determined by how many people can grasp how to use it. To make the interaction between the user and the system effective, the user interface must be appropriately designed. The navigation mechanism, input mechanism, and output mechanism are the three basic components of a user interface.

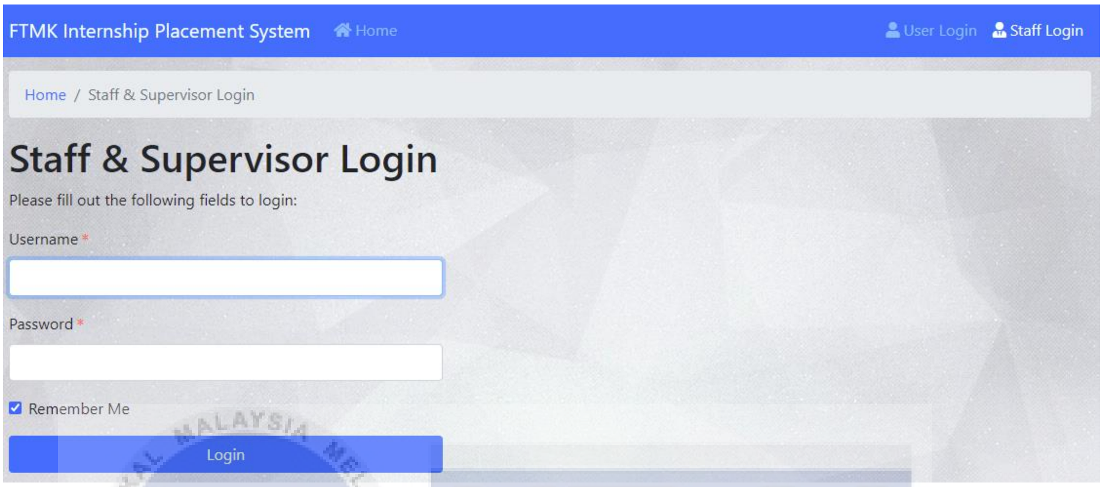
4.2.2.1 Navigation Design

The way a user navigates through a system using buttons or menus is referred to as navigation design. The navigation design is concerned with the flow of the system's user interfaces. Appendix B shows the navigation flow for FIPS.

4.2.2.2 Input Design

The input design of this system depicts the screens or user interfaces that users will use to enter data, as well as the validation criteria that will be applied to the data.

The FIPS consist of two login interface which are staff login, and user login. The staff login is for system administrator, coordinator, and supervisor. Meanwhile for the user login is for the student and organization. The user must key in username and password to be able to login into the system. Figure 4.3 shows the input design for staff login. Figure 4.4 shows the input design user login.


FTMK Internship Placement System

Home / Staff & Supervisor Login

Staff & Supervisor Login

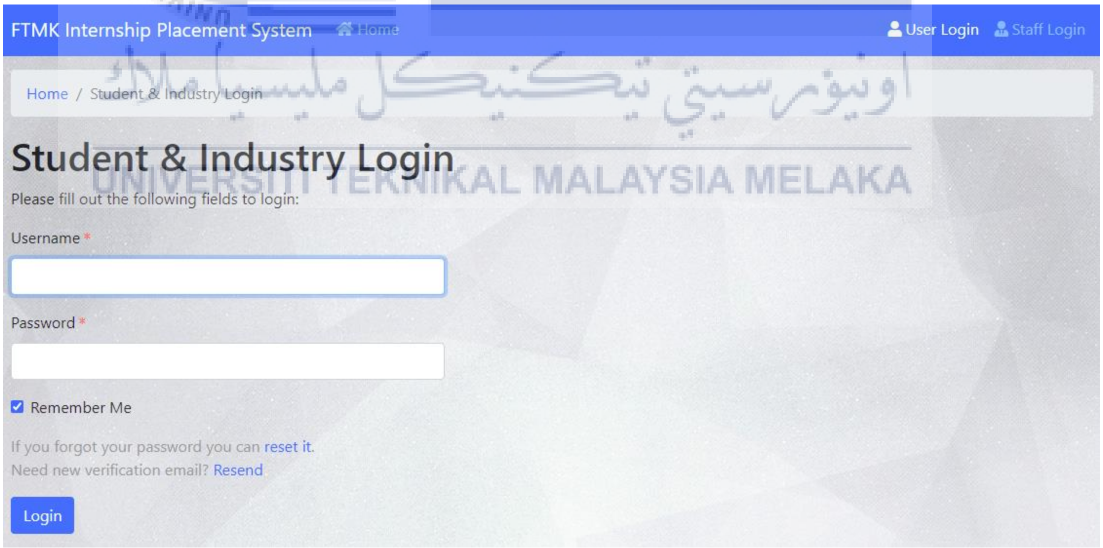
Please fill out the following fields to login:

Username *

Password *

☑ Remember Me

Figure 4.3: Input Design for Staff Login


FTMK Internship Placement System

Home / Student & Industry Login

Student & Industry Login

Please fill out the following fields to login:

Username *

Password *

☑ Remember Me

If you forgot your password you can reset it.
Need new verification email? Resend

Figure 4.4: Input Design for User Login

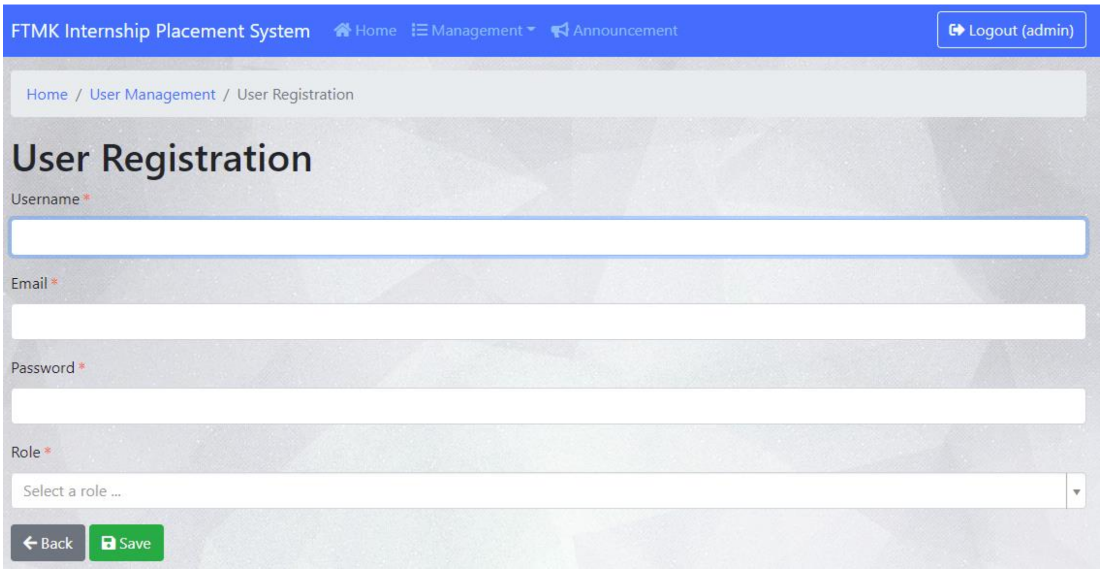
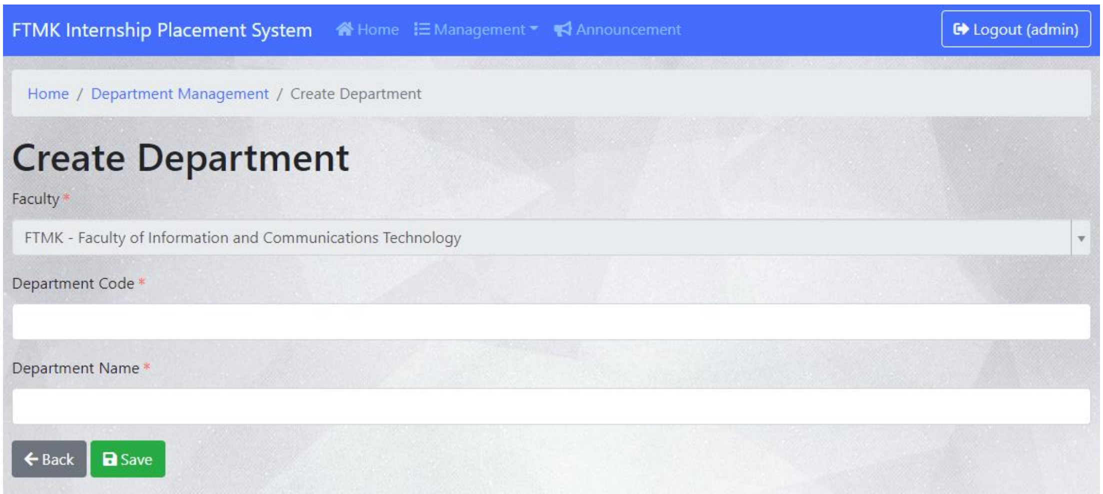


Figure 4.5: Input Design for User Registration (Admin)

Based on figure 4.5, it shows the input design for user registration. The admin must key in username, email, password, and role to register the user. The email input will only accept email format. The username and email will validated for uniqueness. The role dropdown show option such as administrator, coordinator, supervisor, student, and organization. Additional text input or dropdown will be shown based on the chosen role. When the chosen role is coordinator or supervisor, additional input will appear such as Staff No., Name, Gender, Tel. No., Programme. If the chosen role is student, then the additional input will be NRIC No., Matric No., Name, Gender, Religion, Tel. No., Programme, Address. Role such as administrator or organization will not show any additional input.


FTMK Internship Placement System
Home
Management
Announcement
Logout (admin)
Home / Department Management / Create Department
Back
Save

Create Department

Faculty *

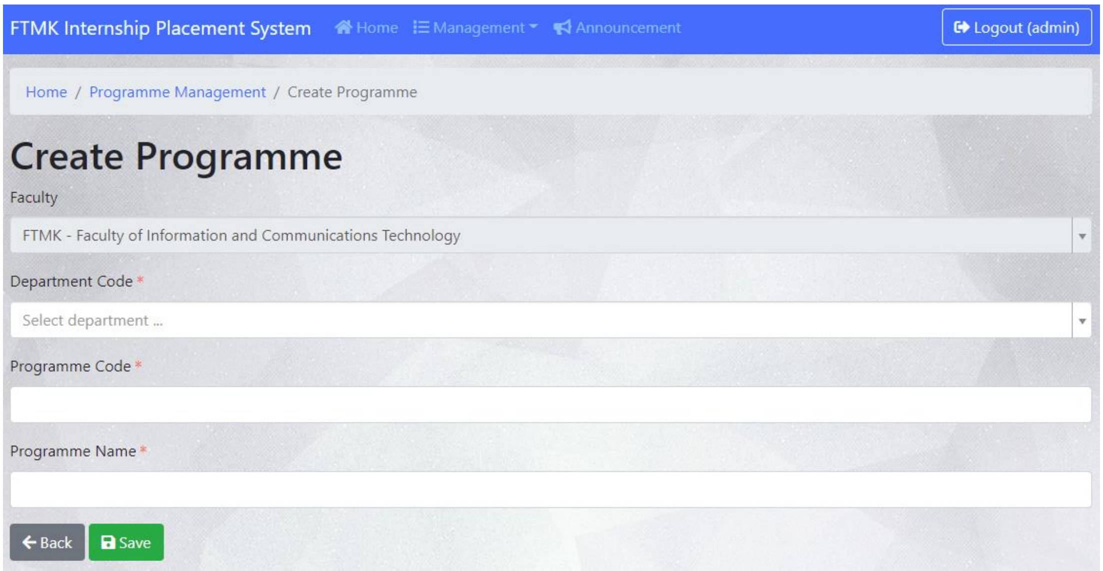
FTMK - Faculty of Information and Communications Technology

Department Code *

Department Name *

Figure 4.6: Input Design for Create Department (Admin)

Based on figure 4.6, it shows the input design for create department. The faculty dropdown will be fixed at FTMK only. While for the department code and department name must be fill in. The department code will be validated and only accept unique department code.



FTMK Internship Placement System | Home | Management | Announcement | Logout (admin)

Home / Programme Management / Create Programme

Create Programme

Faculty: FTMK - Faculty of Information and Communications Technology

Department Code *: Select department ...

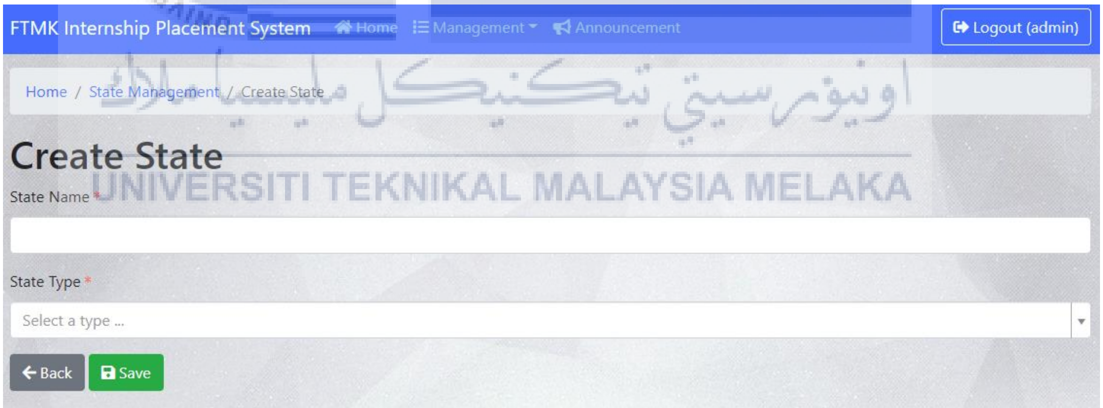
Programme Code *:

Programme Name *:

Back | Save

Figure 4.7: Input Design for Create Programme (Admin)

Based on figure 4.7, it shows the input design for create programme. The faculty dropdown will be fixed at FTMK only. The department code can be chosen from the dropdown, the department code is created from the Department Management module. The programme code and programme name must be fill in. The programme code will only accept unique programe code.



FTMK Internship Placement System | Home | Management | Announcement | Logout (admin)

Home / State Management / Create State

Create State

State Name *:

State Type *: Select a type ...

Back | Save

Figure 4.8: Input Design for Create State (Admin)

Based on figure 4.8, it shows the input design for create state. When creating state, the admin needs to input the state name and state type. The state type dropdown will show the data such as “Saturday and Sunday Holidays” and “Friday and Saturday Holidays”.

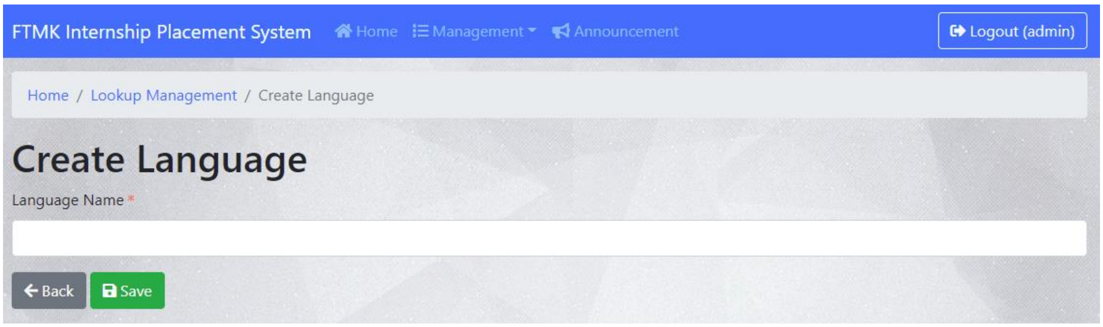


Figure 4.9: Input Design for Create Language (Admin)

Based on figure 4.9, it shows the input design for create language. To create a language for lookup, only language name needs to be input, and the name must be unique.

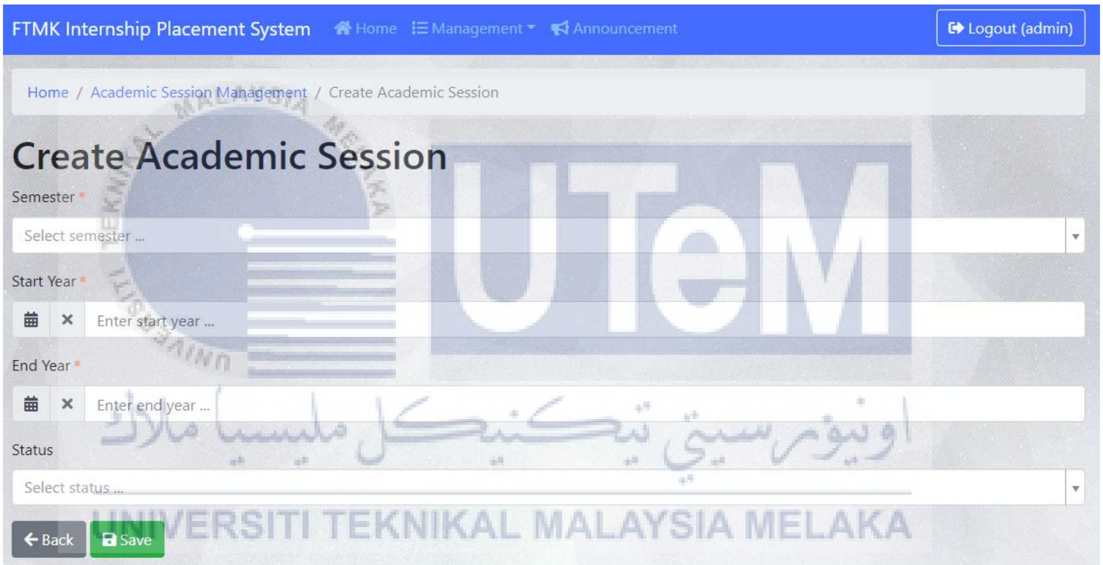


Figure 4.10: Input Design for Create Academic Session (Admin)

Based on figure 4.10, it shows the input design for create academic session. The semester dropdown will show data such as “1”, “2”, and “3”. For the start year and end year input, it will show a datepicker widget and show year only to choose from. The status dropdown will show data such as “Active”, and “Inactive”. Only one session can be active at a time.

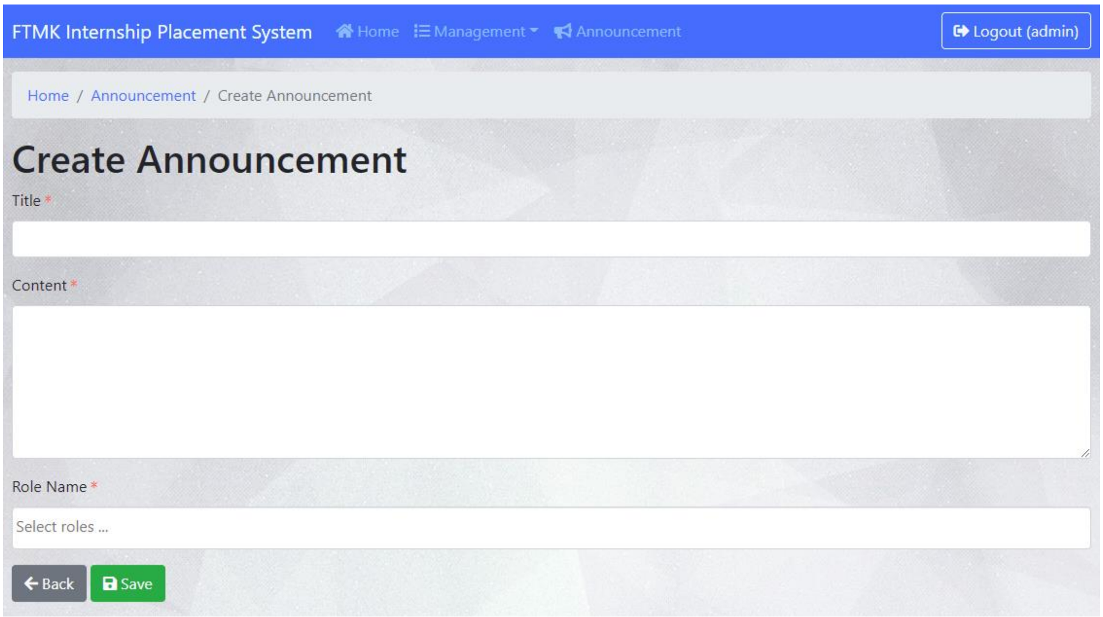
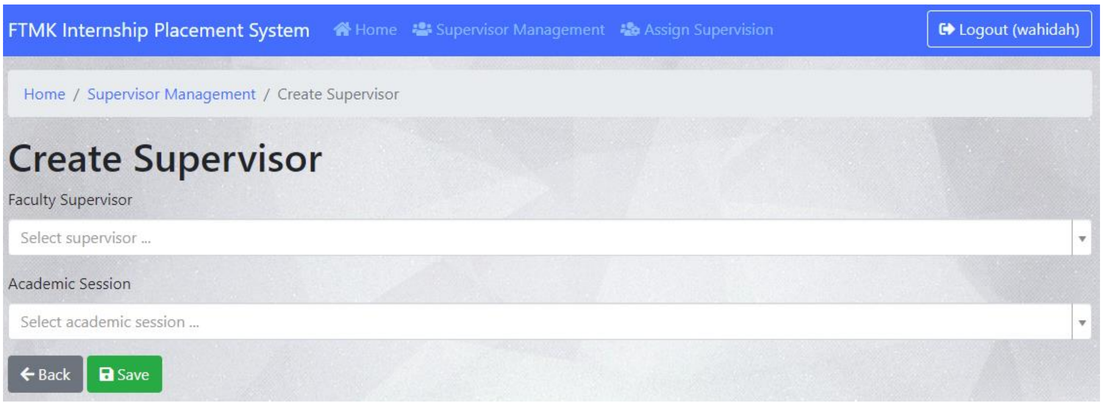


Figure 4.11: Input Design for Create Announcement (Admin)

Based on figure 4.11, it shows the input design for create announcement. The admin can post announcement by inputting data such as title, content, and roles. The role name dropdown will show all the role available in the system. Multiple roles can be selected from the dropdown.


FTMK Internship Placement System
Home
Supervisor Management
Assign Supervision
Logout (wahidah)
Home / Supervisor Management / Create Supervisor
Back
Save

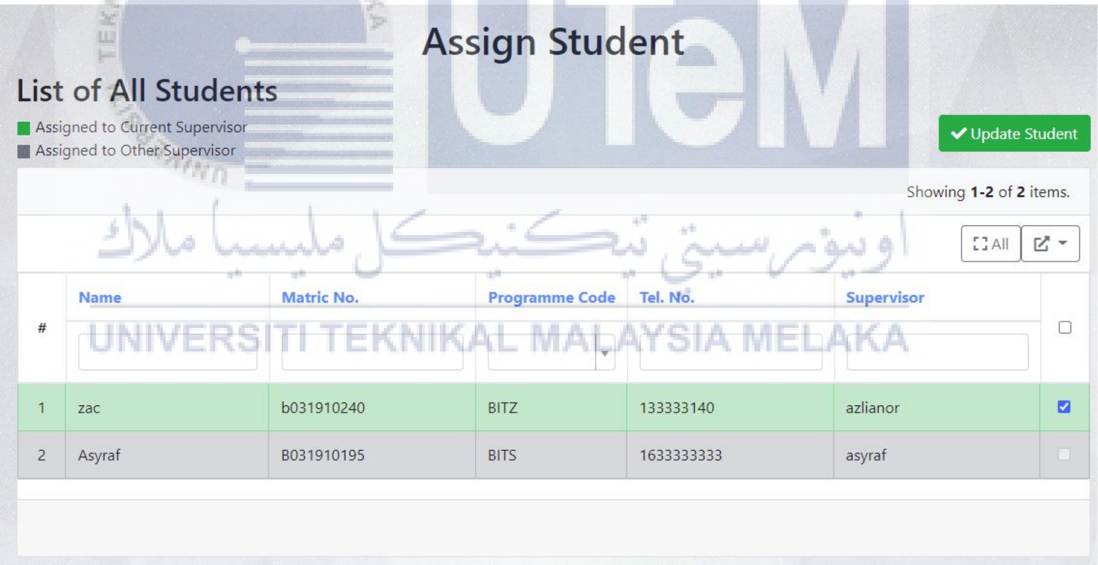
Create Supervisor

Faculty Supervisor: Select supervisor ...

Academic Session: Select academic session ...

Figure 4.11: Input Design for Create Supervisor (Coordinator)

Based on figure 4.11, it shows the input design for create supervisor. The coordinator can create supervisor based on the academic session. The faculty supervisor dropdown will list all the user with role as supervisor, and then assign them with the selected academic session. The academic session dropdown will list the data created from the Academic Session Management module.



Assign Student

List of All Students

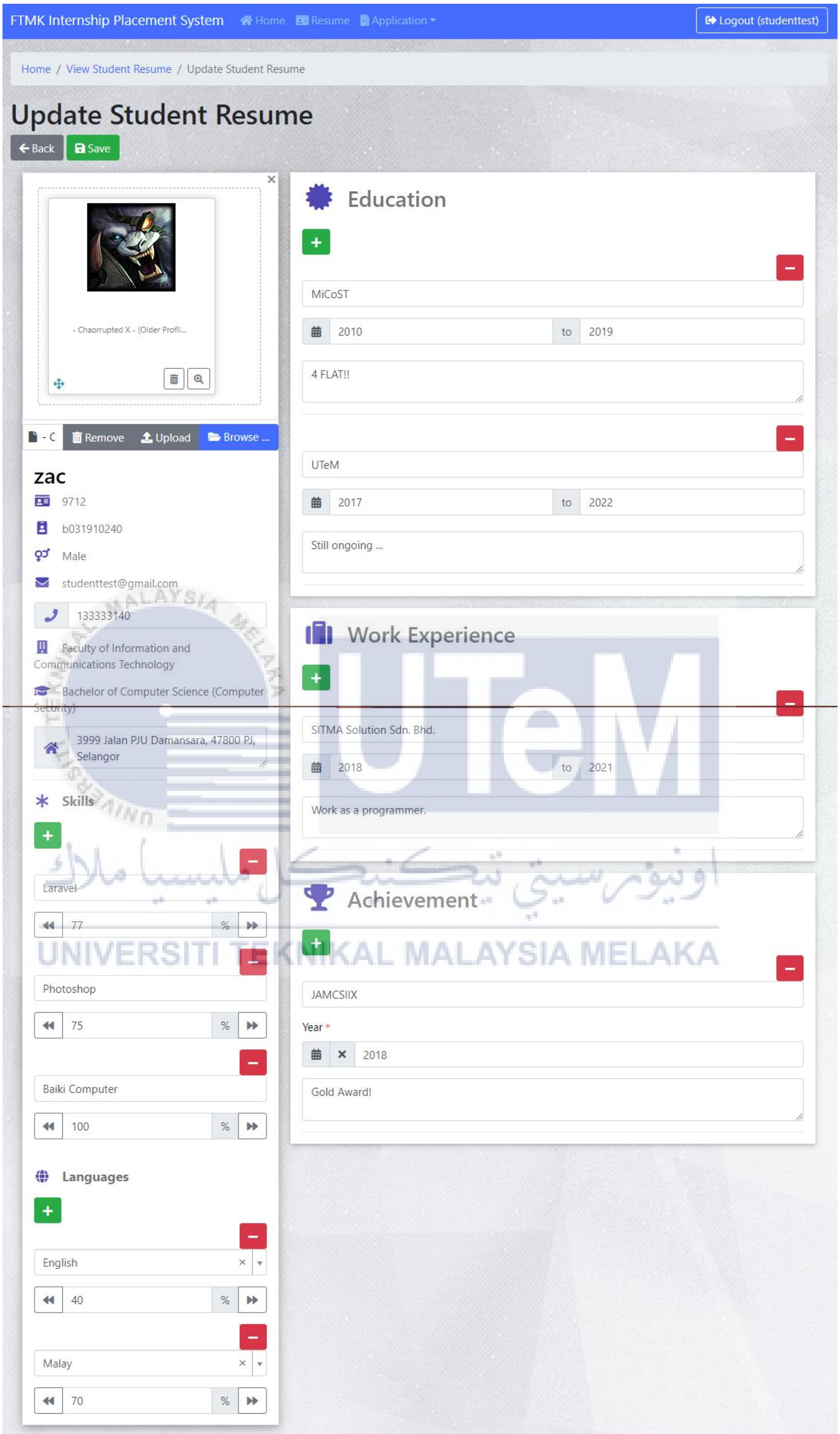
🟩 Assigned to Current Supervisor
▨ Assigned to Other Supervisor

Showing 1-2 of 2 items.

#	Name	Matric No.	Programme Code	Tel. No.	Supervisor	□
1	zac	b031910240	BITZ	133333140	azlianor	☑
2	Asyraf	B031910195	BITS	1633333333	asyraf	□

Figure 4.12: Input Design for Assign Supervision (Coordinator)

Based on figure 4.12, it shows the input design for assign supervision. The coordinator can assign students under a supervisor.


FTMK Internship Placement System
Home
Resume
Application
Logout (studenttest)
Home / View Student Resume / Update Student Resume
Back
Save
Remove
Upload
Browse ...

Update Student Resume

- Chaorrupted X - (Older Profil...

zac

9712

b031910240

Male

studenttest@gmail.com

133333140

Faculty of Information and Communications Technology

Bachelor of Computer Science (Computer Security)

3999 Jalan PJU Damansara, 47800 PJ, Selangor

Skills

Laravel 77 %

Photoshop 75 %

Baiki Computer 100 %

Languages

English 40 %

Malay 70 %

Education

MiCoST

2010 to 2019

4 FLAT!!

UTeM

2017 to 2022

Still ongoing ...

Work Experience

SITMA Solution Sdn. Bhd.

2018 to 2021

Work as a programmer.

Achievement

JAMCSIIX

Year *

2018

Gold Award!

Figure 4.13: Input Design for Student Resume (Student)

Based on figure 4.13, it shows the input design for student resume. Student can input data such as Tel. No., Address, Skills, Languages, Educations, Work Experiences, and Achievements. For Skills, Languages, Educations, Work Experiences, and Achievements section, student can input multiple data. The student can also upload their profile picture. Only image is accepted.

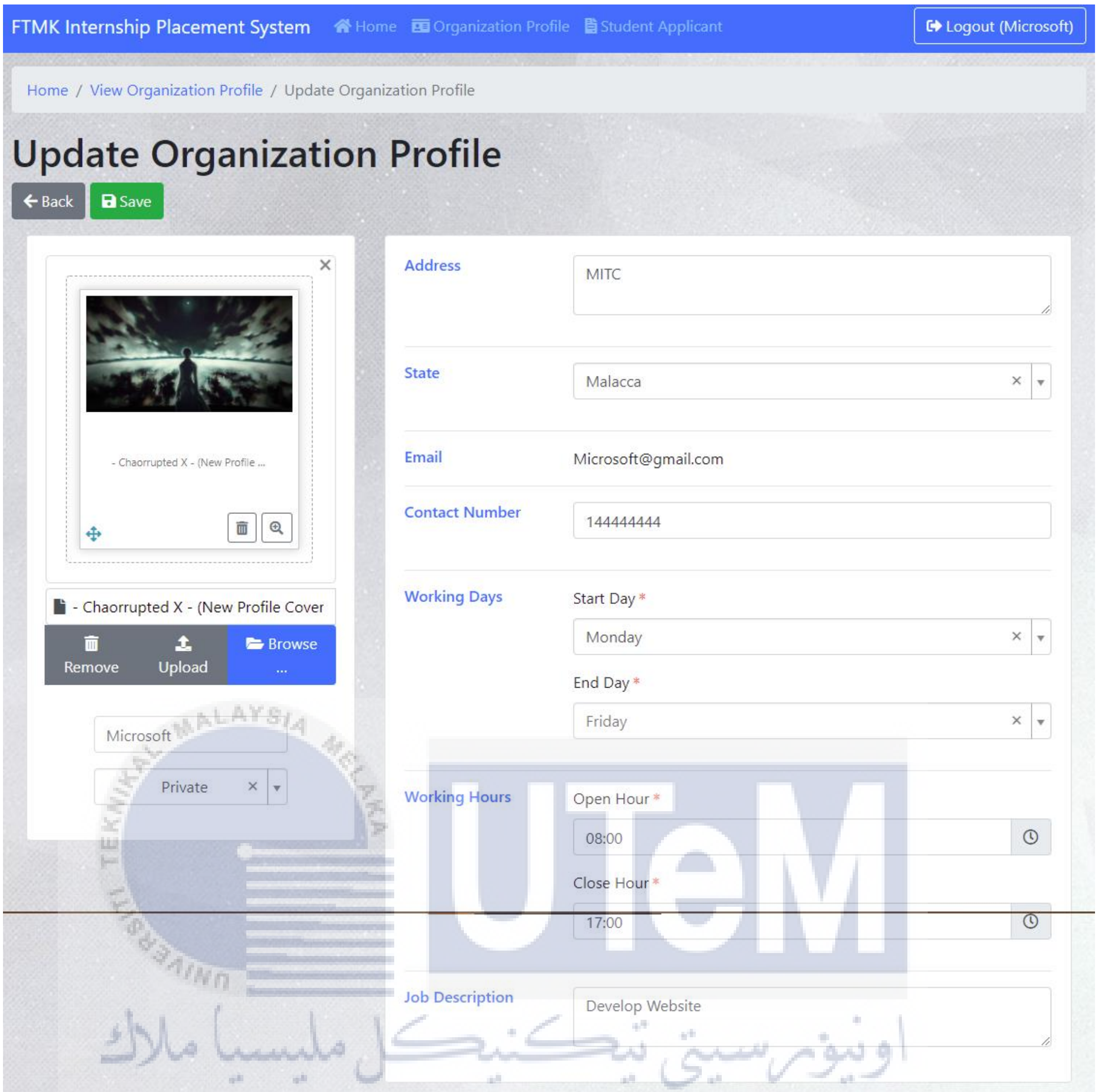
For the skills section, the student needs to input the skill name, and scale it from 0% to 100%.

For the languages section, there will be a dropdown that list the data created from the Language Management module. The scale is also present in the languages section.

For the education section, the student needs to input education name, the start year and end year, and description for the education. The start year and end year will popup the datepicker widget for choosing year.

For the work experience section, the student needs to input workplace name, the start year and end year, and the description to describe more about the work experience.

For the achievement section, the student needs to input the achievement name, the year of achievement, and the description for the achievement. The year input will show the datepicker widget.


FTMK Internship Placement System
Home
Organization Profile
Student Applicant
Logout (Microsoft)
Home / View Organization Profile / Update Organization Profile
Back
Save
Remove
Upload
Browse ...

Update Organization Profile

- Chaorrupted X - (New Profile Cover

Microsoft

Private

Address: MITC

State: Malacca

Email: Microsoft@gmail.com

Contact Number: 144444444

Working Days: Start Day * Monday; End Day * Friday

Working Hours: Open Hour * 08:00; Close Hour * 17:00

Job Description: Develop Website

Figure 4.14: Input Design for Organization Profile (Organization)

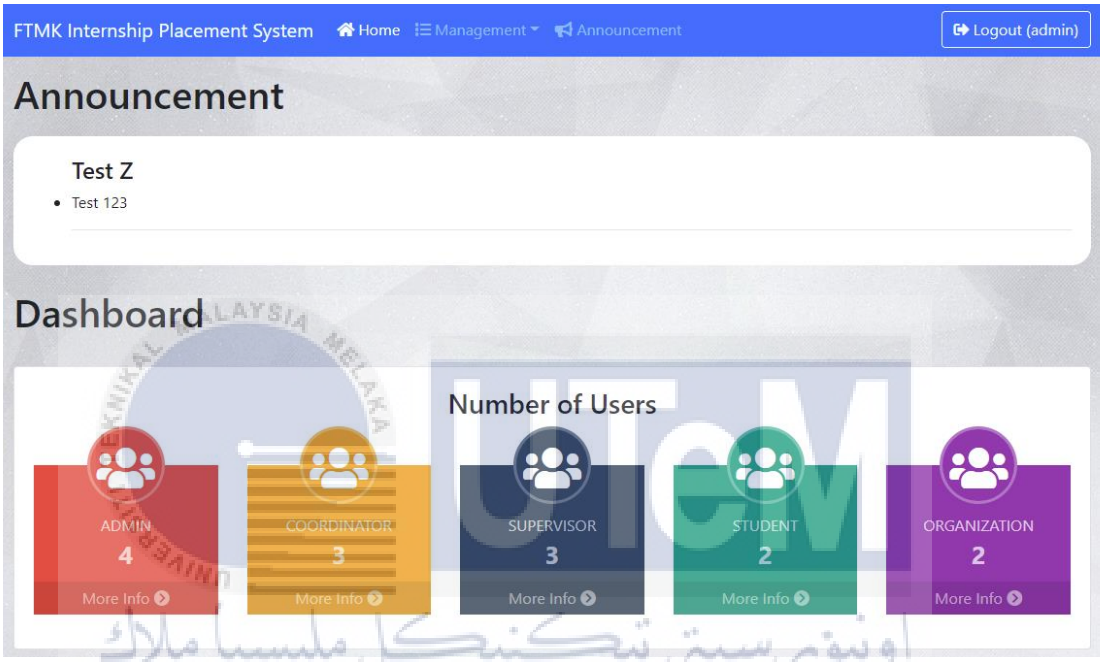
Based on figure 4.14, it shows the input design for organization profile. The organization can upload the logo into the system and input the organization name. The dropdown below the organization name is for choosing the organization type, whether the organization is a government or private sector.

Other than that, the organization can input address, state, contact number, working days, working hours, and job description. The input for working hours will show the timepicker widget to help choose the time.

4.2.2.3 Output Design

The output design is the design of the system that shows the dynamic statistic report generated by the system. The users for FIPS can view the report in the dashboard (home) of the system.

The admin can view number of users registered into the system, based on role assigned. Figure 4.15 shows the dashboard for the system administrator.



Announcement

Test Z

- Test 123

Dashboard

Number of Users

ADMIN	COORDINATOR	SUPERVISOR	STUDENT	ORGANIZATION
4	3	3	2	2

Figure 4.15: Output Design for Admin Dashboard

Coordinator can view the percentage of how many students have supervised and unsupervised. The coordinator also can view number of students under programmes. Figure 4.16 shows the dashboard for coordinator.

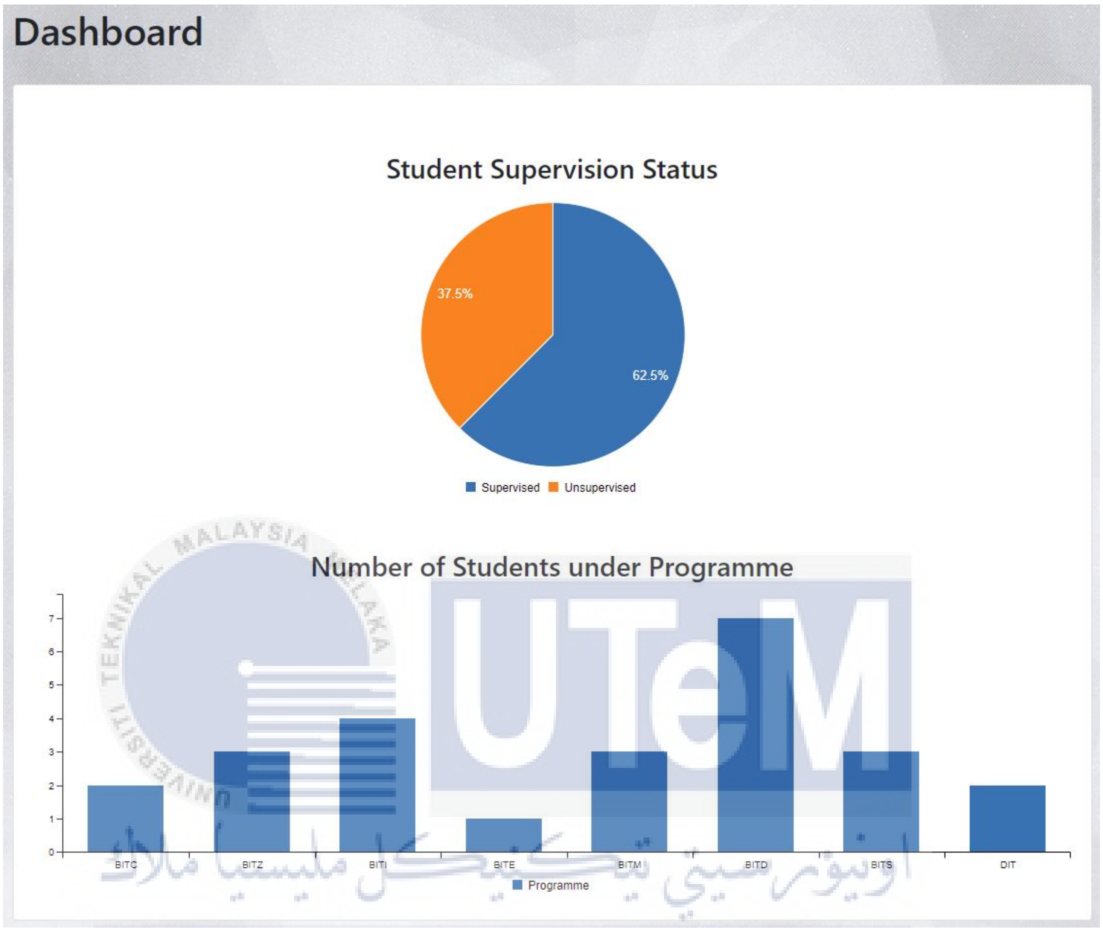
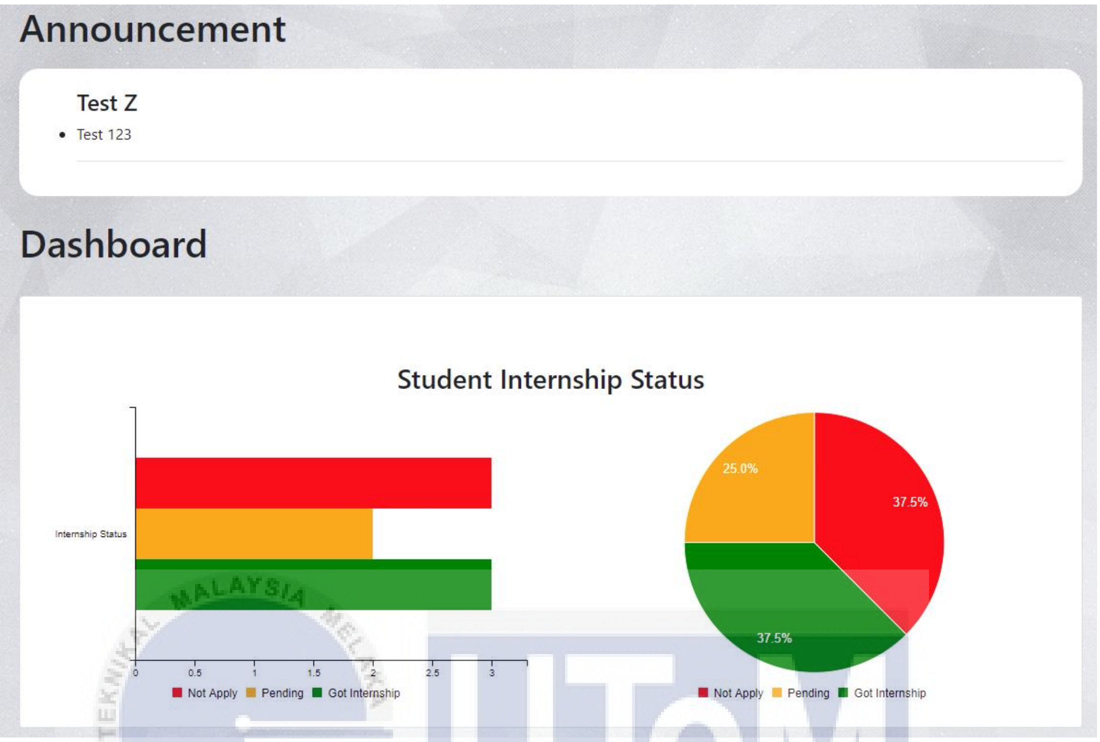

Dashboard
Student Supervision Status
37.5%
62.5%
■ Supervised ■ Unsupervised
Number of Students under Programme
7
6
5
4
3
2
1
0
BITC
BITZ
BITI
BITE
BITM
BITD
BITS
DIT
■ Programme

Figure 4.16: Output Design for Coordinator Dashboard

Supervisor can view the student internship status. Figure 4.17 shows the dashboard for supervisor.



Announcement

Test Z

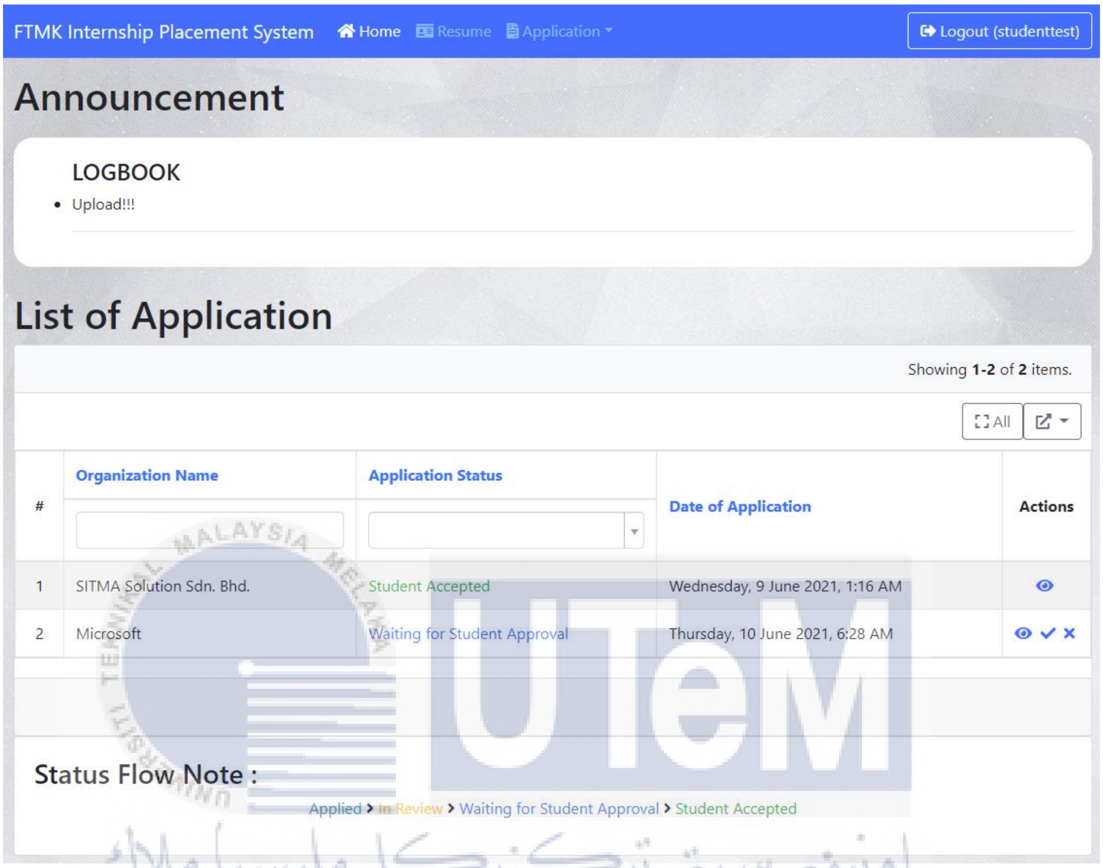
- Test 123

Dashboard

Student Internship Status

Figure 4.17: Output Design for Supervisor Dashboard

The student dashboard contains announcement posted by admin, and the list of application. There is also status flow note display at the bottom. Figure 4.18 shows the dashboard for organization.



Announcement

LOGBOOK

- Upload!!!

List of Application

Showing **1-2** of **2** items.

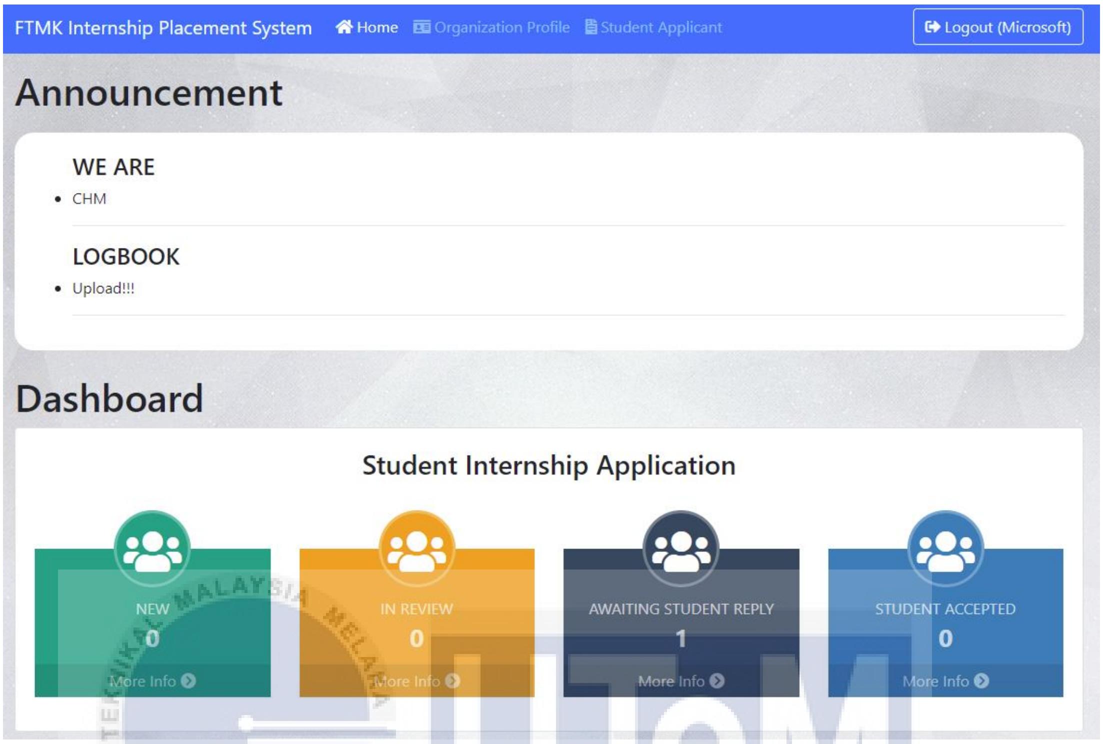
#	Organization Name	Application Status	Date of Application	Actions
1	SITMA Solution Sdn. Bhd.	Student Accepted	Wednesday, 9 June 2021, 1:16 AM	👁
2	Microsoft	Waiting for Student Approval	Thursday, 10 June 2021, 6:28 AM	👁 ✓ ✕

Status Flow Note :

Applied > In Review > Waiting for Student Approval > Student Accepted

Figure 4.18: Output Design for Student Dashboard

For the organization part, the organization can view the status of student internship application. Figure 4.19 shows the dashboard for organization.



Announcement

WE ARE

- CHM

LOGBOOK

- Upload!!!

Dashboard

Student Internship Application

NEW	IN REVIEW	AWAITING STUDENT REPLY	STUDENT ACCEPTED
0	0	1	0

Figure 4.19: Output Design for Organization Dashboard

4.2.3 Database Design

The structuring of data according to a database model is known as database design. This section will address two parts: conceptual database design, which is the process of creating a conceptual representation of a database, as well as the identification of key entities, relationships, and attributes. The Logical Database Design, on the other hand, translates the conceptual representation into the database's logical structure.

4.2.3.1 Conceptual Database Design

The business rules and Entity Relationship Diagram (ERD) of this system demonstrate the flow of data conceptually during the conceptual database design phase.

ERD depicts the conceptual database by displaying the relationships between entities as well as the properties that each entity contains. For FIPS, a total of 19 entities have been discovered (not including the migration entity). Figure 4.20 shows the ERD for the FIPS.

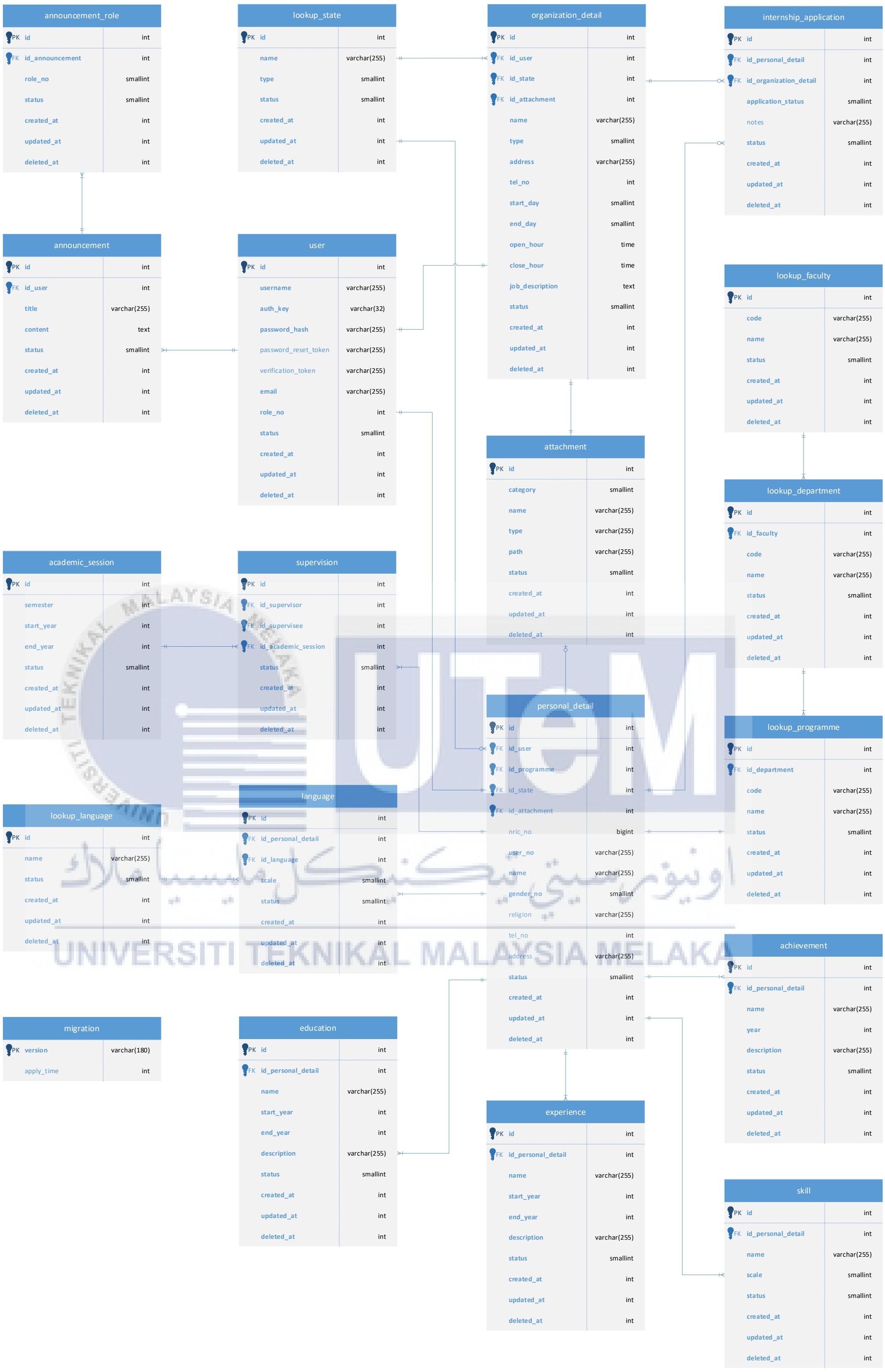


Figure 4.20: Entity Relationship Diagram

Business rules are the description of the organization's policies and procedures. The business rules for FIPS are listed as follows:

i. Relationship between [user], [announcement], and [announcement_role]
   - System Administrator can post many announcements.
   - Each announcement can be assigned to many roles, the announcement will be shown based on the assigned roles.

ii. Relationship between [user], [personal_detail], and [organization_detail]
   - Each user can have one personal detail or one organization detail only.
   - Personal detail is for user with coordinator, supervisor, and student role.
   - Organization detail is for user with organization role only.

iii. Relationship between [personal_detail], [lookup_programme], [lookup_state], [attachment], [supervision], [language], [education], [experience], [skill], and [achievement]
   - Each personal detail can have one programme, and one state.
   - Each student (personal detail) can have one profile picture (attachment) and one report (attachment)
   - Each supervisor (personal detail) can have many students (supervision).
   - Each student (personal detail) can have one supervisor (supervision).
   - Each student (personal detail) can have many languages, educations, experiences, skills, and achievements.

iv. Relationship between [lookup_faculty], [lookup_department], and [lookup_programme]
   - Each faculty can have many departments.
   - Each department can have many programmes.

v. Relationship between [supervision], and [academic_session]
   - Each supervision can be assigned to one academic session.
   - One academic session can have many supervisions.

vi. Relationship between [lookup_language], [language], and [personal_detail]

- Each language can have one lookup language.
- Each lookup language can have many languages.
- The language entity act as a bridge for student (personal_detail) to have many languages.

vii. Relationship between [organization_detail], [lookup_state], [attachment]

- Each organization detail can have one state, and one organization logo (attachment).

viii. Relationship between [personal_detail], [internship_application], [organization_detail]

- Each personal detail can have many internship applications.
- Each organization detail can view and accept or reject many internship applications.

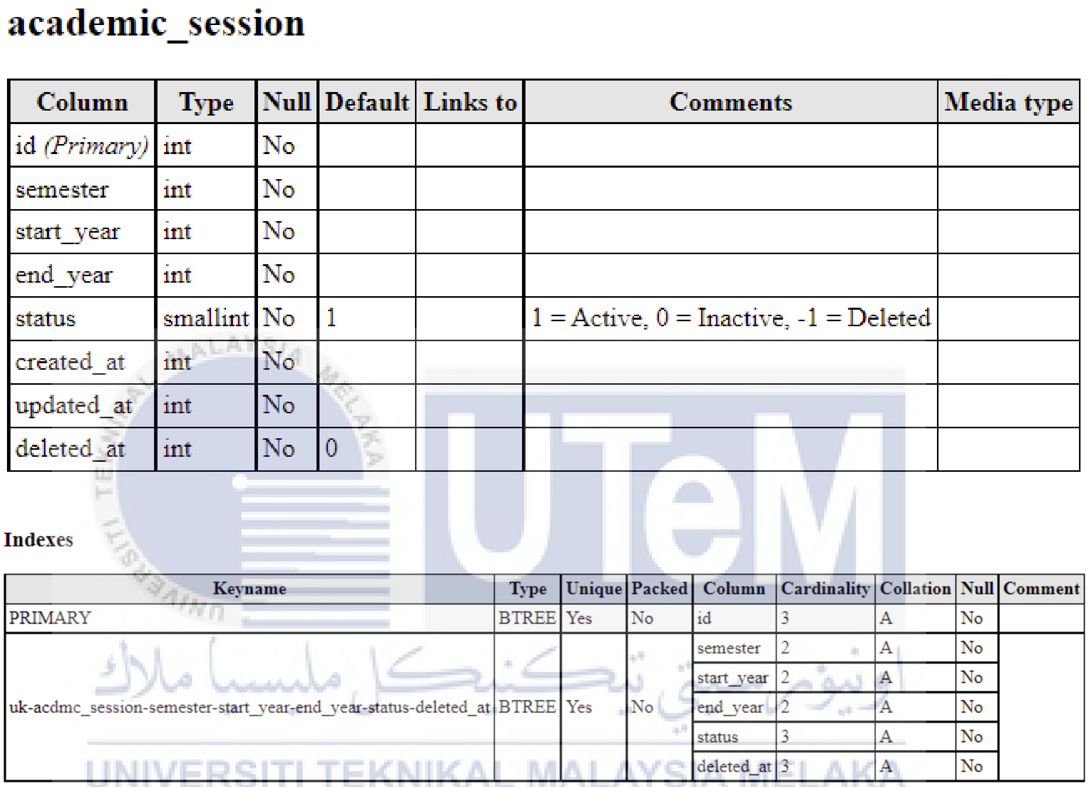
ix. No relationship [migration]

- This entity is used for Yii2 Framework, to keep track of database changes.

4.2.3.2 Logical Database Design

The data dictionary uses logical design to transform the conceptual design into the database's logical structure. Below shows the tables for data dictionary of FIPS.

Table 4.1: Data Dictionary for Academic Session



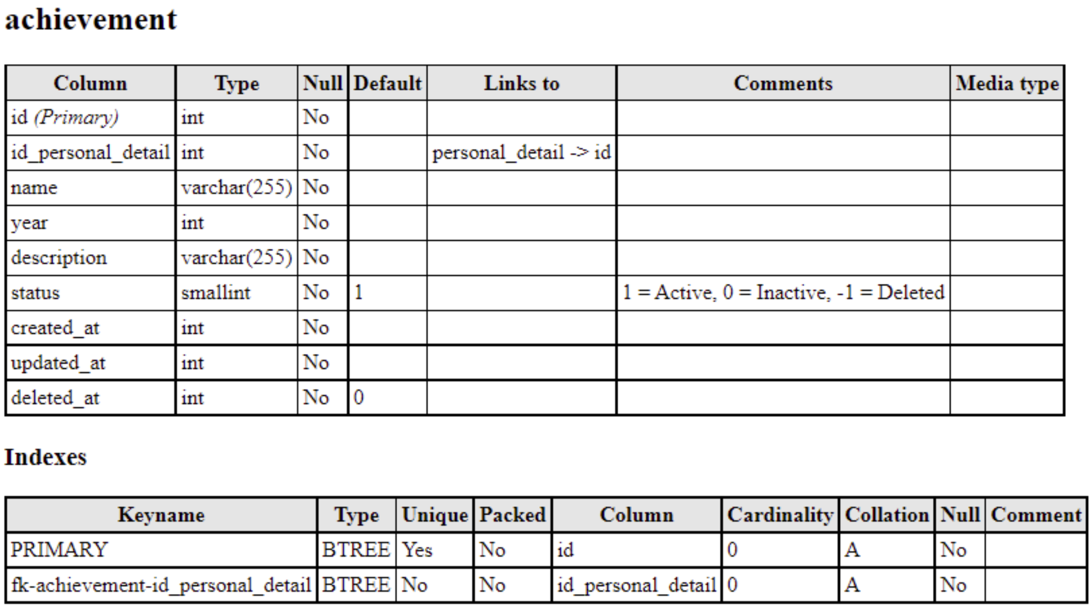
academic_session

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
semester	int	No				
start_year	int	No				
end_year	int	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	3	A	No	
uk-acdmc_session-semester-start_year-end_year-status-deleted_at	BTREE	Yes	No	semester	2	A	No	
				start_year	2	A	No	
				end_year	2	A	No	
				status	3	A	No	
				deleted_at	3	A	No	

Table 4.2: Data Dictionary for Achievement



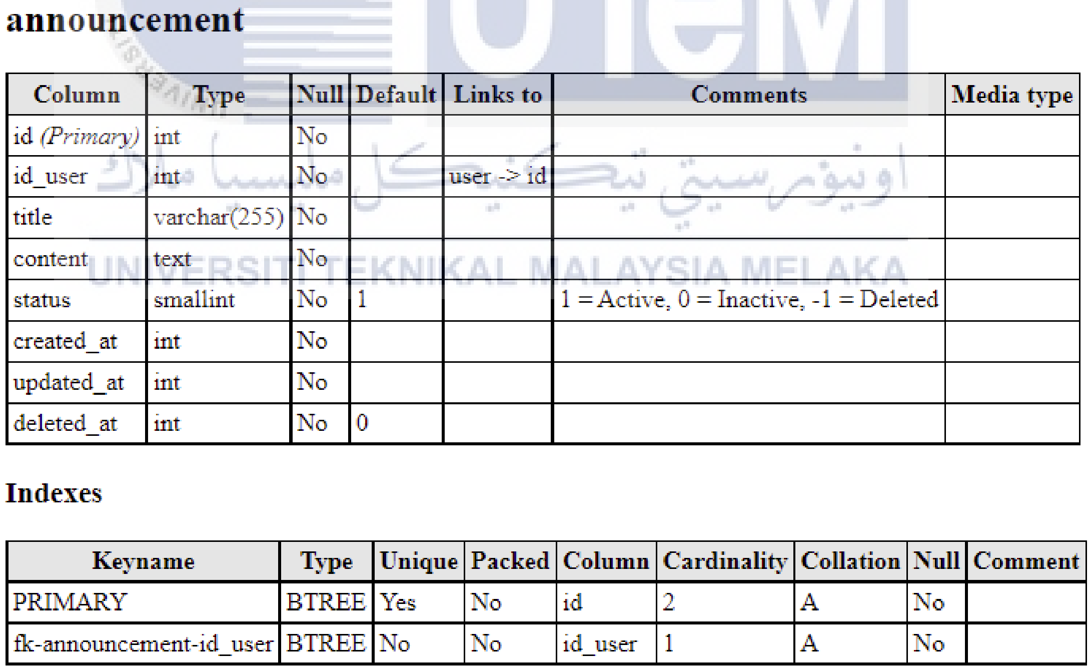
achievement

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
name	varchar(255)	No				
year	int	No				
description	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
fk-achievement-id_personal_detail	BTREE	No	No	id_personal_detail	0	A	No	

Table 4.3: Data Dictionary for Announcement



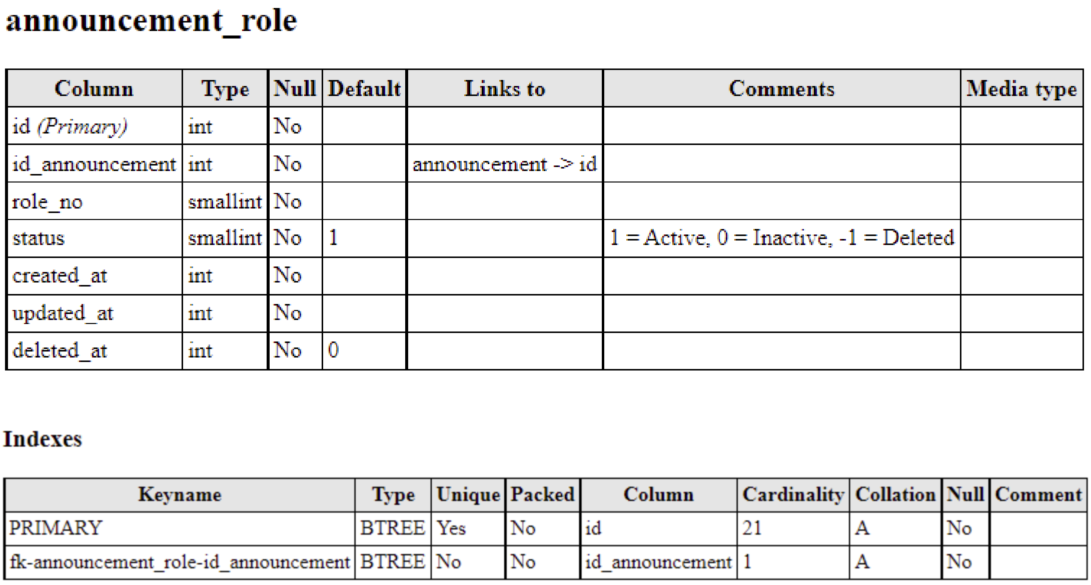
announcement

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_user	int	No		user -> id		
title	varchar(255)	No				
content	text	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	
fk-announcement-id_user	BTREE	No	No	id_user	1	A	No	

Table 4.4: Data Dictionary for Announcement Role



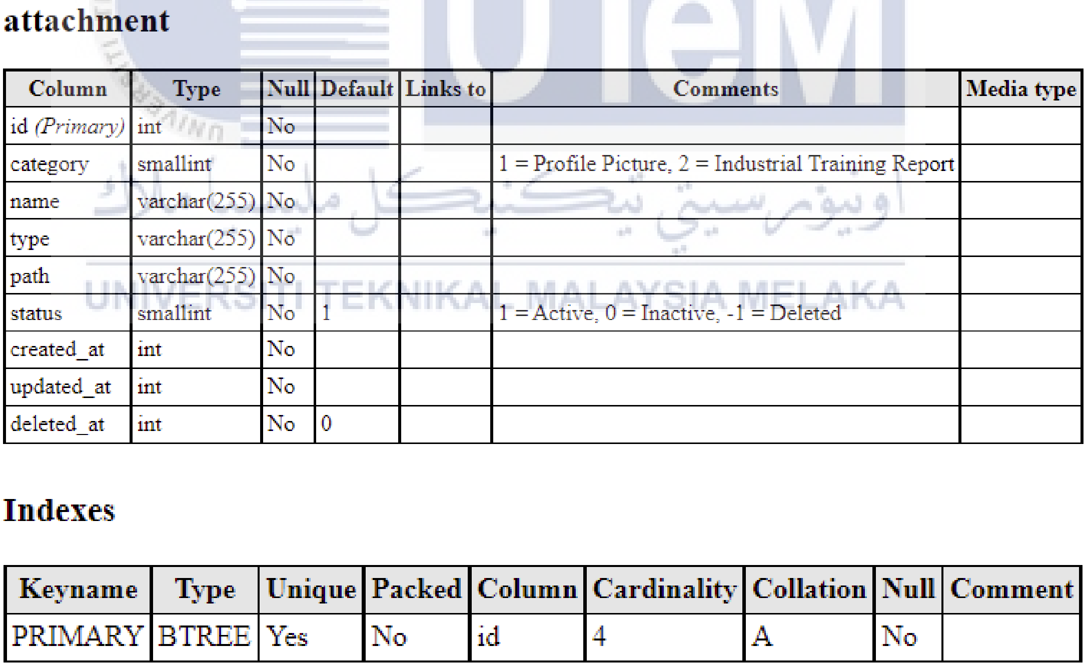
announcement_role

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_announcement	int	No		announcement -> id		
role_no	smallint	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	21	A	No	
fk-announcement_role-id_announcement	BTREE	No	No	id_announcement	1	A	No	

Table 4.5: Data Dictionary for Attachment



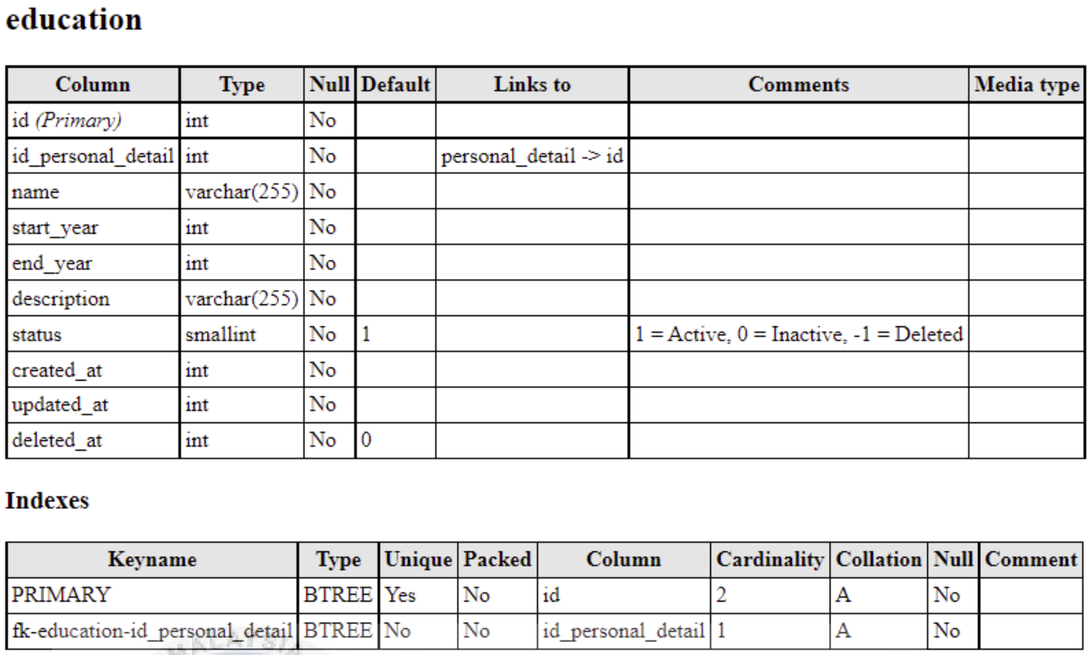
attachment

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
category	smallint	No			1 = Profile Picture, 2 = Industrial Training Report	
name	varchar(255)	No				
type	varchar(255)	No				
path	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	4	A	No	

Table 4.6: Data Dictionary for Education



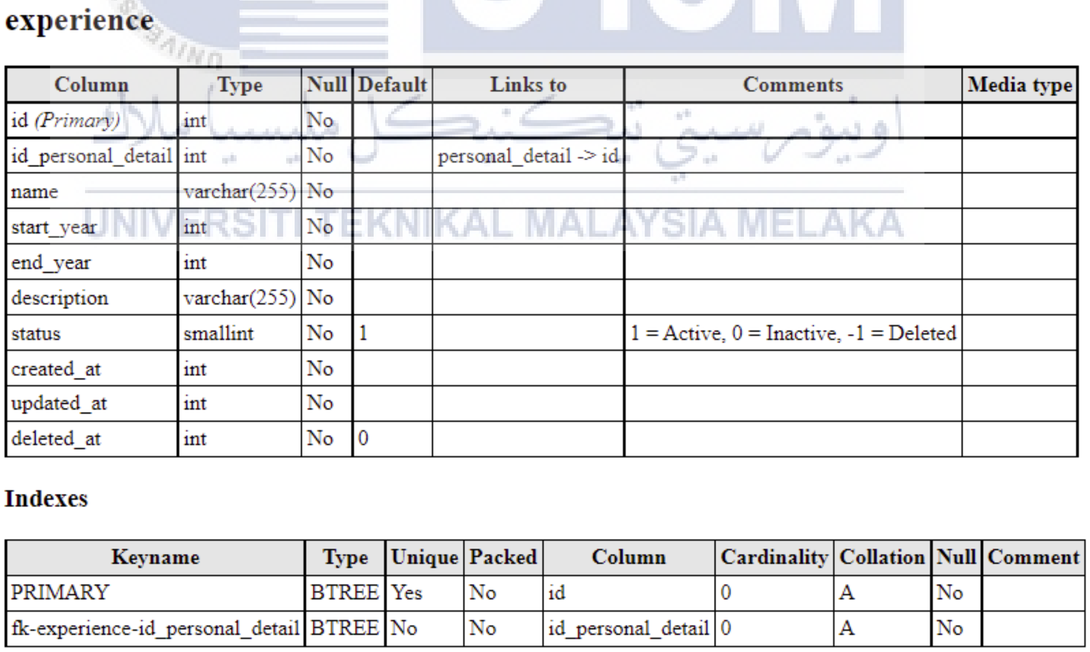
education

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
name	varchar(255)	No				
start_year	int	No				
end_year	int	No				
description	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	
fk-education-id_personal_detail	BTREE	No	No	id_personal_detail	1	A	No	

Table 4.7: Data Dictionary for Experience



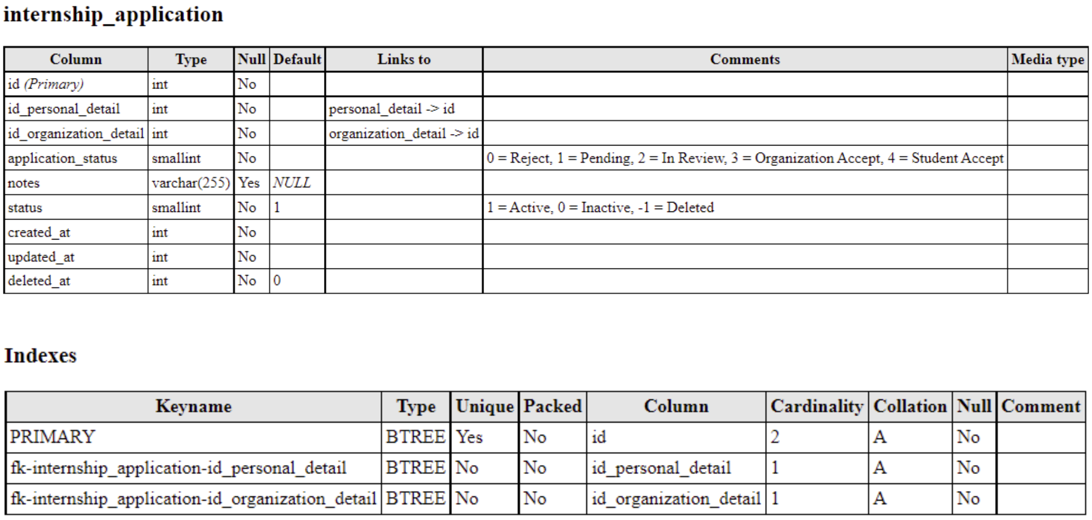
experience

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
name	varchar(255)	No				
start_year	int	No				
end_year	int	No				
description	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
fk-experience-id_personal_detail	BTREE	No	No	id_personal_detail	0	A	No	

Table 4.8: Data Dictionary for Internship Application



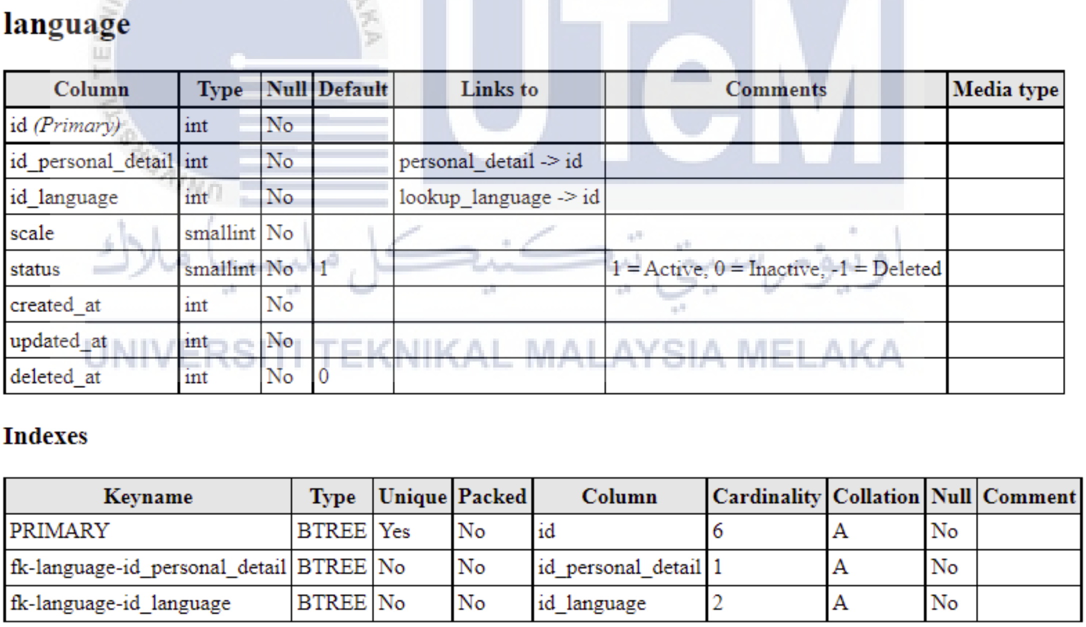
internship_application

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
id_organization_detail	int	No		organization_detail -> id		
application_status	smallint	No			0 = Reject, 1 = Pending, 2 = In Review, 3 = Organization Accept, 4 = Student Accept	
notes	varchar(255)	Yes	*NULL*			
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	
fk-internship_application-id_personal_detail	BTREE	No	No	id_personal_detail	1	A	No	
fk-internship_application-id_organization_detail	BTREE	No	No	id_organization_detail	1	A	No	

Table 4.9: Data Dictionary for Language



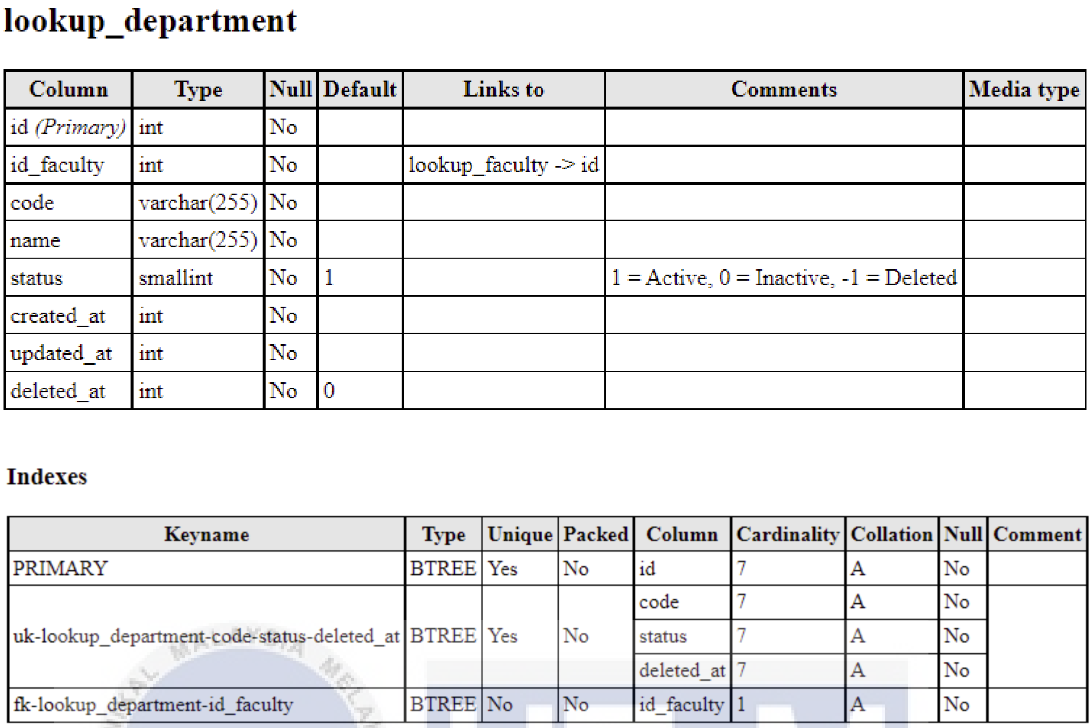
language

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
id_language	int	No		lookup_language -> id		
scale	smallint	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	6	A	No	
fk-language-id_personal_detail	BTREE	No	No	id_personal_detail	1	A	No	
fk-language-id_language	BTREE	No	No	id_language	2	A	No	

Table 4.10: Data Dictionary for Lookup Department



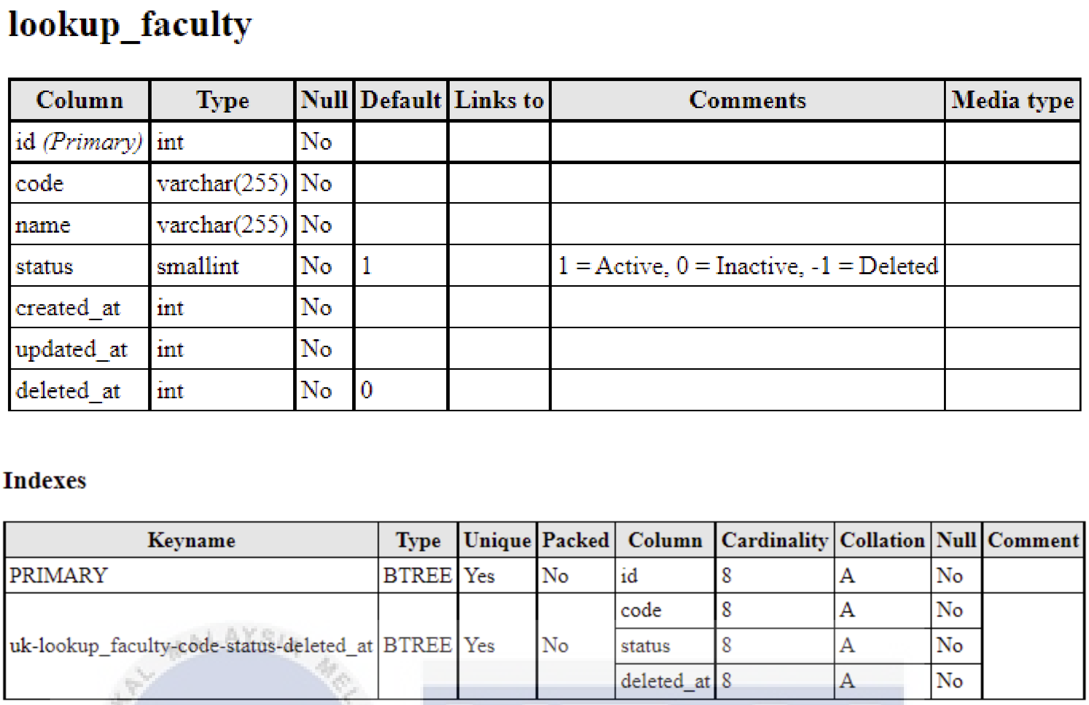
lookup_department

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_faculty	int	No		lookup_faculty -> id		
code	varchar(255)	No				
name	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	7	A	No	
uk-lookup_department-code-status-deleted_at	BTREE	Yes	No	code	7	A	No	
				status	7	A	No	
				deleted_at	7	A	No	
fk-lookup_department-id_faculty	BTREE	No	No	id_faculty	1	A	No	

Table 4.11: Data Dictionary for Lookup Faculty



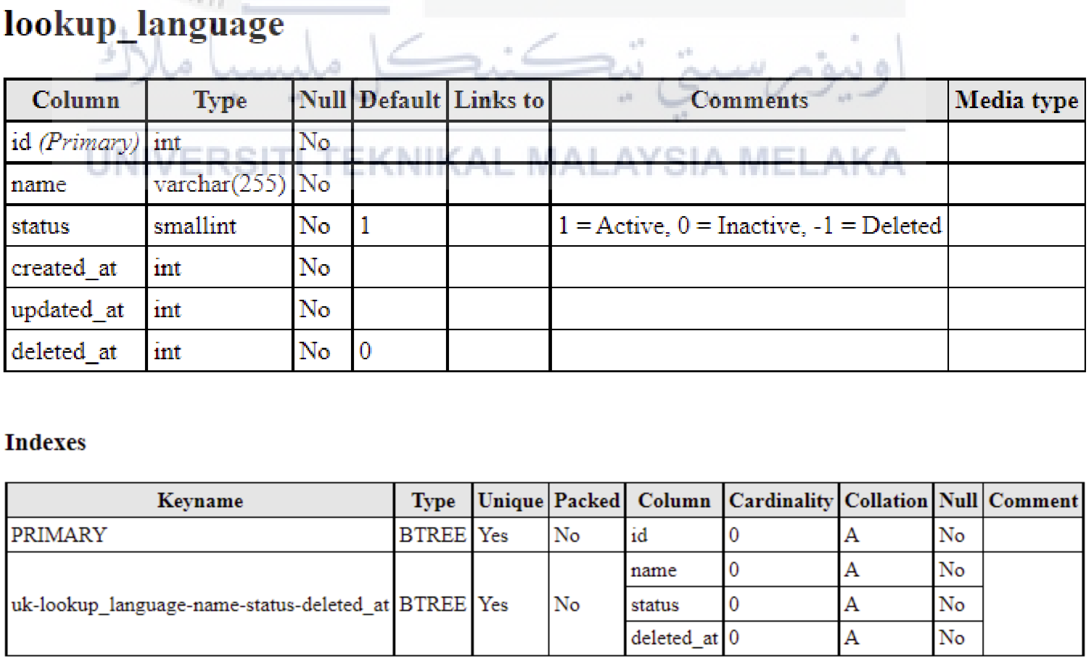
lookup_faculty

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
code	varchar(255)	No				
name	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	8	A	No	
uk-lookup_faculty-code-status-deleted_at	BTREE	Yes	No	code	8	A	No	
				status	8	A	No	
				deleted_at	8	A	No	

Table 4.12: Data Dictionary for Lookup Language



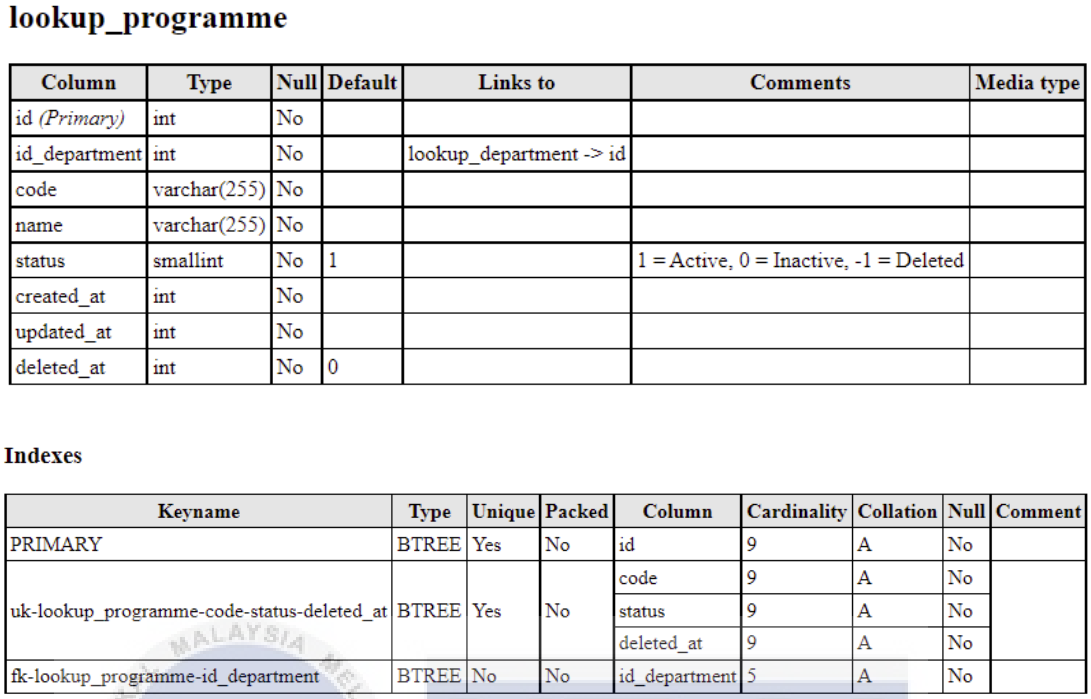
lookup_language

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
name	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
uk-lookup_language-name-status-deleted_at	BTREE	Yes	No	name	0	A	No	
				status	0	A	No	
				deleted_at	0	A	No	

Table 4.13: Data Dictionary for Lookup Programme



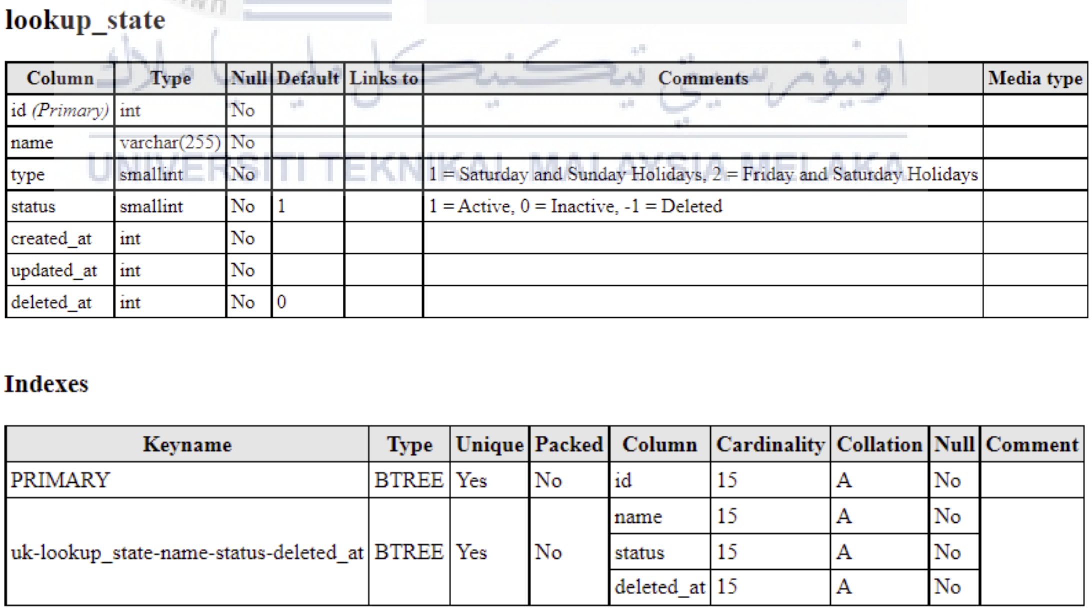
lookup_programme

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_department	int	No		lookup_department -> id		
code	varchar(255)	No				
name	varchar(255)	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	9	A	No	
uk-lookup_programme-code-status-deleted_at	BTREE	Yes	No	code	9	A	No	
				status	9	A	No	
				deleted_at	9	A	No	
fk-lookup_programme-id_department	BTREE	No	No	id_department	5	A	No	

Table 4.14: Data Dictionary for Lookup State



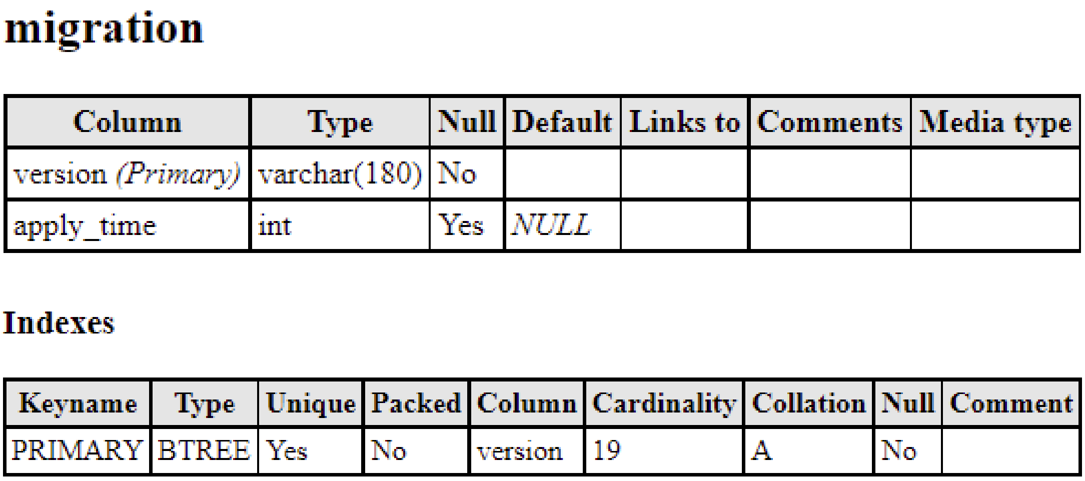
lookup_state

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
name	varchar(255)	No				
type	smallint	No			1 = Saturday and Sunday Holidays, 2 = Friday and Saturday Holidays	
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	15	A	No	
uk-lookup_state-name-status-deleted_at	BTREE	Yes	No	name	15	A	No	
				status	15	A	No	
				deleted_at	15	A	No	

Table 4.15: Data Dictionary for Migration



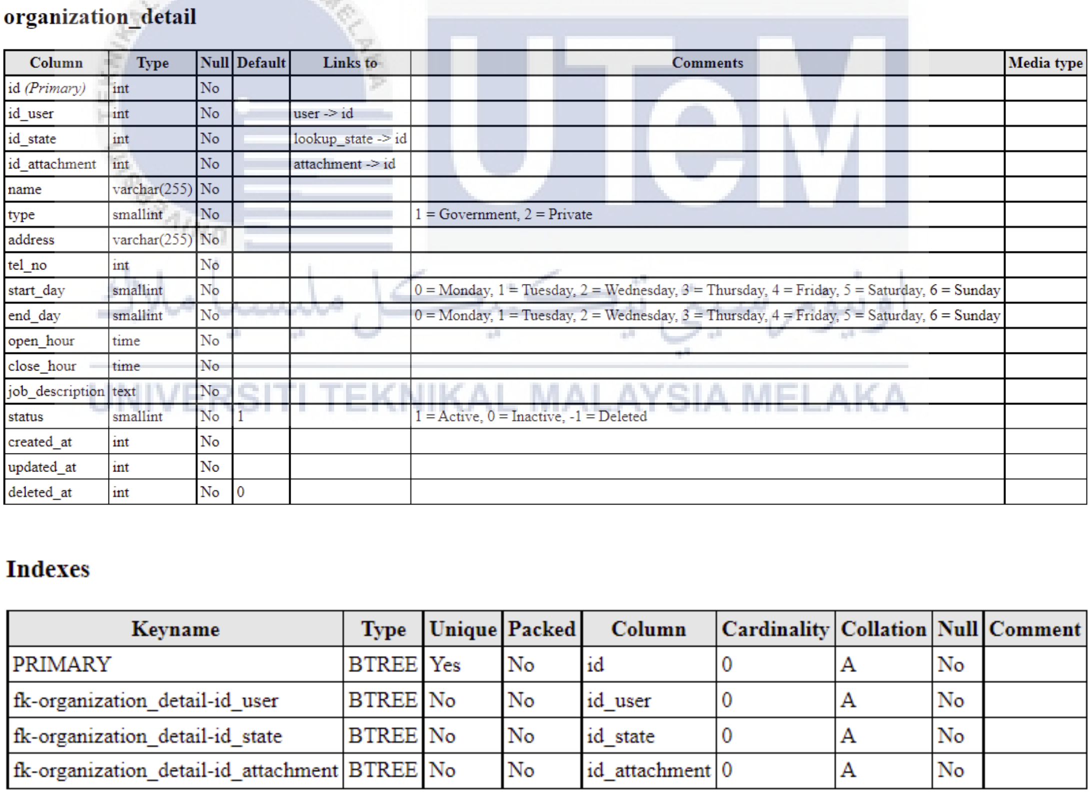
migration

Column	Type	Null	Default	Links to	Comments	Media type
version *(Primary)*	varchar(180)	No				
apply_time	int	Yes	*NULL*			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	version	19	A	No	

Table 4.16: Data Dictionary for Organization Detail



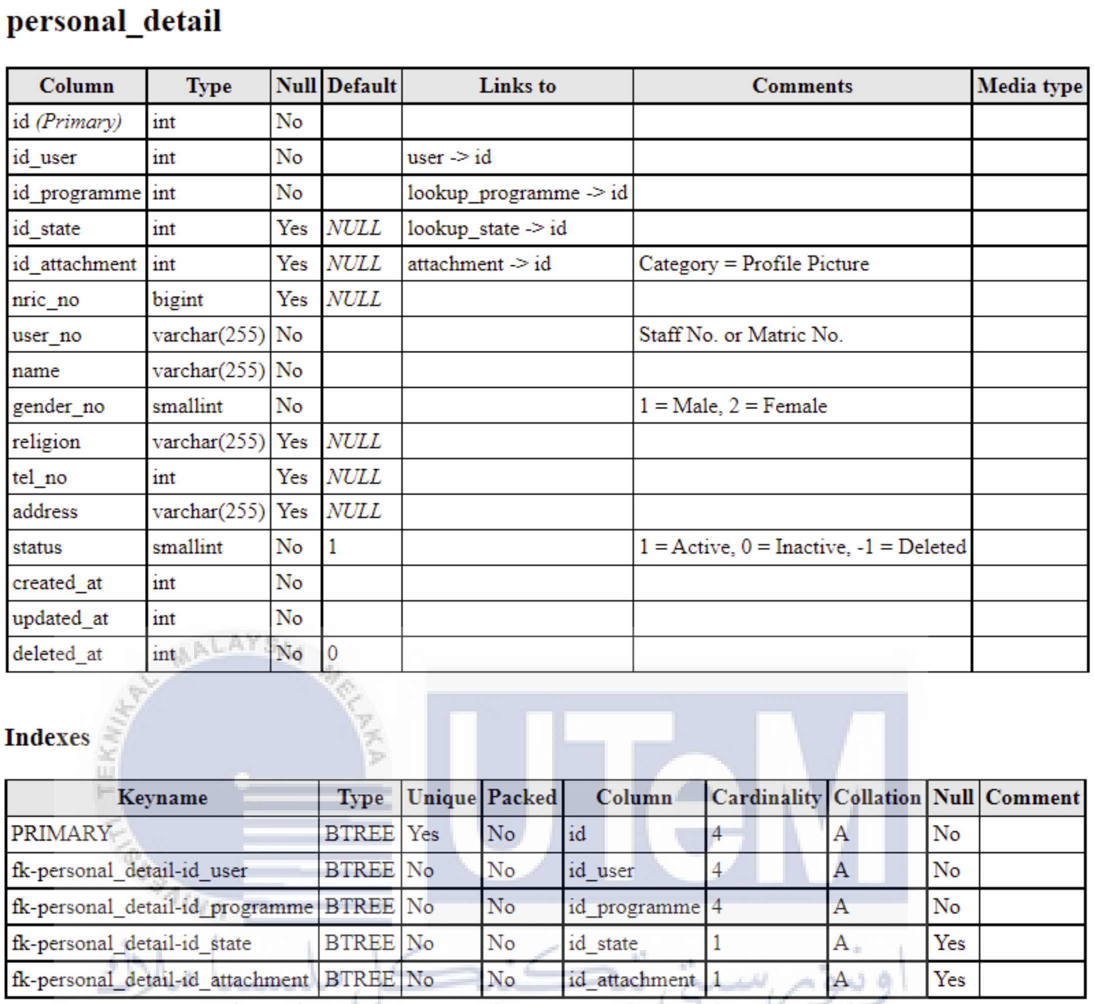
organization_detail

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_user	int	No		user -> id		
id_state	int	No		lookup_state -> id		
id_attachment	int	No		attachment -> id		
name	varchar(255)	No				
type	smallint	No			1 = Government, 2 = Private	
address	varchar(255)	No				
tel_no	int	No				
start_day	smallint	No			0 = Monday, 1 = Tuesday, 2 = Wednesday, 3 = Thursday, 4 = Friday, 5 = Saturday, 6 = Sunday	
end_day	smallint	No			0 = Monday, 1 = Tuesday, 2 = Wednesday, 3 = Thursday, 4 = Friday, 5 = Saturday, 6 = Sunday	
open_hour	time	No				
close_hour	time	No				
job_description	text	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
fk-organization_detail-id_user	BTREE	No	No	id_user	0	A	No	
fk-organization_detail-id_state	BTREE	No	No	id_state	0	A	No	
fk-organization_detail-id_attachment	BTREE	No	No	id_attachment	0	A	No	

Table 4.17: Data Dictionary for Personal Detail



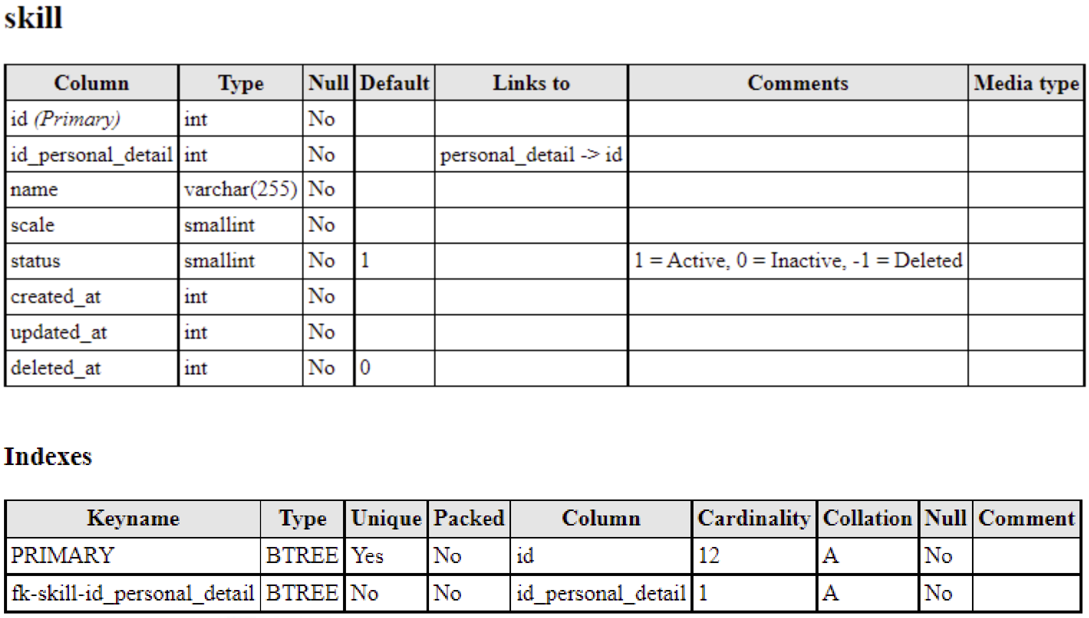
personal_detail

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_user	int	No		user -> id		
id_programme	int	No		lookup_programme -> id		
id_state	int	Yes	*NULL*	lookup_state -> id		
id_attachment	int	Yes	*NULL*	attachment -> id	Category = Profile Picture	
nric_no	bigint	Yes	*NULL*			
user_no	varchar(255)	No			Staff No. or Matric No.	
name	varchar(255)	No				
gender_no	smallint	No			1 = Male, 2 = Female	
religion	varchar(255)	Yes	*NULL*			
tel_no	int	Yes	*NULL*			
address	varchar(255)	Yes	*NULL*			
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	4	A	No	
fk-personal_detail-id_user	BTREE	No	No	id_user	4	A	No	
fk-personal_detail-id_programme	BTREE	No	No	id_programme	4	A	No	
fk-personal_detail-id_state	BTREE	No	No	id_state	1	A	Yes	
fk-personal_detail-id_attachment	BTREE	No	No	id_attachment	1	A	Yes	

Table 4.18: Data Dictionary for Skill



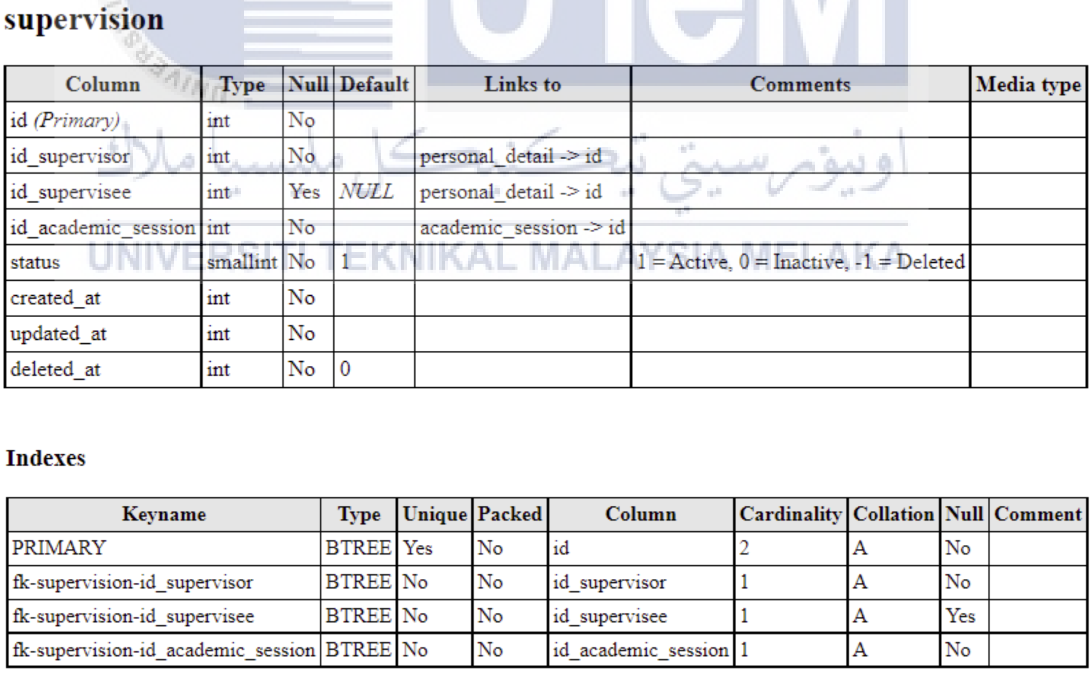
skill

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_personal_detail	int	No		personal_detail -> id		
name	varchar(255)	No				
scale	smallint	No				
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	12	A	No	
fk-skill-id_personal_detail	BTREE	No	No	id_personal_detail	1	A	No	

Table 4.19: Data Dictionary for Supervision



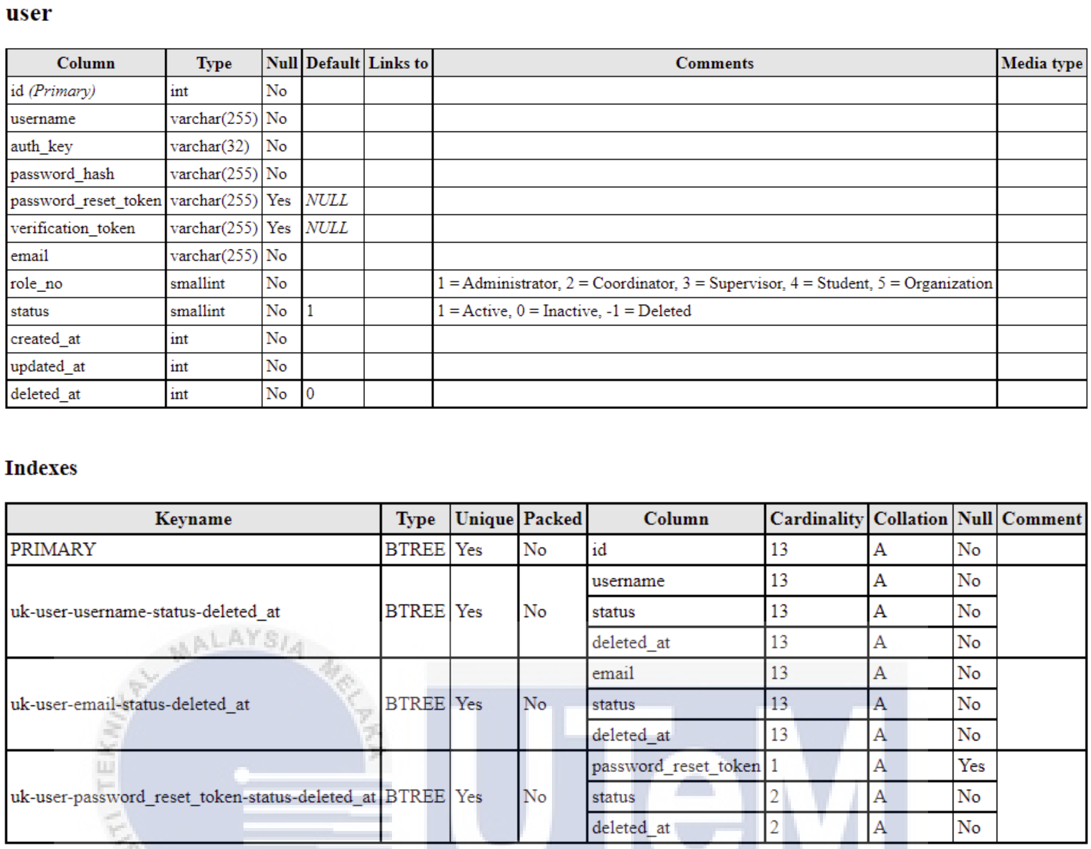
supervision

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
id_supervisor	int	No		personal_detail -> id		
id_supervisee	int	Yes	*NULL*	personal_detail -> id		
id_academic_session	int	No		academic_session -> id		
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	
fk-supervision-id_supervisor	BTREE	No	No	id_supervisor	1	A	No	
fk-supervision-id_supervisee	BTREE	No	No	id_supervisee	1	A	Yes	
fk-supervision-id_academic_session	BTREE	No	No	id_academic_session	1	A	No	

Table 4.20: Data Dictionary for User



user

Column	Type	Null	Default	Links to	Comments	Media type
id *(Primary)*	int	No				
username	varchar(255)	No				
auth_key	varchar(32)	No				
password_hash	varchar(255)	No				
password_reset_token	varchar(255)	Yes	*NULL*			
verification_token	varchar(255)	Yes	*NULL*			
email	varchar(255)	No				
role_no	smallint	No			1 = Administrator, 2 = Coordinator, 3 = Supervisor, 4 = Student, 5 = Organization	
status	smallint	No	1		1 = Active, 0 = Inactive, -1 = Deleted	
created_at	int	No				
updated_at	int	No				
deleted_at	int	No	0			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	13	A	No	
uk-user-username-status-deleted_at	BTREE	Yes	No	username	13	A	No	
				status	13	A	No	
				deleted_at	13	A	No	
uk-user-email-status-deleted_at	BTREE	Yes	No	email	13	A	No	
				status	13	A	No	
				deleted_at	13	A	No	
uk-user-password_reset_token-status-deleted_at	BTREE	Yes	No	password_reset_token	1	A	Yes	
				status	2	A	No	
				deleted_at	2	A	No	

4.3 Detailed Design

This part goes through the software design as well as the physical database design to prepare for the implementation phase.

4.3.1 Software Design

Software design is a technique for giving programmers more comprehensive instructions on how to code the system's components. The properties and methods of a class are described in the class description. Below shows the tables of description for each class of the FIPS.

Table 4.21: Class Description for Login Form

LOGIN FORM	
Responsibility	This will prompt for user login information
Attributes	username, password
INPUT	
Responsibility	To input the user login detail
Input Parameters	None
Output Parameters	None
I/O Table	user
Pre-condition	Display input for login details
Post-condition	Send the input data to authenticate user details

Algorithm	BEGIN
	INPUT: username
	INPUT: password
	validateLogin();
	END

Table 4.22: Class Description for User

USER	
Responsibility	This will prompt for user registration information
Attributes	username, email, password, role
INPUT	
Responsibility	To register user into the system
Input Parameters	None
Output Parameters	None
I/O Table	user, personal_detail
Pre-condition	Display input for user registration
Post-condition	Insert user registration details into the database
Algorithm	BEGIN INPUT: username INPUT: email INPUT: password INPUT: role save(); END

Table 4.23: Class Description for Lookup Department

LOOKUP DEPARTMENT	
Responsibility	This will prompt for department information
Attributes	code, name
INPUT	
Responsibility	To register department into the system
Input Parameters	None
Output Parameters	None
I/O Table	lookup_faculty, lookup_department
Pre-condition	Display input for creating department
Post-condition	Insert department details into the database

Algorithm	BEGIN
	    INPUT: code
	    INPUT: name
	    IF code IS UNIQUE THEN
	        save();
	    ENDIF
	END

Table 4.24: Class Description for Lookup Programme

LOOKUP PROGRAMME	
Responsibility	This will prompt for programme information
Attributes	department_code, code, name
INPUT	
Responsibility	To register programme into the system
Input Parameters	None
Output Parameters	None
I/O Table	lookup_faculty, lookup_department, lookup_programme
Pre-condition	Display input for creating programme
Post-condition	Insert programme details into the database

Algorithm	BEGIN
	INPUT: department_code
	INPUT: code
	INPUT: name
	IF code IS UNIQUE THEN
	save();
	ENDIF
	END

Table 4.25: Class Description for Lookup State

LOOKUP STATE	
Responsibility	This will prompt for state information
Attributes	name, type
INPUT	
Responsibility	To register state into the system
Input Parameters	None
Output Parameters	None
I/O Table	lookup_state
Pre-condition	Display input for creating state
Post-condition	Insert state details into the database

Algorithm	BEGIN
	    INPUT: name
	    INPUT: type
	    IF name IS UNIQUE THEN
	        save();
	    ENDIF
	END

Table 4.26: Class Description for Lookup Language

LOOKUP LANGUAGE	
Responsibility	This will prompt for language information
Attributes	name
INPUT	
Responsibility	To register language into the system
Input Parameters	None
Output Parameters	None
I/O Table	lookup_language
Pre-condition	Display input for creating language
Post-condition	Insert language name into the database

Algorithm	BEGIN
	    INPUT: name
	    IF name IS UNIQUE THEN
	        save();
	    ENDIF
	END

Table 4.27: Class Description for Academic Session

ACADEMIC SESSION	
Responsibility	This will prompt for academic session information
Attributes	semester, start_year, end_year, status
INPUT	
Responsibility	To register academic session into the system
Input Parameters	None
Output Parameters	None
I/O Table	academic_session
Pre-condition	Display input for creating academic session
Post-condition	Insert academic session details into the database

Algorithm	BEGIN
	INPUT: semester
	INPUT: start_year
	INPUT: end_year
	INPUT: status
	IF semester AND start_year AND end_year AND status IS UNIQUE THEN
	save();
	ENDIF
	END

Table 4.28: Class Description for Announcement

ANNOUNCEMENT	
Responsibility	This will prompt for announcement information
Attributes	title, content, role_name
INPUT	
Responsibility	To post announcement
Input Parameters	None
Output Parameters	None
I/O Table	announcement, role_announcement
Pre-condition	Display input for posting announcement
Post-condition	Display announcement in home based on role

Algorithm	BEGIN
	INPUT: title
	INPUT: content
	INPUT: role_name
	save();
	END

Table 4.29: Class Description for Supervision

SUPERVISION	
Responsibility	This will prompt for supervision information
Attributes	id_supervisor, id_supervisee, id_academic_session
INPUT	
Responsibility	To assign supervisor with academic session
Input Parameters	None
Output Parameters	None
I/O Table	supervision, personal_detail, academic_session
Pre-condition	Display input to assign supervisor with academic session
Post-condition	Supervisor assigned with academic session

Algorithm	BEGIN
	INPUT: id_supervisor
	INPUT: id_academic_session
	save();
	END

DISPLAY	
Responsibility	To display list of students under supervisor
Input Parameters	None
Output Parameters	None
I/O Table	supervision, personal_detail, academic_session
Pre-condition	Fetch list of students under supervisor from database
Post-condition	Display list of students under supervisor

Algorithm	BEGIN
	listStudentSupervisor();
	END

Table 4.30: Class Description for Personal Detail

PERSONAL DETAIL	
Responsibility	This will prompt for personal detail information
Attributes	id_attachment, tel_no, address
INPUT	
Responsibility	To update student personal detail
Input Parameters	None
Output Parameters	None
I/O Table	personal_detail
Pre-condition	Display input for personal detail information
Post-condition	Insert personal detail information into the database

Algorithm	BEGIN
	    getUploadedImageInstance();
	    INPUT: tel_no
	    INPUT: address
	    save();
	END

Table 4.31: Class Description for Skill

SKILL	
Responsibility	This will prompt for skill information
Attributes	name, scale
INPUT	
Responsibility	To update student skills
Input Parameters	None
Output Parameters	None
I/O Table	skill
Pre-condition	Display input for skill information
Post-condition	Insert skill information into the database

Algorithm	BEGIN
	INPUT: name
	INPUT: scale
	save();
	END

Table 4.32: Class Description for Language

LANGUAGE	
Responsibility	This will prompt for language information
Attributes	name, scale
INPUT	
Responsibility	To update student language
Input Parameters	None
Output Parameters	None
I/O Table	language, lookup_language
Pre-condition	Display input for language information
Post-condition	Insert language information into the database

Algorithm	BEGIN
	INPUT: name
	INPUT: scale
	save();
	END

Table 4.33: Class Description for Education

EDUCATION	
Responsibility	This will prompt for education information
Attributes	name, start_year, end_year, description
INPUT	
Responsibility	To update student education
Input Parameters	None
Output Parameters	None
I/O Table	education
Pre-condition	Display input for education information
Post-condition	Insert education information into the database

Algorithm	BEGIN
	    INPUT: name
	    INPUT: start_year
	    INPUT: end_year
	    INPUT: description
	    save();
	END

Table 4.34: Class Description for Experience

EXPERIENCE	
Responsibility	This will prompt for experience information
Attributes	name, start_year, end_year, description
INPUT	
Responsibility	To update student work experience
Input Parameters	None
Output Parameters	None
I/O Table	experience
Pre-condition	Display input for experience information
Post-condition	Insert experience information into the database

Algorithm	BEGIN
	    INPUT: name
	    INPUT: start_year
	    INPUT: end_year
	    INPUT: description
	    save();
	END

Table 4.35: Class Description for Achievement

ACHIEVEMENT	
Responsibility	This will prompt for achievement information
Attributes	name, year, description
INPUT	
Responsibility	To update student achievement
Input Parameters	None
Output Parameters	None
I/O Table	experience
Pre-condition	Display input for achievement information
Post-condition	Insert achievement information into the database

Algorithm	BEGIN
	INPUT: name
	INPUT: year
	INPUT: description
	save();
	END

Table 4.36: Class Description for Organization Detail

ORGANIZATION DETAIL	
Responsibility	This will prompt for organization detail information
Attributes	id_attachment, name, type, address, id_state, tel_no, start_day, end_day, open_hour, close_hour, job_description
INPUT	
Responsibility	To update organization profile
Input Parameters	None
Output Parameters	None
I/O Table	organization_detail
Pre-condition	Display input for organization detail information
Post-condition	Insert organization detail information into the database

Algorithm	BEGIN
	getUploadedImageInstance();
	INPUT: name
	INPUT: type
	INPUT: address
	INPUT: id_state
	INPUT: tel_no
	INPUT: start_day
	INPUT: end_day
	INPUT: open_hour
	INPUT: close_hour
	INPUT: job_description
	save();
	END

4.3.2 Physical Database Design

According to the conceptual and logical design, physical database design is to design the schema level, which is the Data Definition Language (DDL). Create the fundamental structure for the system database using DDL.

The DDL is used to define the database structure, schema, and sub-schema components for the database. The database for FIPS is created with the name ‘ftmk_internship_placement_system’.

There are 20 total tables in the FIPS database. The database is using soft delete, where it will use the attribute status and deleted_at, to indicate that the data is deleted. Since the database is using soft delete, any attribute that is unique, need to have a composite index with the attribute status and deleted_at, to ensure the data that are active and deleted are different. Appendix C shows the complete DDL for the FIPS.

4.4 Conclusion

Finally, the design phase is utilized to define or refine the results of the preliminary design analysis and the details design outcome. A high-level design, which includes system architecture, user interface design, and database design, is also described in this chapter. The architecture of the system is viewed in system architecture. In client-server architecture, the architectural view represents. The navigation bar is one of the sorts of navigation flow that is utilized for this. The input design was used to define the information on the screen and to input data. The output design specifies the different sorts of output, such as a dynamic statistic report.

The database design process included both conceptual and logical considerations. The business rules for FIPS were developed during the conceptual database design, and the findings were utilized to model the entities' relationships and characteristics using the ERD. The data dictionary was then created.

The output from the design process will be utilized as the basis for the next chapter, Chapter 5.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

This chapter explains about the implementation that include in this project. The implementation phase involves adapting and applying many of strategies that the team development uses in the project including gather requirements, analysis, design, development, and testing. In this chapter, several aspects of the implementation such as software development environment setup and software configuration management will be explained in detailed.

There are two phases in the software configuration management that are configuration environment setup and version control procedure. The configuration environment setup will configure at both client computer and server and install support tools to run the system. The version control procedures describe the procedures involved which will periodically upgrade and generate latest version of the system.

At the end of the implementation phase, a complete functional system will be produced. This would then be sent for testing to identify bugs and debugging.

5.2 Software Development Environment Setup

The software development environment setup section provides a detailed run down of the platform, required hardware installation, software components and programming language use to implement this system. Figure 5.1 shows the deployment diagram for the FIPS.

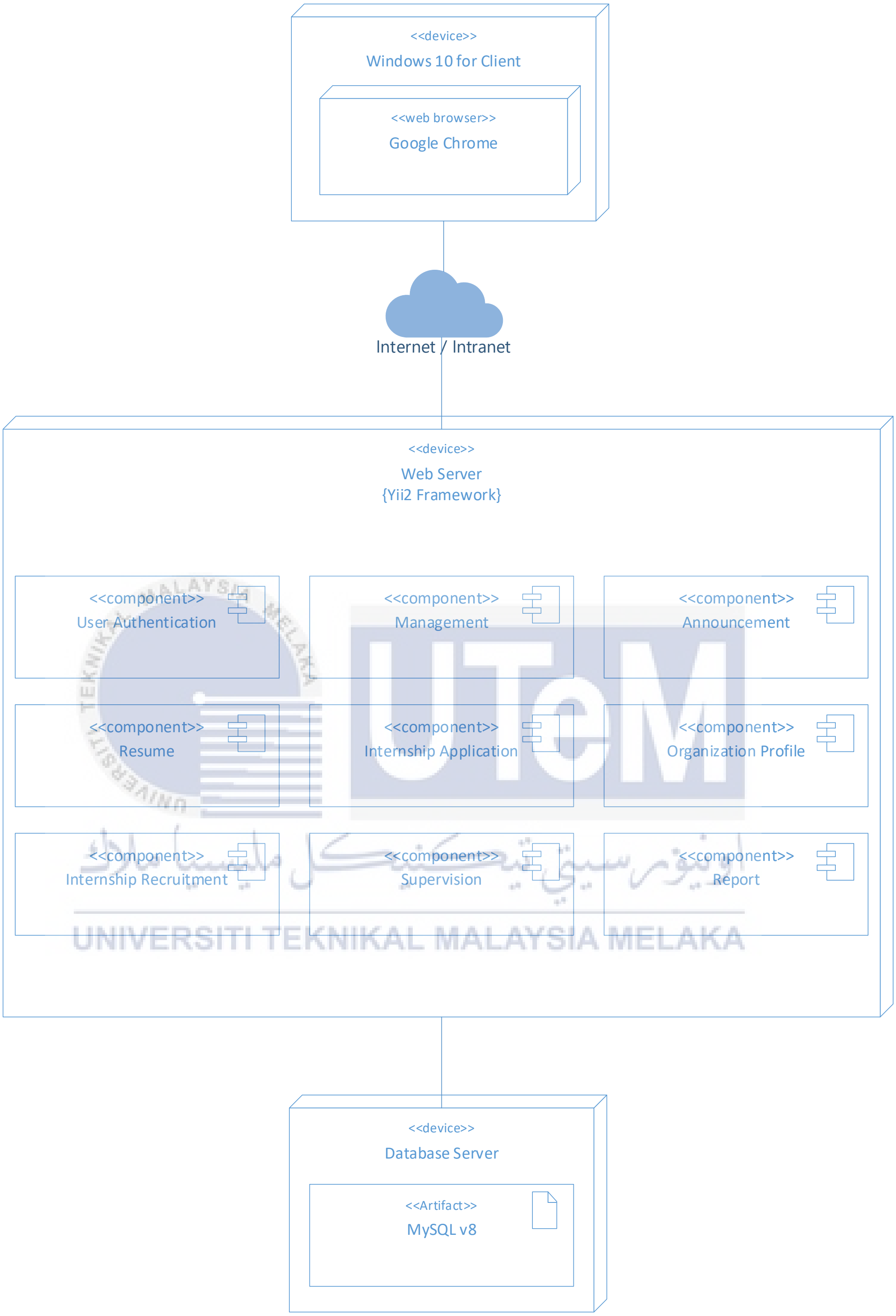


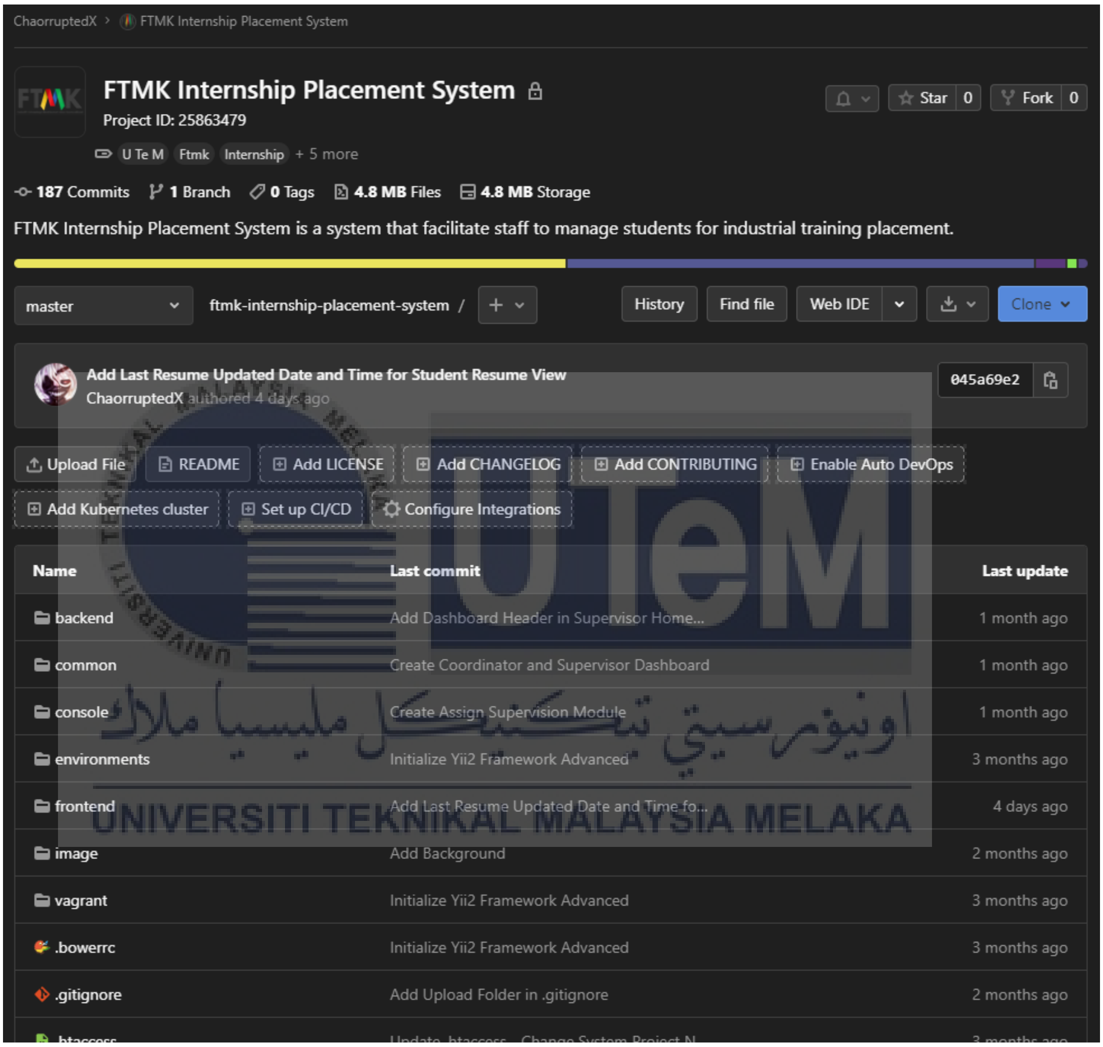
Figure 5.1: Deployment Diagram

5.3 Software Configuration Management

Software configuration management is a very important aspect in software development life cycle as it manages the evolving of system after many different versions of the software are created. The changes that have been implemented and how these changes have been included in the software must be kept in track. This section explains how the configuration management being designed and setup in the project and procedure in managing the source code version.

5.3.1 Configuration Environment Setup

Git is a distributed revision control system, while GitLab is a DevOps lifecycle tool that includes a Git repository over the web. During the development, the FIPS use GitLab to track and control changes in the system coding. The coding of the FIPS is available in the GitLab repository. Figure 5.2 shows the GitLab interface for FIPS.



FTMK Internship Placement System
Project ID: 25863479

FTMK Internship Placement System is a system that facilitate staff to manage students for industrial training placement.

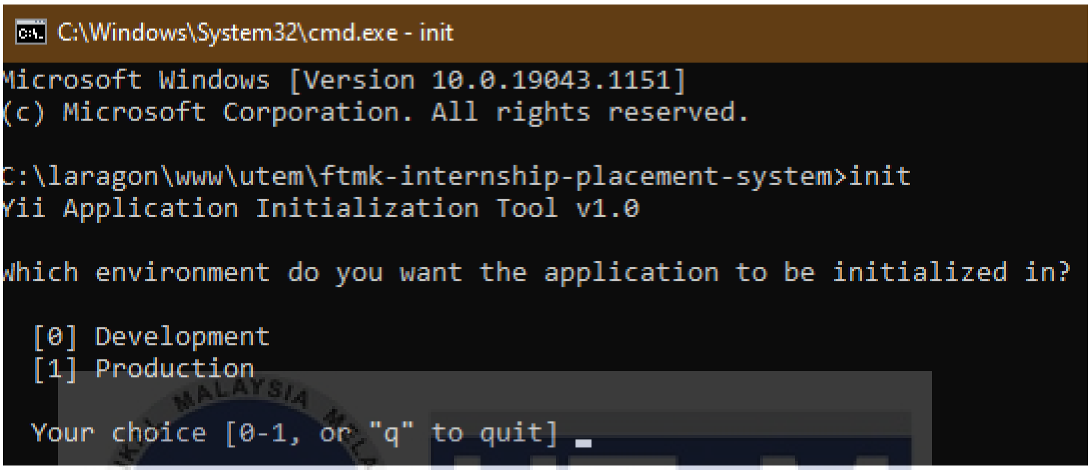
Add Last Resume Updated Date and Time for Student Resume View
ChaorruptedX authored 4 days ago

Name	Last commit	Last update
backend	Add Dashboard Header in Supervisor Home...	1 month ago
common	Create Coordinator and Supervisor Dashboard	1 month ago
console	Create Assign Supervision Module	1 month ago
environments	Initialize Yii2 Framework Advanced	3 months ago
frontend	Add Last Resume Updated Date and Time fo...	4 days ago
image	Add Background	2 months ago
vagrant	Initialize Yii2 Framework Advanced	3 months ago
.bowerrc	Initialize Yii2 Framework Advanced	3 months ago
.gitignore	Add Upload Folder in .gitignore	2 months ago
.htaccess	Update .htaccess - Change System Project N...	3 months ago

Figure 5.2: GitLab

5.3.2 Version Control Procedure

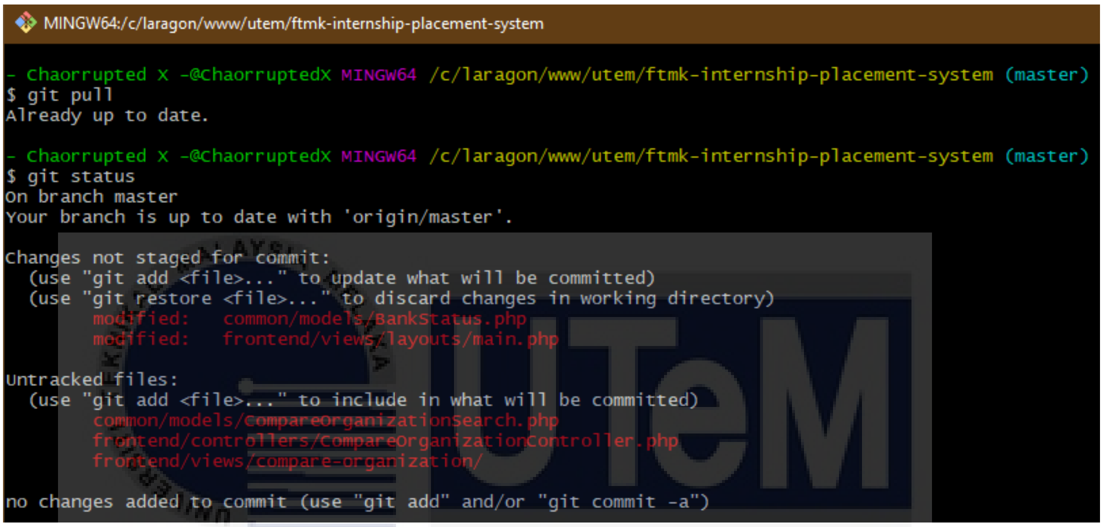
The coding version can be controlled by using the Git commands. Once the system has been cloned from the GitLab by using `git clone` command, and initialize the environment as development, the implementation phase will start. Figure 5.3 shows the environment setup command.



```
Microsoft Windows [Version 10.0.19043.1151]
(c) Microsoft Corporation. All rights reserved.

C:\laragon\www\utem\ftmk-internship-placement-system>init
Yii Application Initialization Tool v1.0

Which environment do you want the application to be initialized in?

  [0] Development
  [1] Production

  Your choice [0-1, or "q" to quit] _
```

Figure 5.3: Environment Setup Command

During the development, after some parts or modules have been completed, the code that have been change can be view by using the `git status` command. This will return the current working branch. Next, use `git add` command to add the files to the staging area, then `git commit` command to record the changes made to the files to a local repository. The message can be including in each commit explaining the changes made in commit. Finally, use `git push` to send local commits to the remote repository which is GitLab. Figure 5.4 shows several Git commands run in Git Bash.



MINGW64:/c/laragon/www/utem/ftmk-internship-placement-system

```
- Chaorrupted X -@ChaorruptedX MINGW64 /c/laragon/www/utem/ftmk-internship-placement-system (master)
$ git pull
Already up to date.

- Chaorrupted X -@ChaorruptedX MINGW64 /c/laragon/www/utem/ftmk-internship-placement-system (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   common/models/BankStatus.php
        modified:   frontend/views/layouts/main.php

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        common/models/CompareOrganizationSearch.php
        frontend/controllers/CompareOrganizationController.php
        frontend/views/compare-organization/

no changes added to commit (use "git add" and/or "git commit -a")
```

Figure 5.4: Git Bash

5.4 Implementation Status

Table 5.1 shows the progress of the development status for each of the system modules.

Table 5.1: Implementation Status

Module Name	Description	Duration to Complete	Date Completed
User Authentication	Authenticate user, and authorize user based on access control	1 Week	20 April, 2021
Management	Manage user, department, programme, state, language, and session	1 Month	16 May, 2021
Announcement	Post announcement	1 Week	21 April, 2021
Resume	Generate resume from the system	2 Weeks	8 June, 2021
Internship Application	Manage internship application, apply internship, compare organization, generate offer letter, and feedback form	2 Weeks	22 June, 2021
Organization Profile	Generate organization profile from the system	1 Week	8 June, 2021

Internship Recruitment	Advertise organization’s requirements and review student for industrial training	1 Week	9 June, 2021
Supervision	Manage and assign supervisor with student, view assigned students	2 Weeks	23 June, 2021
Report	Provide dynamic statistic report	1 Week	24 June, 2021

5.5 Conclusion

After the development environment has been set up with the installation of various types of software, the system is implemented or coded as designed in the implementation phase.

The version of source code is maintained by using GitLab where it is available in the remote repository. By using GitLab, all the version source code are kept in the history in GitLab.

The output from this phase, which is the complete system, will be used as input for the next phase, which is testing phase.

CHAPTER 6: TESTING

6.1 Introduction

The system testing is the last phase in the System Development Life Cycle (SDLC) of FTMK Internship Placement System (FIPS). The purpose of system testing is to minimize the error occur in the system and provide interface and flow that is appropriate for the users of FIPS. The result of software testing will be done by the system developer and the users of FIPS.

Preparation of the test plan, which outlines the organization accountable for each activity as well as the risks connected with the test plan, test environment, and test schedule, will be among the testing tasks. Aside from that, appropriate testing methodologies for system testing will be chosen to limit the hazards inherent in the computer system. The factors that have traditionally been emphasized in the selection of test techniques are correctness, dependability, and convenience of use. Each test case and intended result for each module will be designed and documented as part of the test design process. Finally, the system developer analyzed the scale of the system by analyzing the test results and analyzing the testing results.

6.2 Test Plan

The strategy of a test plan is divided into three phases: test organization, test environment, and test schedule. The testing phase of the SDLC will be covered in this planning. The user who is involved in the testing process is determined by the test organization. The test environment is the site or location where the testing will take place, and the test schedule is the arrangement for the time and circles of the testing.

6.2.1 Test Organization

The personnel in charge of the activity during the testing process are known as the test organization. Personnel descriptions exist for those involved in system application testing. The variety of people involved in the testing phase is a benefit for FIPS evaluation. The test organization's team includes a system developer, an FTMK student, AMTIS Solution staff, and FTMK staff.

System developer is the person in charge to test and evaluate the FIPS. The student from FTMK will test the system using the student role. Staff from AMTIS Solution will test the system as organization role. Finally, staff from FTMK will test the system using coordinator and supervisor role. Table 6.1 shows the user of test organization and the task for each user during the testing process.

Table 6.1: User and Task for the Testing Phase

Module Name	Description
System Developer	The user that involves in testing, analyst and document the result of testing. Ensure the FIPS can be run successfully
FTMK Staff	Test the system using coordinator and supervisor role
FTMK Student	Test the system using the student role
AMTIS Staff	Test the system using the organization role

6.2.2 Test Environment

The test environment is made up of the location and setting in which the testing will take place. In this system application testing, there are various different sorts of environments. The test environment is critical for the system developer to guarantee that the testing results are as expected based on real-world system application usage.

For the test environment, the FIPS will be host online, and meeting with the user will be done using platform such as Microsoft Teams, Google Meet, Webex, or Zoom. The system developer will guide the user during the testing process.

6.2.3 Test Schedule

The system developer can use the test schedule as a guide. The name of the module, the test activity, the duration, the test start date, and the test end date are all included. This is to guarantee that the testing process is carried out in a systematic way. Table 6.2 shows the test schedule for the FIPS.

Table 6.2: Test Schedule for FIPS

Module	Test Activity	Duration	Start Date	End Date
User Authentication	Unit Testing, User Acceptance Testing	1 day	27/8/2021	27/8/2021
Management	Unit Testing	2 days	26/8/2021	27/8/2021
Announcement	Unit Testing, User Acceptance Testing	1 day	27/8/2021	27/8/2021
Resume	Unit Testing, User Acceptance Testing	3 days	23/8/2021	25/8/2021
Internship Application	Unit Testing, User Acceptance Testing	5 days	23/8/2021	27/8/2021
Organization Profile	Unit Testing, User Acceptance Testing	3 days	23/8/2021	25/8/2021
Internship Recruitment	Unit Testing, User Acceptance Testing	5 days	23/8/2021	27/8/2021
Supervision	Unit Testing, User Acceptance Testing	2 days	25/8/2021	26/8/2021
Report	Unit Testing, User Acceptance Testing	1 day	25/8/2021	25/8/2021

6.3 Test Strategy

To pin-up the testing process, the system developer requires a plan. The top-down testing framework strategy and white-box and black-box test classes are used in the system application testing process. The top-down view reveals the first module to test, which is the system application's primary module, internship application, which is the software structure's height level module. The lower module, user authentication, was the last to be tested.

Software replacement simulators required for modules not available when performing unit or integration testing. Top-down testing will produce many stubs, which means the dummy module from the top module. The benefit of this technique is that system application testing will be conducted on a consistent time schedule and in a systematic manner.

6.3.1 Classes of Tests

The classes of test are divided into security testing, error handling test, and user acceptance testing.

i. Security Test

To identify, analyst, and secure every way and hole that exist at the system application.

ii. Error Handling Test

To ensure the application can deny any wrong inputs from the user. The error message will display should the user input the wrong detail such as inserting the wrong username and password.

iii. User Acceptance Testing

Test the acceptance of user whether the system is user friendly and appropriate according to the business process.

6.4 Test Design

Test designs consist of two phases which are test description and test data.

6.4.1 Test Description

For each module, a test description will be built in the form type that includes module identification, test cases, and expected results. The software tester performs the testing depending on the testing script that the system developer has created. Tables below shows the test description for each module in FIPS.

Table 6.3: Test Description for User Authentication Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0000	Enter the wrong username and password	The system will display "Incorrect username or password"	OK
FIPS_TD_0001	Enter the correct username and password	User are logged in into the system	OK

Table 6.4: Test Description for Management Module

Test ID	Action	Expected Output	Actual Result
User Management			
FIPS_TD_0100	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0101	Insert an existing username	The system will display username already taken	OK
FIPS_TD_0102	Insert an existing email	The system will display email already taken	OK
FIPS_TD_0103	Insert the data correctly	The system will save the data into the database	OK
Department Management			
FIPS_TD_0110	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0111	Insert an existing department code	The system will display department code already taken	OK
FIPS_TD_0112	Insert the data correctly	The system will save the data into the database	OK
Programme Management			
FIPS_TD_0120	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0121	Insert an existing programme code	The system will display programme code already taken	OK
FIPS_TD_0122	Insert the data correctly	The system will save the data into the database	OK
State Management			

FIPS_TD_0130	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0131	Insert an existing state name	The system will display state name already taken	OK
FIPS_TD_0132	Insert the data correctly	The system will save the data into the database	OK
Language Management			
FIPS_TD_0140	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0141	Insert an existing language name	The system will display language name already taken	OK
FIPS_TD_0142	Insert the data correctly	The system will save the data into the database	OK
Session Management			
FIPS_TD_0150	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0151	Insert an existing semester, start year, and end year	The system will display academic session already exist	OK
FIPS_TD_0152	Insert the data correctly	The system will save the data into the database	OK

Table 6.5: Test Description for Announcement Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0200	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0201	Insert the data correctly	The system will save the data into the database	OK

Table 6.6: Test Description for Resume Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0300	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0301	Insert the start year greater than end year	The system will display end year must be greater than start year	OK
FIPS_TD_0302	Insert both referees from the same person	The system will display both referees must be unique	OK
FIPS_TD_0303	Insert the data correctly	The system will save the data into the database	OK

Table 6.7: Test Description for Internship Application Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0400	Go to List of Organization, and apply for internship	The button will change to 'Internship Applied'	OK
FIPS_TD_0401	At the dashboard, accept the organization approval	The internship application status will be updated to 'Student Accepted'	OK
FIPS_TD_0402	At the dashboard, reject the organization approval	The internship application status will be updated to 'Rejected'	OK

Table 6.8: Test Description for Organization Profile Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0500	Click save button without inserting any data	The system will display the attributes cannot be blank	OK
FIPS_TD_0501	Insert the data correctly	The system will save the data into the database	OK

Table 6.9: Test Description for Internship Recruitment Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0600	Approve the student application	The internship status will be updated to ‘Waiting for Student Approval’	OK
FIPS_TD_0601	Reject the student application	The internship status will be updated to ‘Rejected’	OK

Table 6.10: Test Description for Supervision Module

Test ID	Action	Expected Output	Actual Result
Assign Supervision			
FIPS_TD_0700	Assign student under a supervisor	The student will be assigned under the supervisor	OK
Student List			
FIPS_TD_0710	Go to the student list	Shows the list of students under supervisor	OK

Table 6.11: Test Description for Report Module

Test ID	Action	Expected Output	Actual Result
FIPS_TD_0800	View the report	The report data is correct according to the data in the system	OK

6.4.2 Test Data

Tables below shows the test data for each test description documented in the previous section.

Table 6.11: Test Data for User Authentication Module

User Authentication Module		
Test No.	**Attribute**	**Data**
TEST_D_01	**User Login:**	
	Username	b031910250
	Password	Zaki0123

Table 6.12: Test Data for Management Module

Management Module		
Test No.	**Attribute**	**Data**
TEST_D_02	**User Management:**	
	Username	b031910195
	Email	b031910195@student.utem.edu.my
	Password	Asyraf0123
	Role	Student
TEST_D_03	**Department Management:**	
	Department Code	SE
	Department Name	Department of Software Engineering
TEST_D_04	**Programme Management:**	
	Department Code	SE
	Programme Code	BITS
	Programme Name	Bachelor of Computer Science (Software Development)
TEST_D_05	**State Management:**	
	State Name	Selangor
	State Type	Saturday and Sunday Holidays
TEST_D_06	**Language Management:**	
	Language Name	Malay
TEST_D_07	**Session Management:**	
	Semester	1
	Start Year	2021
	End Year	2022
	Start Internship Date	2021-07-19
	End Internship Date	2021-09-24
	Status	Active

Table 6.13: Test Data for Announcement Module

Announcement Module		
Test No.	**Attribute**	**Data**
TEST_D_08	**Post Announcement:**	
	Title	Update Resume
	Content	Pleaes update your resume before applying for an internship placement. Thank you.
	Role Name	Student

Table 6.13: Test Data for Resume Module

Resume Module		
Test No.	**Attribute**	**Data**
TEST_D_09	**Update Resume:**	
	Profile Picture	[Image]
	Telephone Number	0133430140
	Address	3911, Kelas E, Jalan Tebuan, Lapangan Terbang Antarabangsa Senai, 81250 Senai, Johor
	Skill Name	Laravel
	Skill Rate	85%
	Language Name	English
	Language Rate	75%
	Education Name	MiCoST
	Education Start Year	2014
	Education End Year	2017
	Work Experience Name	AMTIS Solution Sdn. Bhd
	Work Experience Start Year	2017
	Work Experience End Year	2019
	Work Experience Description	Work as a programmer
	Achievement Name	JAMCSIIX
	Achievement Year	2018
	Achievement Description	Achieved gold award
	First Referee	Dr. Wahidah
	Second Referee	Ts. Azlianor

Table 6.14: Test Data for Organization Profile Module

Organization Profile Module		
Test No.	**Attribute**	**Data**
TEST_D_10	**Update Organization Profile:**	
	Logo	[Image]
	Organization Name	AMTIS Solution Sdn. Bhd.
	Organization Type	Private
	Address	No.5, Jalan TU 40, Business Park@MITC, 75450 Ayer Keroh, Melaka
	State	Melaka
	Contact Number	06-232 6663
	Start Day	Monday
	End Day	Friday
	Open Hour	08:30
	Close Hour	18:00
	Job Scope Requirements	BITD, BITS, DIT
	Job Description	Solution in Information Technology & Knowledge
	Accepting Internship	Yes

6.5 Test Results and Analysis

The User Acceptance Testing result for FIPS can refer to Appendix E, F, G, and H. Tables below shows the overall test result and analysis for the FIPS.

Test No.	Module	Result	Scale (1 – 5)
TEST_R_00	**Login** • System Administrator • Coordinator • Supervisor • Student • Organization	OK	5
TEST_R_01	**User Management** • Register User • View User • Update User • Remove User • Search User	OK	5
TEST_R_02	**Department Management** • Register Department • View Department • Update Department • Remove Department • Search Department	OK	5
TEST_R_03	**Programme Management** • Register Department • View Department • Update Department • Remove Department • Search Department	OK	5
TEST_R_04	**State Management** • Add State • View State	OK	5

	- Update State - Remove State - Search State		
TEST_R_05	**Language Management** - Add Language - View Language - Update Language - Remove Language - Search Language	OK	5
TEST_R_06	**Session Management** - Register Session - View Session - Update Session - Remove Session - Search Session	OK	5
TEST_R_07	**Announcement** - Post Announcement - View Announcement - Update Announcement - Remove Announcement - Search Announcement	OK	5
TEST_R_08	**Resume** - Update Resume	OK	4
TEST_R_09	**Internship Application** - Search Organization - View Organization Profile - Apply for Internship - Accept Organization Approval	OK	4
TEST_R_10	**Organization Profile** - Update Organization Profile	OK	5

TEST_R_11	**Internship Recruitment** • Search Student Applicant • View Student Resume • Accept Student Applicant • Reject Student Applicant	OK	4
TEST_R_12	**Supervision** • Search Student • Assign Student under Supervisor • View List of Student under Supervisor	OK	5
TEST_R_13	**Report** • View Report	OK	5

6.6 Conclusion

In the test phase, the test organization are system developer, FTMK student, AMTIS staff, and FTMK staff. The test organization are responsible for the activity during the testing process. The test environment will be conducted online, by using meeting platform such as Microsoft Teams, Google Meet, Webex, or Zoom. The system developer will guide the user during the testing phase.

Several tests have been done, such as security testing, error handling, and user acceptance testing. This is to ensure the stability and correctness of the system according to the business process.

In the next chapter, conclusion, will conclude the overall documentation of FIPS.

CHAPTER 7: CONCLUSION

7.1 Observation on Weaknesses and Strengths

The FTMK Internship Placement System (FIPS) is a web application that is developed to provide online internship placement procedure. From the observation made towards the developing of the FIPS, the strengths and weaknesses of the system is identified and will be discussed in this chapter.

There are several strengths identified in this system. By using this system, the student can get an internship placement in a short time. Student resume can be generated from the system, thus helps the student to setup their resume in a short time. Besides that, the registered organizations will provide an internship recruitment for the students. This system can also be used as a platform for FTMK coordinator and supervisor to keep track and monitor the student internship application status.

Some weaknesses are also has been identified in this system. Firstly, the system does not have a notification. Secondly, there are some bugs found by the user during the User Acceptance Testing session.

7.2 Propositions for Improvement

Based on the weakness identified, several improvements of this system can be done. First, by adding notification, it can be used to inform the student who has apply the internship. With the notification, the student can be alert that the application has been reviewed, accepted, or rejected by the organization. Other than the improvement from the identified weaknesses, there are other things that can be consider for improvement of this system such as the implementation of logbook. By adding logbook in the system, student can write their logbook in the system, and approval of organization and supervisor can be done through the system. Furthermore, industrial training marks module can be implemented to provide the supervisor and organization to key in the student internship marks based on their working performance and report. Finally, fixing bugs found by the user during the User Acceptance Testing session.

7.3 Project Contribution

The FIPS helps the FTMK staff, student, and organization to manage for industrial training process. The student can easily generate resume from the system, and then choose any organization they like to apply for an internship. The organization can view the student resume and make decision whether to accept or reject the student internship applicant. For FTMK coordinator and supervisor, able to view statistic chart data about the overall FTMK student internship placement status.

7.4 Conclusion

The FTMK Internship Placement System had achieved all the objective. The first objective is to develop a web-based system as a platform for students to search and apply industrial training by referring to the requirements given by the organization and to allow internship committee to monitor and track student's internship application status and industrial training progress.

The second objective is to design a solution that facilitate students, internship committee and organizations related to industrial training. The system able to help the staff and student for the internship placement process.

The third objective is to test the system for efficiency and user satisfaction in terms of usability by the students, internship committee and organizations. The user acceptance testing has been done to ensure the system meets the user needs.

All objectives of this system have been already achieved, hopefully the FIPS can contribute to the staff and student of FTMK and organization to facilitate the process for the internship placement.

REFERENCES

A. B. Kmail, M. M. (2015). An Automatic Online Recruitment System Based on Exploiting Multiple Semantic Resources and Concept-Relatedness Measures. 620-627. doi:10.1109/ICTAI.2015.95

Avinash S. Kapse, V. S. (2012). E- Recruitment. 82-86. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.686.5816&rep=rep1&type=pdf

Evanthia Faliagka, A. T. (2012). An integrated e-recruitment system for automated personality mining and applicant ranking. 551-568. doi:10.1108/10662241211271545

Evanthia Faliagka, K. R. (2012). Application of Machine Learning Algorithms to an online Recruitment System. 215-220. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.885.909&rep=rep1&type=pdf

Khalil Ghazzawi, A. A. (2014). Critical Success Factors of the E-Recruitment System. *Journal of Human Resources Management and Labor Studies*, 159-170. Retrieved from http://jhrmls.com/journals/jhrmls/Vol_2_No_2_June_2014/10.pdf

Loganesh Sivabalan, R. Y. (2014). How to Transform the Traditional Way of Recruitment into Online System. 178-185. Retrieved from https://www.researchgate.net/profile/Assc-Prof-Dr-Rashad-Yazdanifard/publication/268747592_How_to_Transform_the_Traditional_Way_of_Recruitment_into_Online_System/links/54756fee0cf29afed6126e36/How-to-Transform-the-Traditional-Way-of-Recruitment-into-Onlin

Loong, K. C. (2011). *Internship Placement System and Supervision System.* Universiti Teknikal Malaysia Melaka (UTeM). Retrieved from http://digitalcollection.utem.edu.my/7361/

M. Diaby, E. V. (2013). Toward the next generation of recruitment tools: An online social network-based job recommender system. 821-828. doi:10.1145/2492517.2500266

Mary Grace G. Ventura, R. P. (2013). Effectiveness of Online Job Recruitment System: Evidence. 152-159. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1073.7171&rep=rep1&type=pdf

T. Milczarek, B. S. (2008). Job recruitment system based on Java Enterprise Edition 5 platform. 457-460.

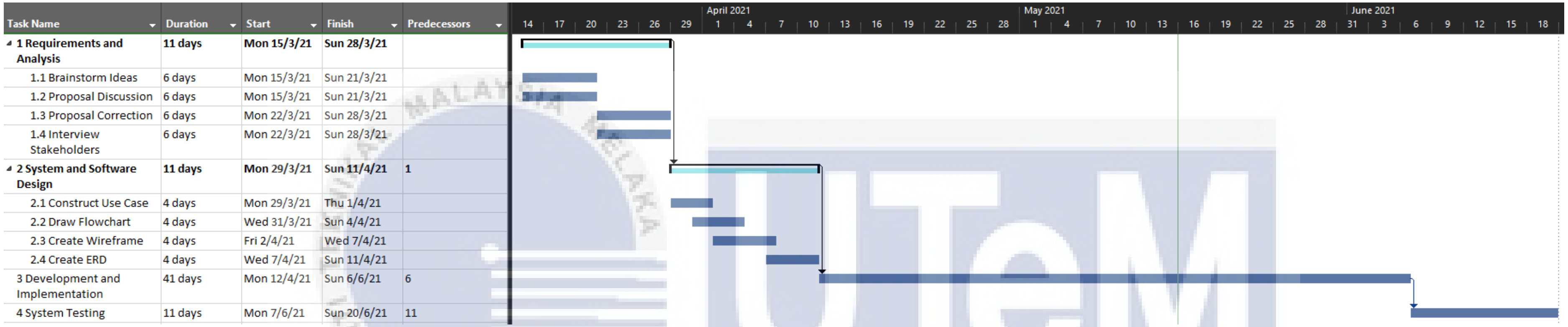
Universiti Teknikal Malaysia Melaka (UTeM). (2019). *Online Log Book System Guide*. Retrieved from UTeM FTKEE: https://ftkee.utem.edu.my/phocadownloadpap/PEI/Taklimat_Sistem_LI_Buku_Log_Pelajar_Aug2019_opt.pdf

Universiti Teknologi Malaysia (UTM). (2020). Guidelines for Industrial Training. Retrieved from https://engineering.utm.my/computing/industrialtraining/wp-content/uploads/sites/143/sites/143/2020/03/Industrial-Training-Guideline.pdf

Universiti Teknologi MARA (UiTM). (2020). Guidelines for Internship (Special Covid-19). Retrieved from https://fbm.uitm.edu.my/images/download/industrial_training/covid19/GUIDELINES_FOR_INTERNSHIP_2020_Covid19.pdf

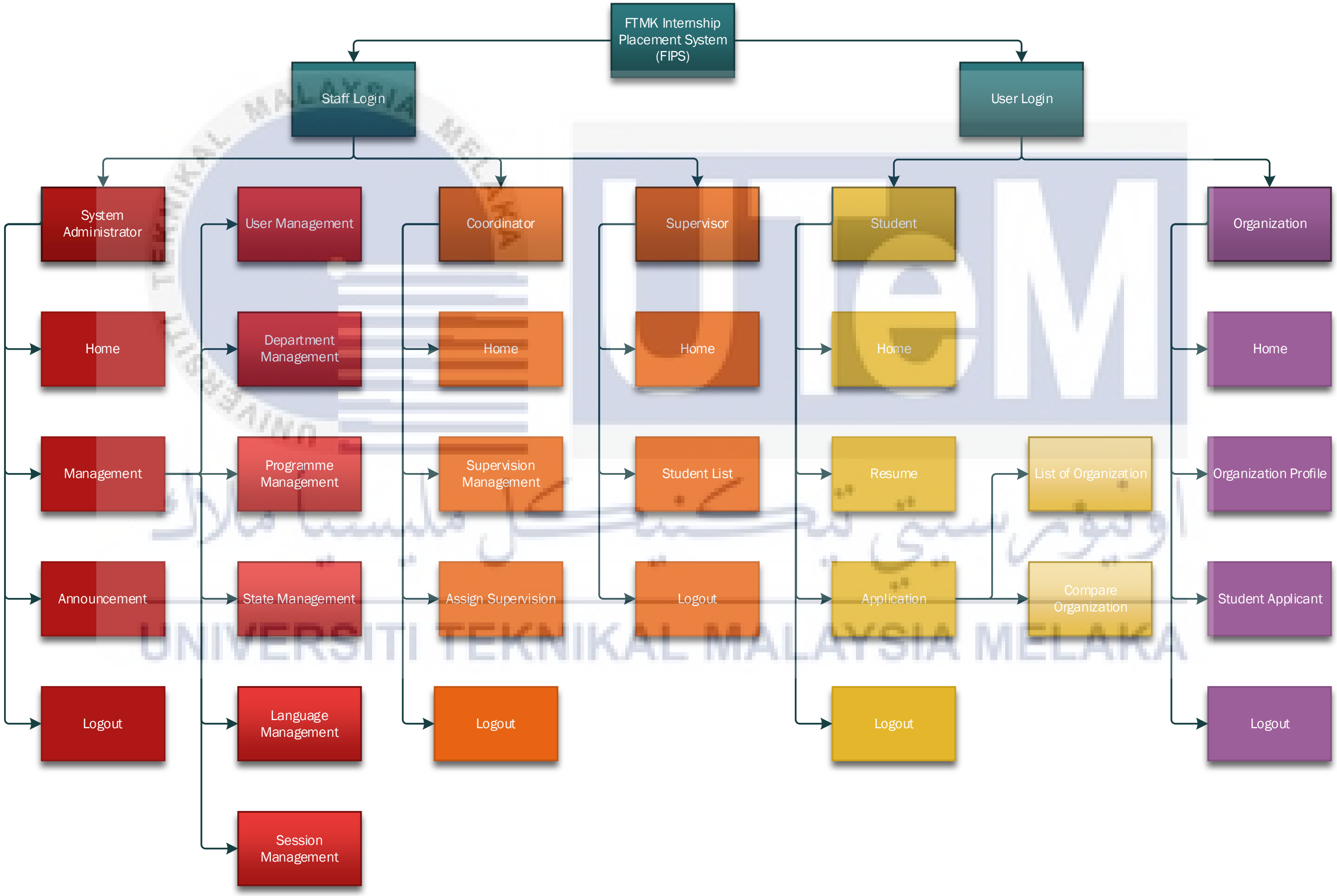
APPENDICES

APPENDIX A: GANTT CHART

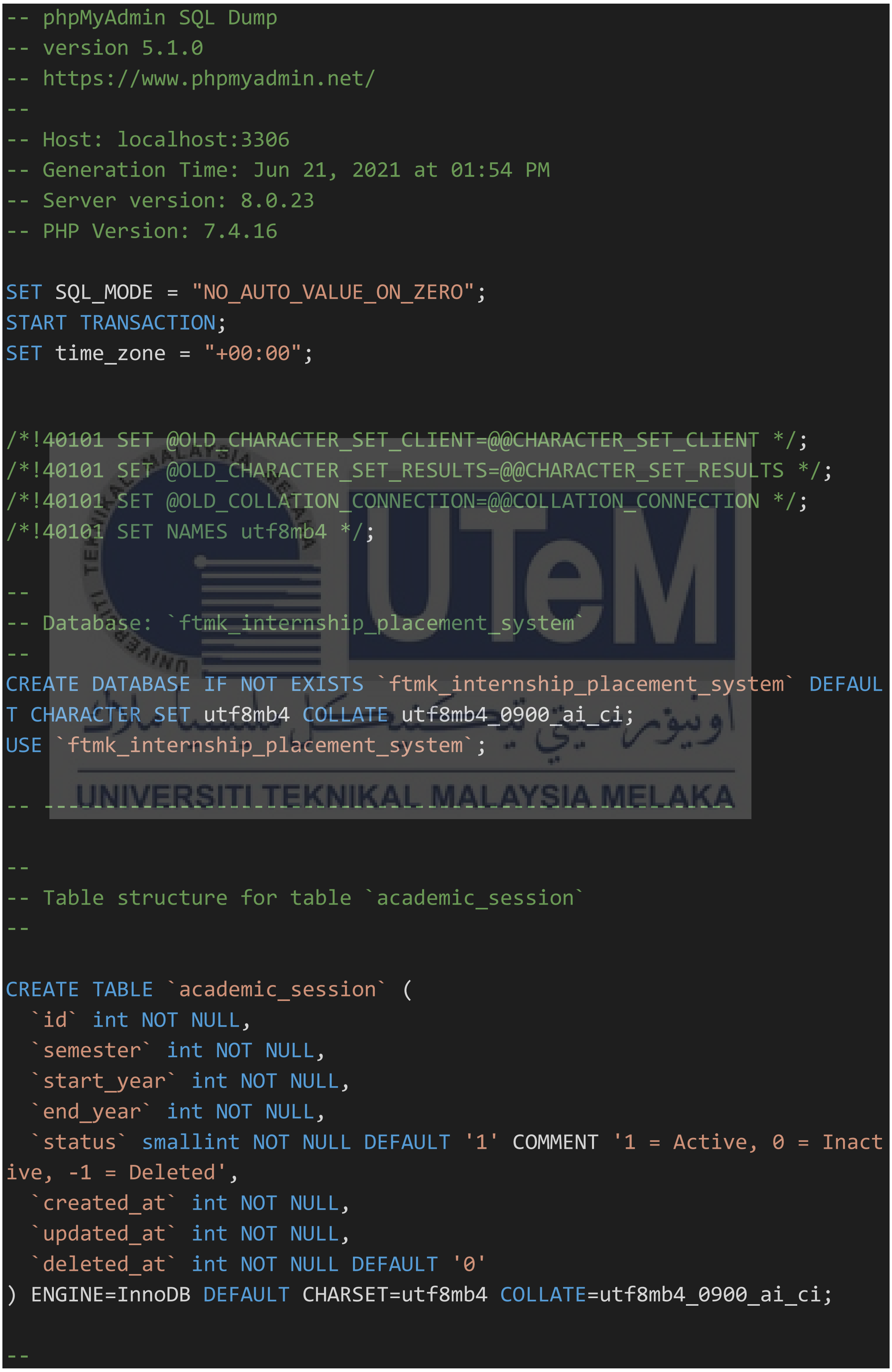


Task Name	Duration	Start	Finish	Predecessors
1 Requirements and Analysis	**11 days**	**Mon 15/3/21**	**Sun 28/3/21**	
1.1 Brainstorm Ideas	6 days	Mon 15/3/21	Sun 21/3/21	
1.2 Proposal Discussion	6 days	Mon 15/3/21	Sun 21/3/21	
1.3 Proposal Correction	6 days	Mon 22/3/21	Sun 28/3/21	
1.4 Interview Stakeholders	6 days	Mon 22/3/21	Sun 28/3/21	
2 System and Software Design	**11 days**	**Mon 29/3/21**	**Sun 11/4/21**	**1**
2.1 Construct Use Case	4 days	Mon 29/3/21	Thu 1/4/21	
2.2 Draw Flowchart	4 days	Wed 31/3/21	Sun 4/4/21	
2.3 Create Wireframe	4 days	Fri 2/4/21	Wed 7/4/21	
2.4 Create ERD	4 days	Wed 7/4/21	Sun 11/4/21	
3 Development and Implementation	41 days	Mon 12/4/21	Sun 6/6/21	6
4 System Testing	11 days	Mon 7/6/21	Sun 20/6/21	11

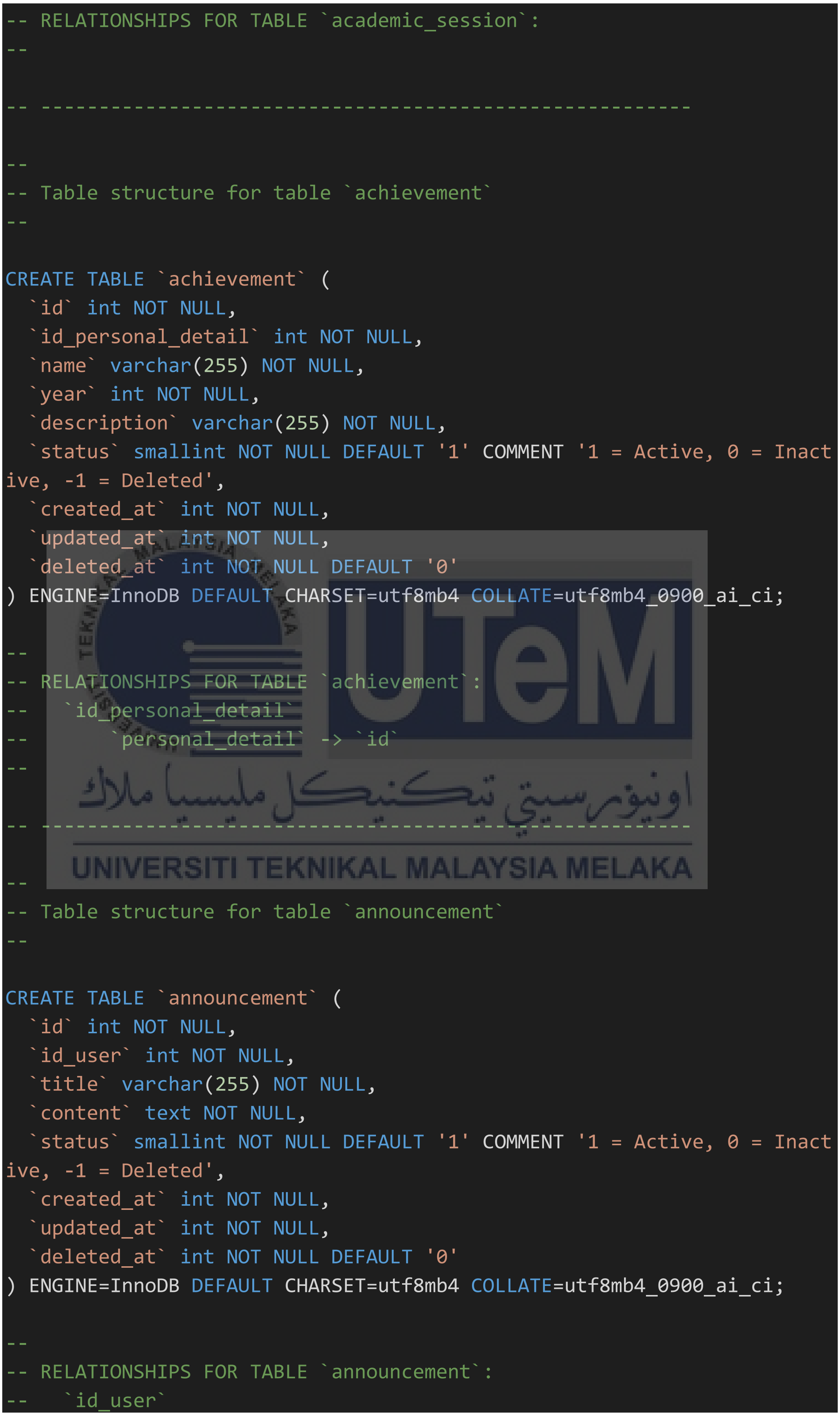
APPENDIX B: NAVIGATION FLOW



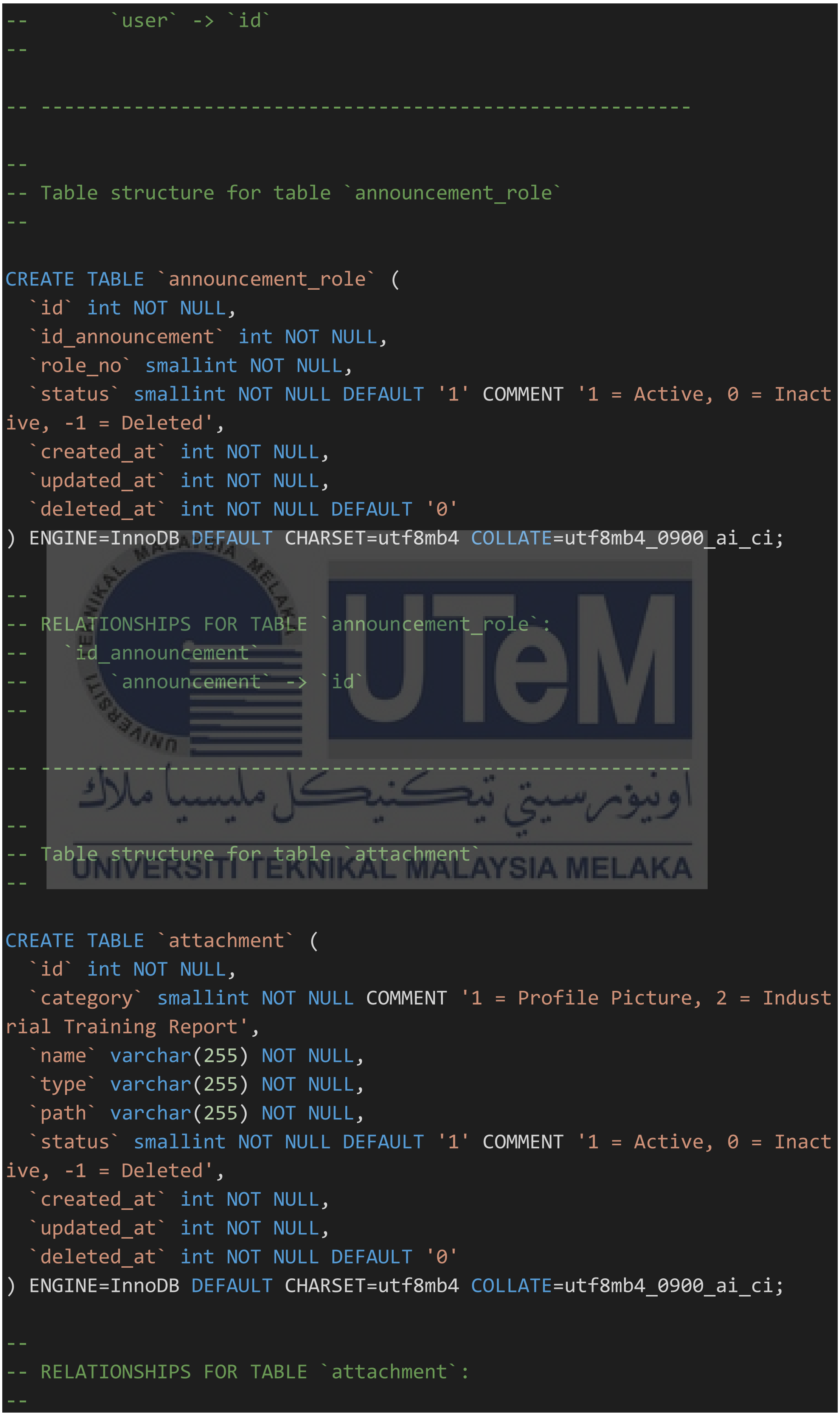
APPENDIX C: DATA DEFINITION LANGUAGE



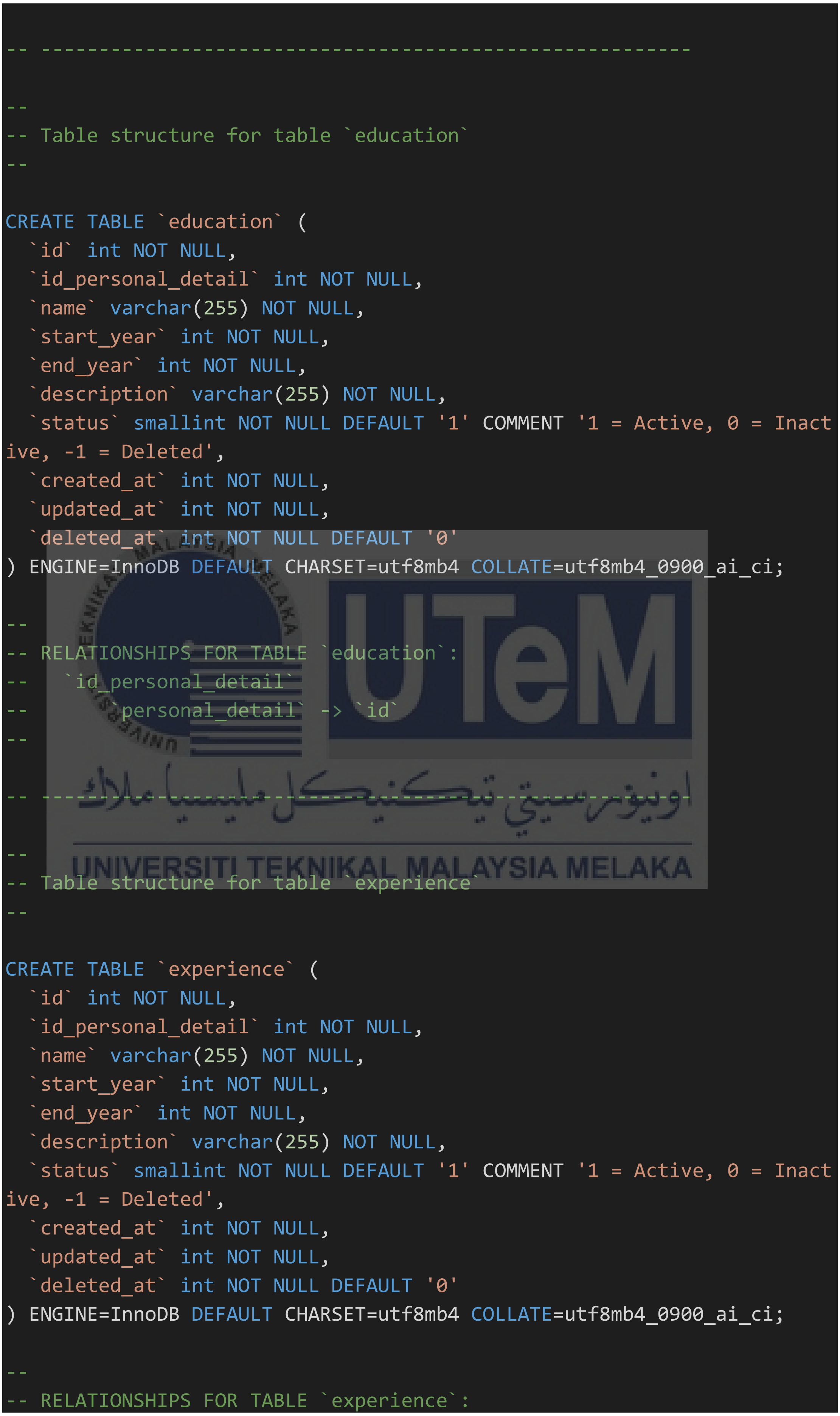
```
-- phpMyAdmin SQL Dump
-- version 5.1.0
-- https://www.phpmyadmin.net/
--
-- Host: localhost:3306
-- Generation Time: Jun 21, 2021 at 01:54 PM
-- Server version: 8.0.23
-- PHP Version: 7.4.16

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `ftmk_internship_placement_system`
--
CREATE DATABASE IF NOT EXISTS `ftmk_internship_placement_system` DEFAUL
T CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
USE `ftmk_internship_placement_system`;

-- --------------------------------------------------------

--
-- Table structure for table `academic_session`
--

CREATE TABLE `academic_session` (
  `id` int NOT NULL,
  `semester` int NOT NULL,
  `start_year` int NOT NULL,
  `end_year` int NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
```



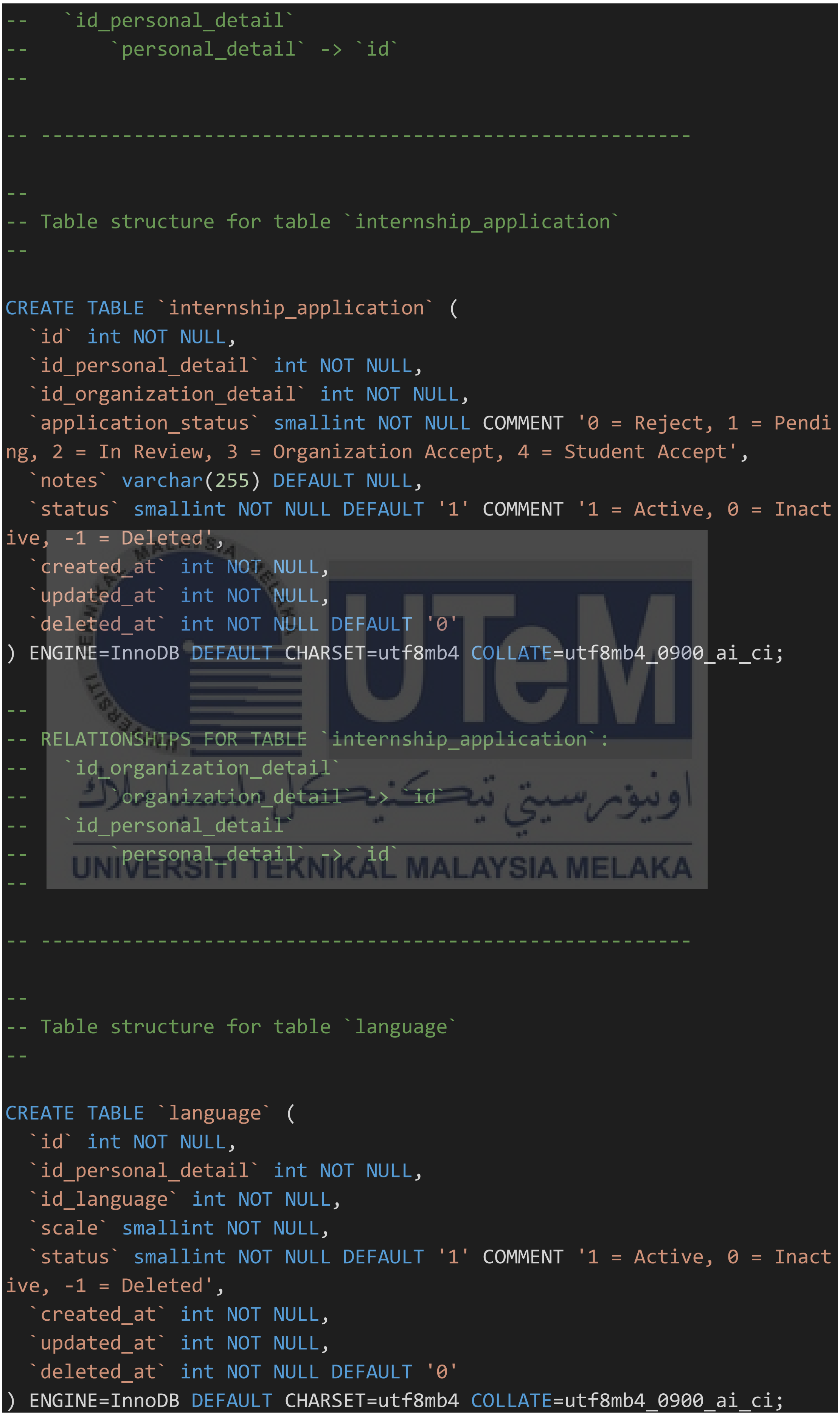
```
-- RELATIONSHIPS FOR TABLE `academic_session`:
--

-- --------------------------------------------------------

--
-- Table structure for table `achievement`
--

CREATE TABLE `achievement` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `year` int NOT NULL,
  `description` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `achievement`:
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `announcement`
--

CREATE TABLE `announcement` (
  `id` int NOT NULL,
  `id_user` int NOT NULL,
  `title` varchar(255) NOT NULL,
  `content` text NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `announcement`:
--   `id_user`
```



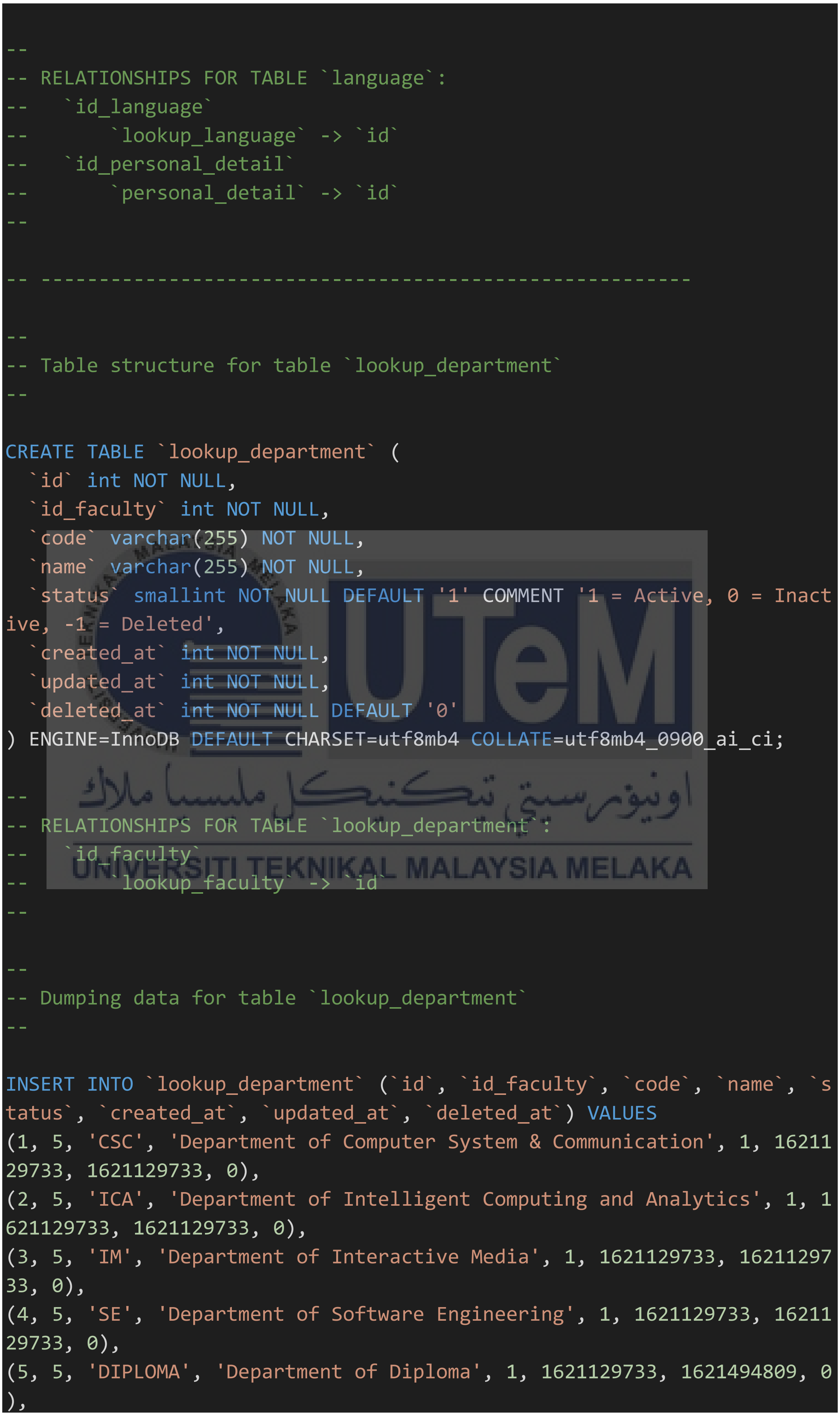
```
--       `user` -> `id`
--


-- --------------------------------------------------------


--
-- Table structure for table `announcement_role`
--

CREATE TABLE `announcement_role` (
  `id` int NOT NULL,
  `id_announcement` int NOT NULL,
  `role_no` smallint NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `announcement_role`:
--   `id_announcement`
--       `announcement` -> `id`
--


-- --------------------------------------------------------


--
-- Table structure for table `attachment`
--

CREATE TABLE `attachment` (
  `id` int NOT NULL,
  `category` smallint NOT NULL COMMENT '1 = Profile Picture, 2 = Indust
rial Training Report',
  `name` varchar(255) NOT NULL,
  `type` varchar(255) NOT NULL,
  `path` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `attachment`:
--
```



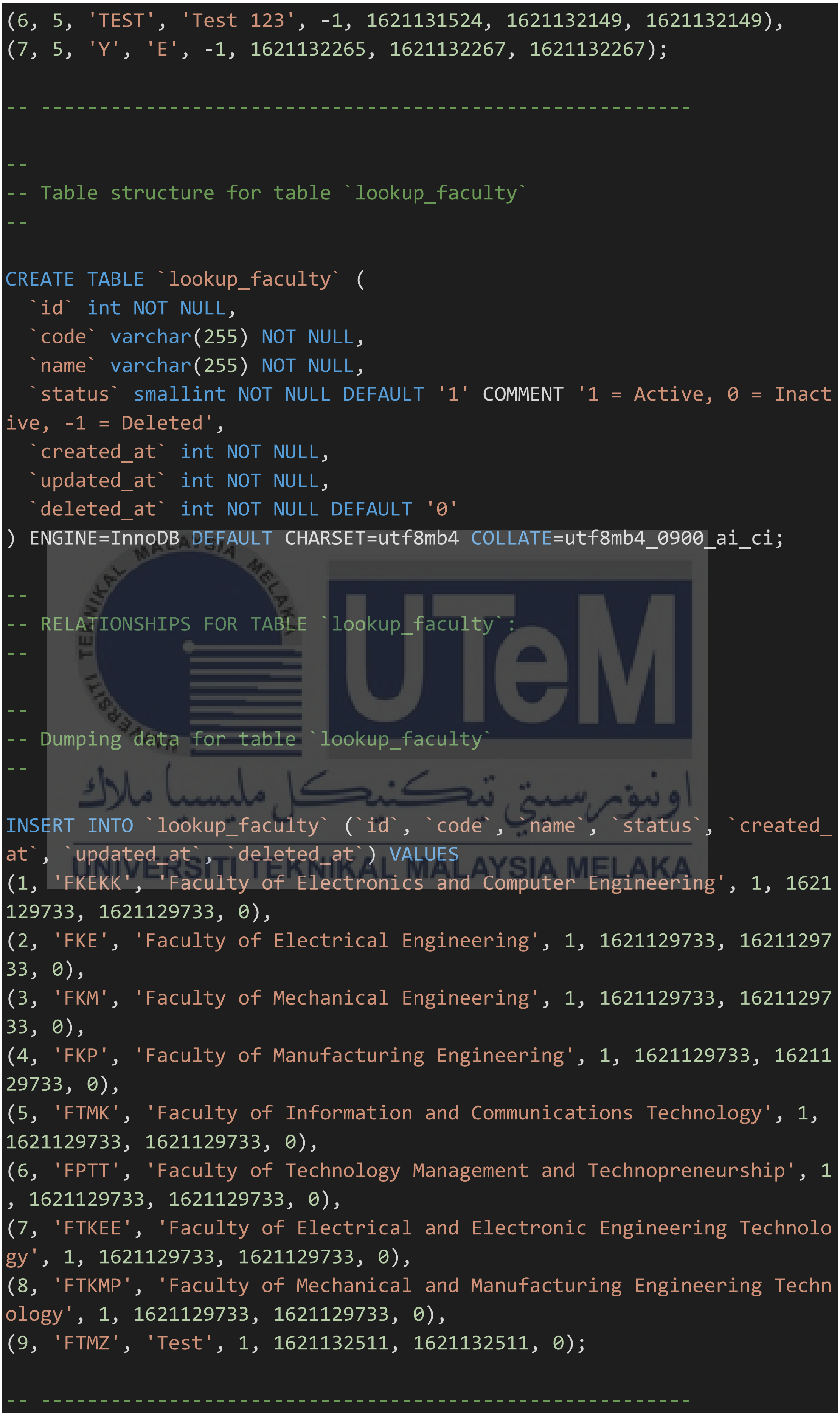
```
-- --------------------------------------------------------


--
-- Table structure for table `education`
--

CREATE TABLE `education` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `start_year` int NOT NULL,
  `end_year` int NOT NULL,
  `description` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `education`:
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------


--
-- Table structure for table `experience`
--

CREATE TABLE `experience` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `start_year` int NOT NULL,
  `end_year` int NOT NULL,
  `description` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `experience`:
```



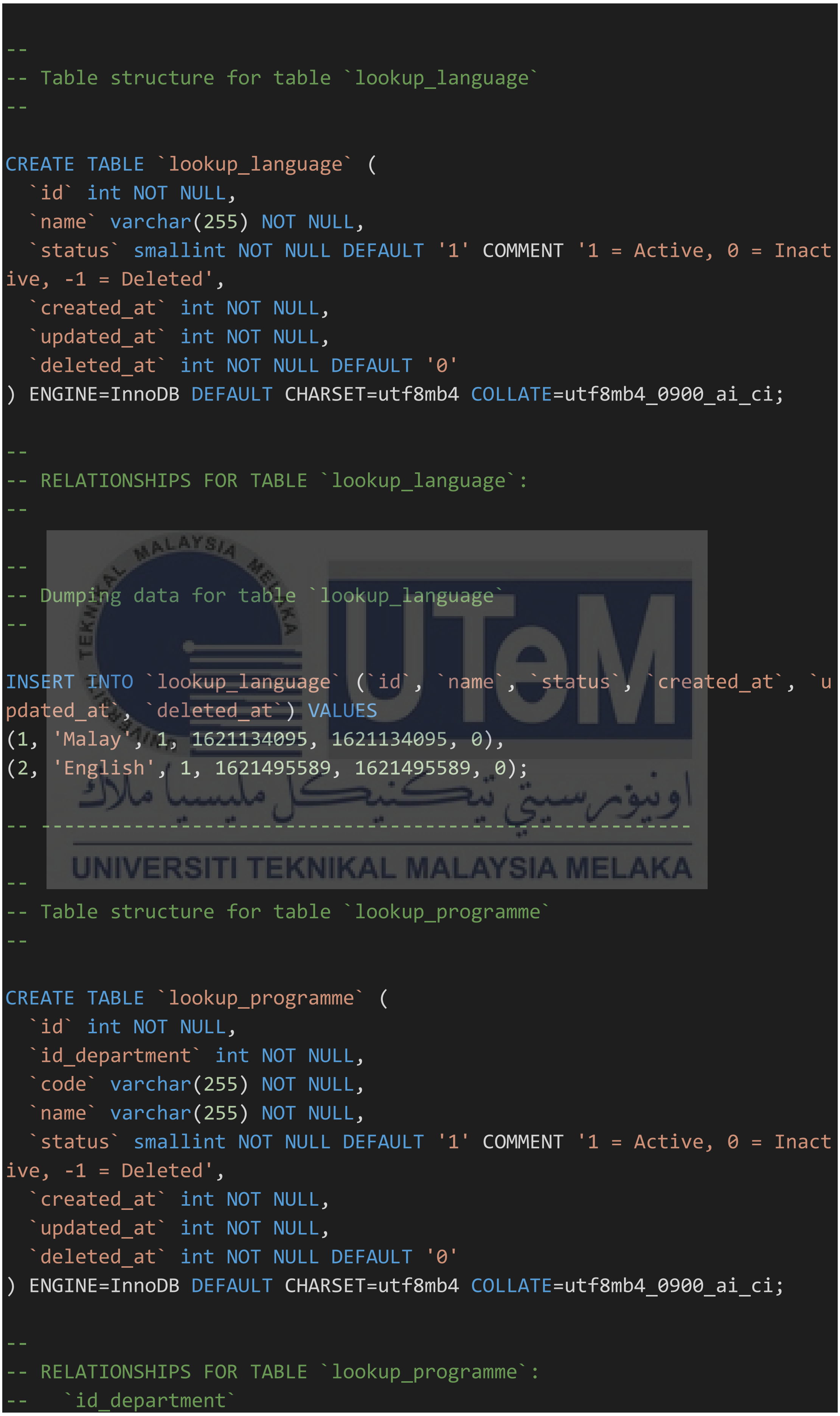
```
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `internship_application`
--

CREATE TABLE `internship_application` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `id_organization_detail` int NOT NULL,
  `application_status` smallint NOT NULL COMMENT '0 = Reject, 1 = Pendi
ng, 2 = In Review, 3 = Organization Accept, 4 = Student Accept',
  `notes` varchar(255) DEFAULT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `internship_application`:
--   `id_organization_detail`
--       `organization_detail` -> `id`
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `language`
--

CREATE TABLE `language` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `id_language` int NOT NULL,
  `scale` smallint NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```



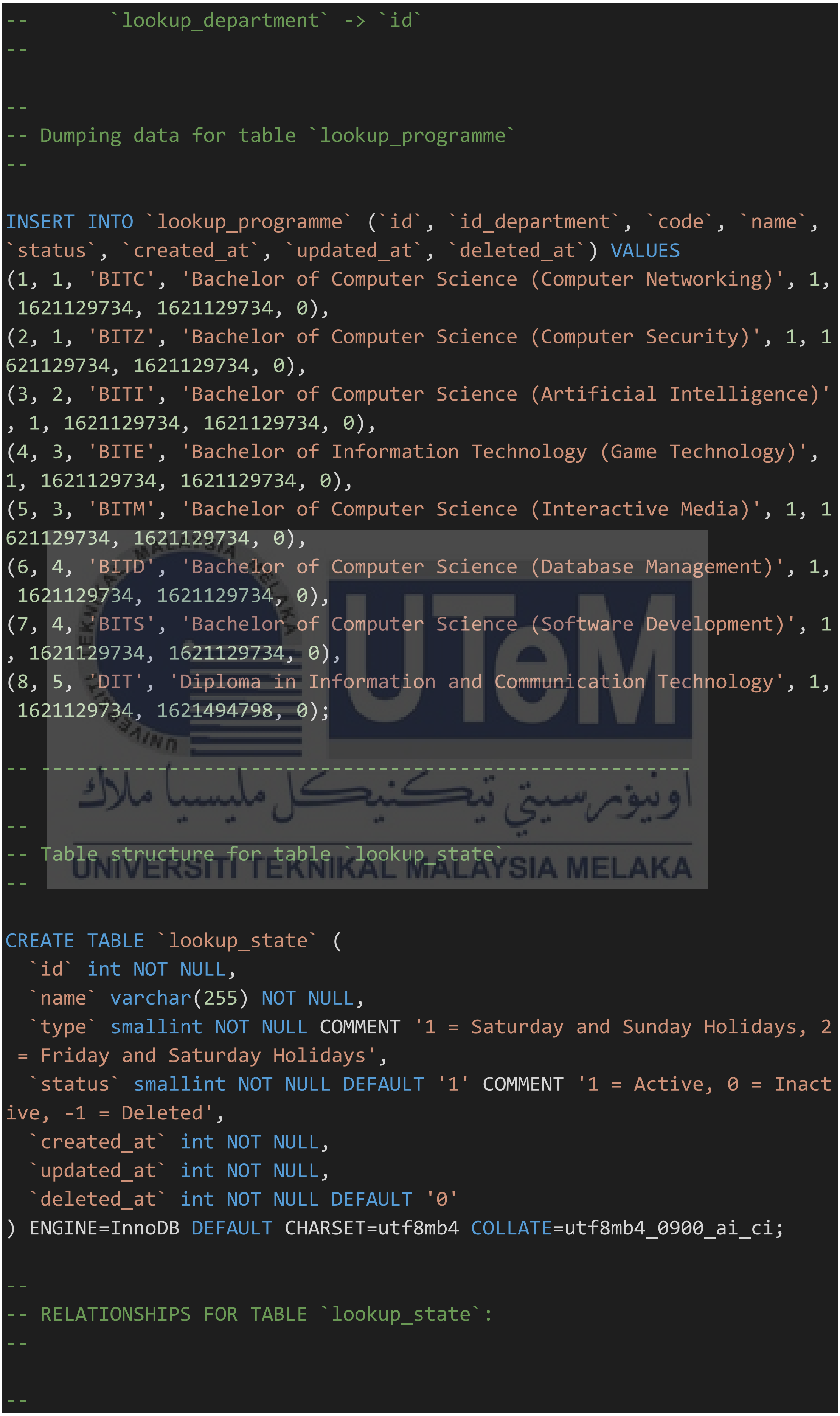
```
--
-- RELATIONSHIPS FOR TABLE `language`:
--   `id_language`
--       `lookup_language` -> `id`
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `lookup_department`
--

CREATE TABLE `lookup_department` (
  `id` int NOT NULL,
  `id_faculty` int NOT NULL,
  `code` varchar(255) NOT NULL,
  `name` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `lookup_department`:
--   `id_faculty`
--       `lookup_faculty` -> `id`
--

--
-- Dumping data for table `lookup_department`
--

INSERT INTO `lookup_department` (`id`, `id_faculty`, `code`, `name`, `s
tatus`, `created_at`, `updated_at`, `deleted_at`) VALUES
(1, 5, 'CSC', 'Department of Computer System & Communication', 1, 16211
29733, 1621129733, 0),
(2, 5, 'ICA', 'Department of Intelligent Computing and Analytics', 1, 1
621129733, 1621129733, 0),
(3, 5, 'IM', 'Department of Interactive Media', 1, 1621129733, 16211297
33, 0),
(4, 5, 'SE', 'Department of Software Engineering', 1, 1621129733, 16211
29733, 0),
(5, 5, 'DIPLOMA', 'Department of Diploma', 1, 1621129733, 1621494809, 0
),
```



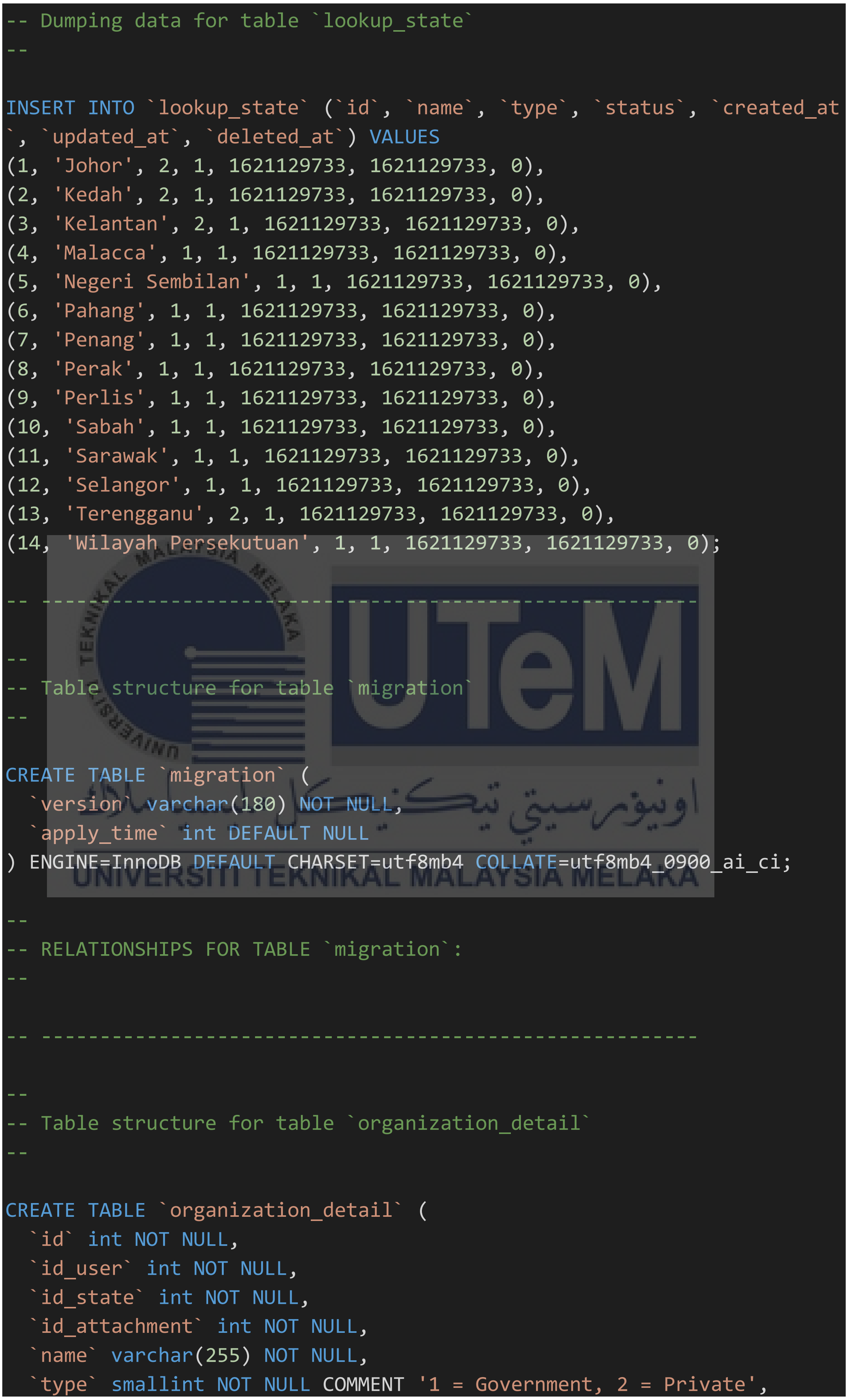
```
(6, 5, 'TEST', 'Test 123', -1, 1621131524, 1621132149, 1621132149),
(7, 5, 'Y', 'E', -1, 1621132265, 1621132267, 1621132267);

-- --------------------------------------------------------

--
-- Table structure for table `lookup_faculty`
--

CREATE TABLE `lookup_faculty` (
  `id` int NOT NULL,
  `code` varchar(255) NOT NULL,
  `name` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `lookup_faculty`:
--

--
-- Dumping data for table `lookup_faculty`
--

INSERT INTO `lookup_faculty` (`id`, `code`, `name`, `status`, `created_
at`, `updated_at`, `deleted_at`) VALUES
(1, 'FKEKK', 'Faculty of Electronics and Computer Engineering', 1, 1621
129733, 1621129733, 0),
(2, 'FKE', 'Faculty of Electrical Engineering', 1, 1621129733, 16211297
33, 0),
(3, 'FKM', 'Faculty of Mechanical Engineering', 1, 1621129733, 16211297
33, 0),
(4, 'FKP', 'Faculty of Manufacturing Engineering', 1, 1621129733, 16211
29733, 0),
(5, 'FTMK', 'Faculty of Information and Communications Technology', 1,
1621129733, 1621129733, 0),
(6, 'FPTT', 'Faculty of Technology Management and Technopreneurship', 1
, 1621129733, 1621129733, 0),
(7, 'FTKEE', 'Faculty of Electrical and Electronic Engineering Technolo
gy', 1, 1621129733, 1621129733, 0),
(8, 'FTKMP', 'Faculty of Mechanical and Manufacturing Engineering Techn
ology', 1, 1621129733, 1621129733, 0),
(9, 'FTMZ', 'Test', 1, 1621132511, 1621132511, 0);

-- --------------------------------------------------------
```



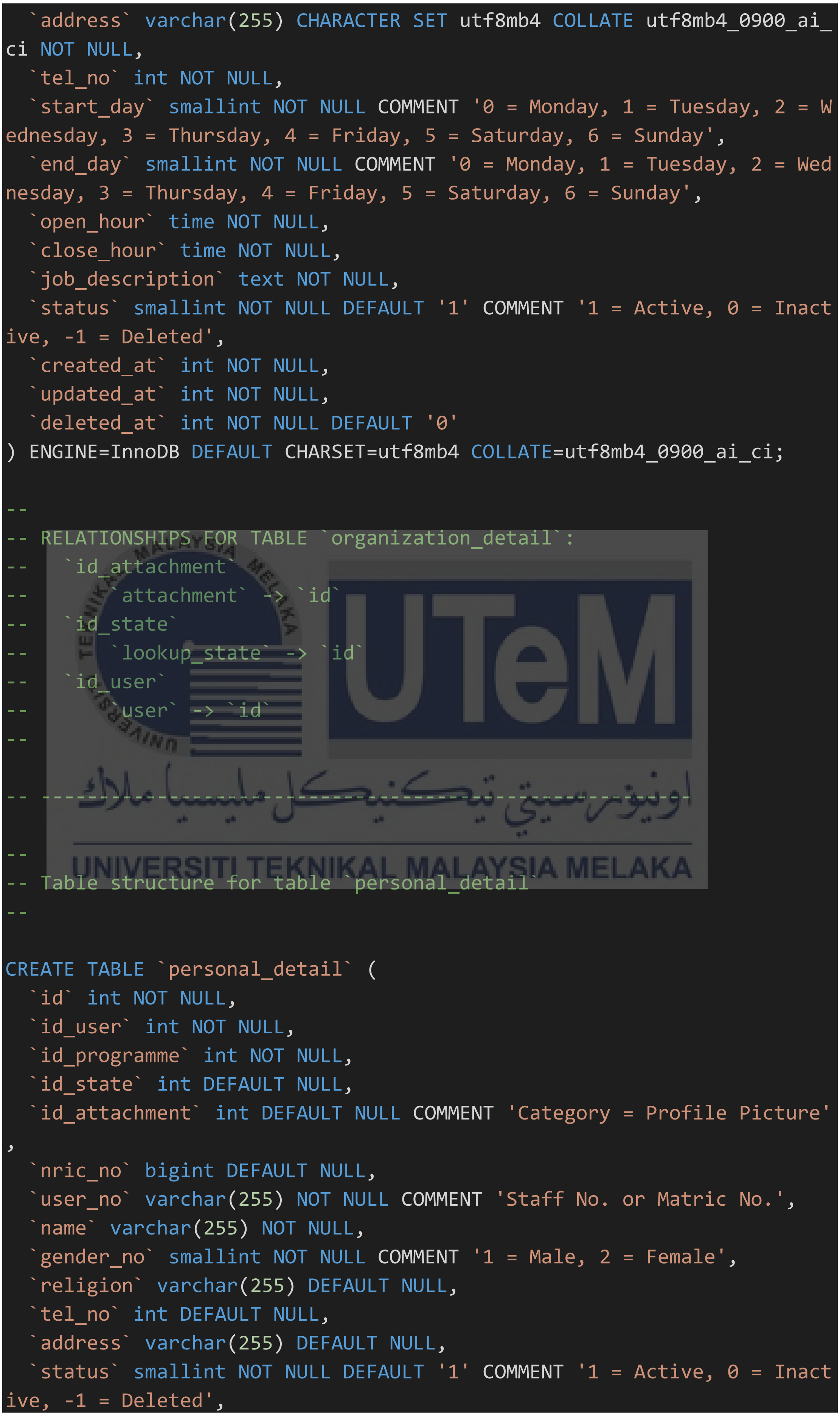
```
--
-- Table structure for table `lookup_language`
--

CREATE TABLE `lookup_language` (
  `id` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `lookup_language`:
--

--
-- Dumping data for table `lookup_language`
--

INSERT INTO `lookup_language` (`id`, `name`, `status`, `created_at`, `u
pdated_at`, `deleted_at`) VALUES
(1, 'Malay', 1, 1621134095, 1621134095, 0),
(2, 'English', 1, 1621495589, 1621495589, 0);

-- --------------------------------------------------------

--
-- Table structure for table `lookup_programme`
--

CREATE TABLE `lookup_programme` (
  `id` int NOT NULL,
  `id_department` int NOT NULL,
  `code` varchar(255) NOT NULL,
  `name` varchar(255) NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `lookup_programme`:
--   `id_department`
```



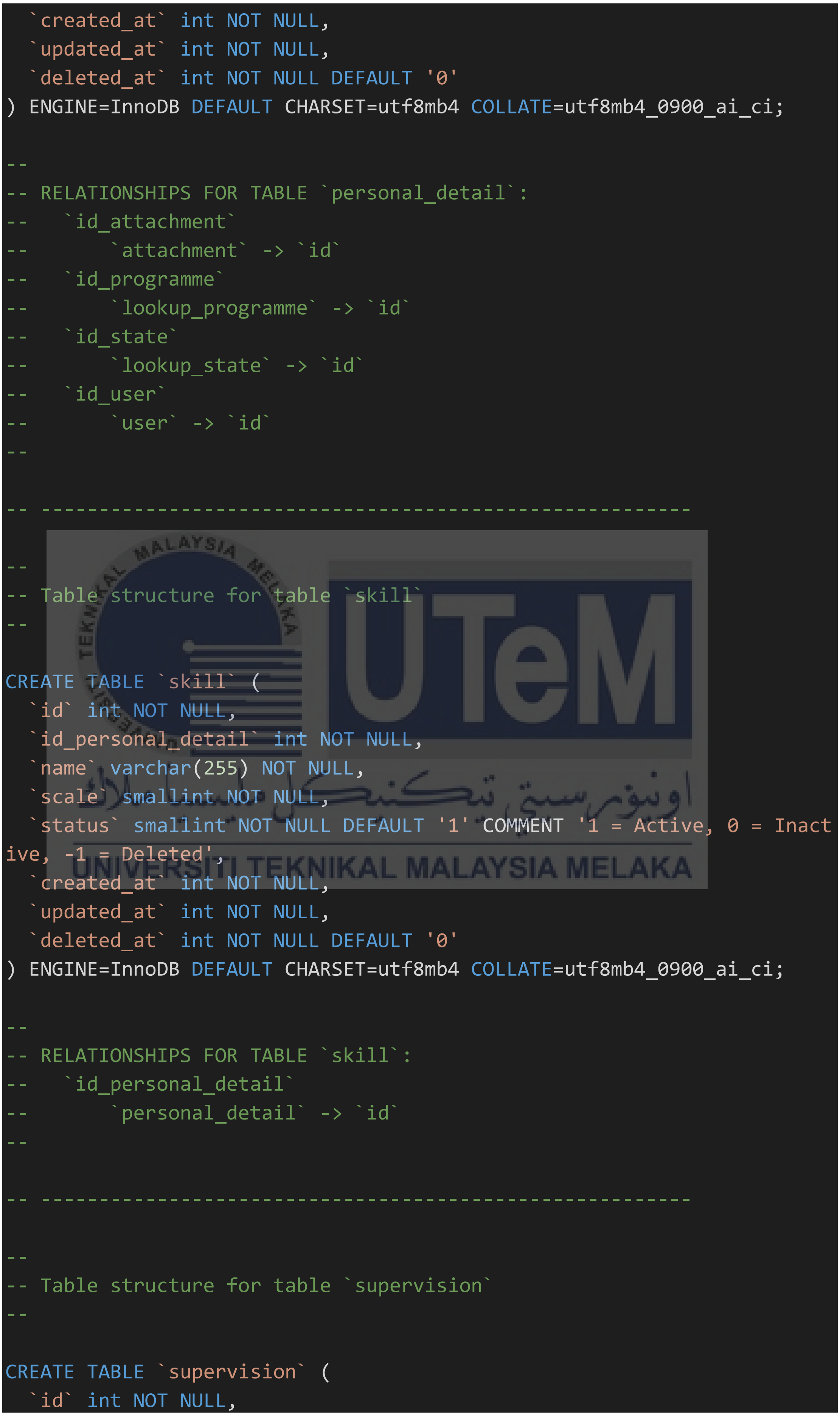
```
--       `lookup_department` -> `id`
--


--
-- Dumping data for table `lookup_programme`
--

INSERT INTO `lookup_programme` (`id`, `id_department`, `code`, `name`,
`status`, `created_at`, `updated_at`, `deleted_at`) VALUES
(1, 1, 'BITC', 'Bachelor of Computer Science (Computer Networking)', 1,
 1621129734, 1621129734, 0),
(2, 1, 'BITZ', 'Bachelor of Computer Science (Computer Security)', 1, 1
621129734, 1621129734, 0),
(3, 2, 'BITI', 'Bachelor of Computer Science (Artificial Intelligence)'
, 1, 1621129734, 1621129734, 0),
(4, 3, 'BITE', 'Bachelor of Information Technology (Game Technology)',
1, 1621129734, 1621129734, 0),
(5, 3, 'BITM', 'Bachelor of Computer Science (Interactive Media)', 1, 1
621129734, 1621129734, 0),
(6, 4, 'BITD', 'Bachelor of Computer Science (Database Management)', 1,
 1621129734, 1621129734, 0),
(7, 4, 'BITS', 'Bachelor of Computer Science (Software Development)', 1
, 1621129734, 1621129734, 0),
(8, 5, 'DIT', 'Diploma in Information and Communication Technology', 1,
 1621129734, 1621494798, 0);

-- --------------------------------------------------------

--
-- Table structure for table `lookup_state`
--

CREATE TABLE `lookup_state` (
  `id` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `type` smallint NOT NULL COMMENT '1 = Saturday and Sunday Holidays, 2
 = Friday and Saturday Holidays',
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `lookup_state`:
--


--
```



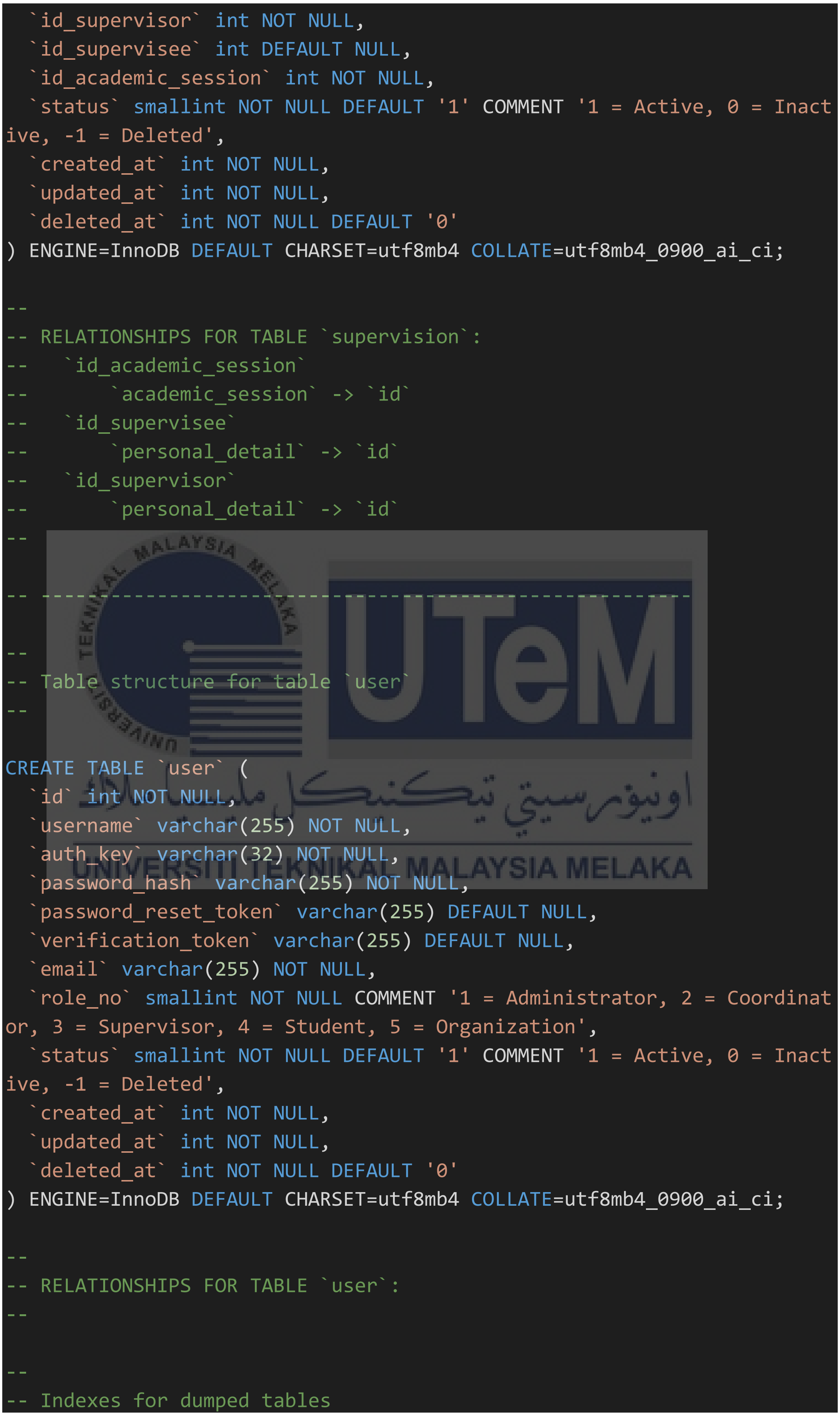
```
-- Dumping data for table `lookup_state`
--

INSERT INTO `lookup_state` (`id`, `name`, `type`, `status`, `created_at
`, `updated_at`, `deleted_at`) VALUES
(1, 'Johor', 2, 1, 1621129733, 1621129733, 0),
(2, 'Kedah', 2, 1, 1621129733, 1621129733, 0),
(3, 'Kelantan', 2, 1, 1621129733, 1621129733, 0),
(4, 'Malacca', 1, 1, 1621129733, 1621129733, 0),
(5, 'Negeri Sembilan', 1, 1, 1621129733, 1621129733, 0),
(6, 'Pahang', 1, 1, 1621129733, 1621129733, 0),
(7, 'Penang', 1, 1, 1621129733, 1621129733, 0),
(8, 'Perak', 1, 1, 1621129733, 1621129733, 0),
(9, 'Perlis', 1, 1, 1621129733, 1621129733, 0),
(10, 'Sabah', 1, 1, 1621129733, 1621129733, 0),
(11, 'Sarawak', 1, 1, 1621129733, 1621129733, 0),
(12, 'Selangor', 1, 1, 1621129733, 1621129733, 0),
(13, 'Terengganu', 2, 1, 1621129733, 1621129733, 0),
(14, 'Wilayah Persekutuan', 1, 1, 1621129733, 1621129733, 0);

-- --------------------------------------------------------

--
-- Table structure for table `migration`
--

CREATE TABLE `migration` (
  `version` varchar(180) NOT NULL,
  `apply_time` int DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `migration`:
--

-- --------------------------------------------------------

--
-- Table structure for table `organization_detail`
--

CREATE TABLE `organization_detail` (
  `id` int NOT NULL,
  `id_user` int NOT NULL,
  `id_state` int NOT NULL,
  `id_attachment` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `type` smallint NOT NULL COMMENT '1 = Government, 2 = Private',
```



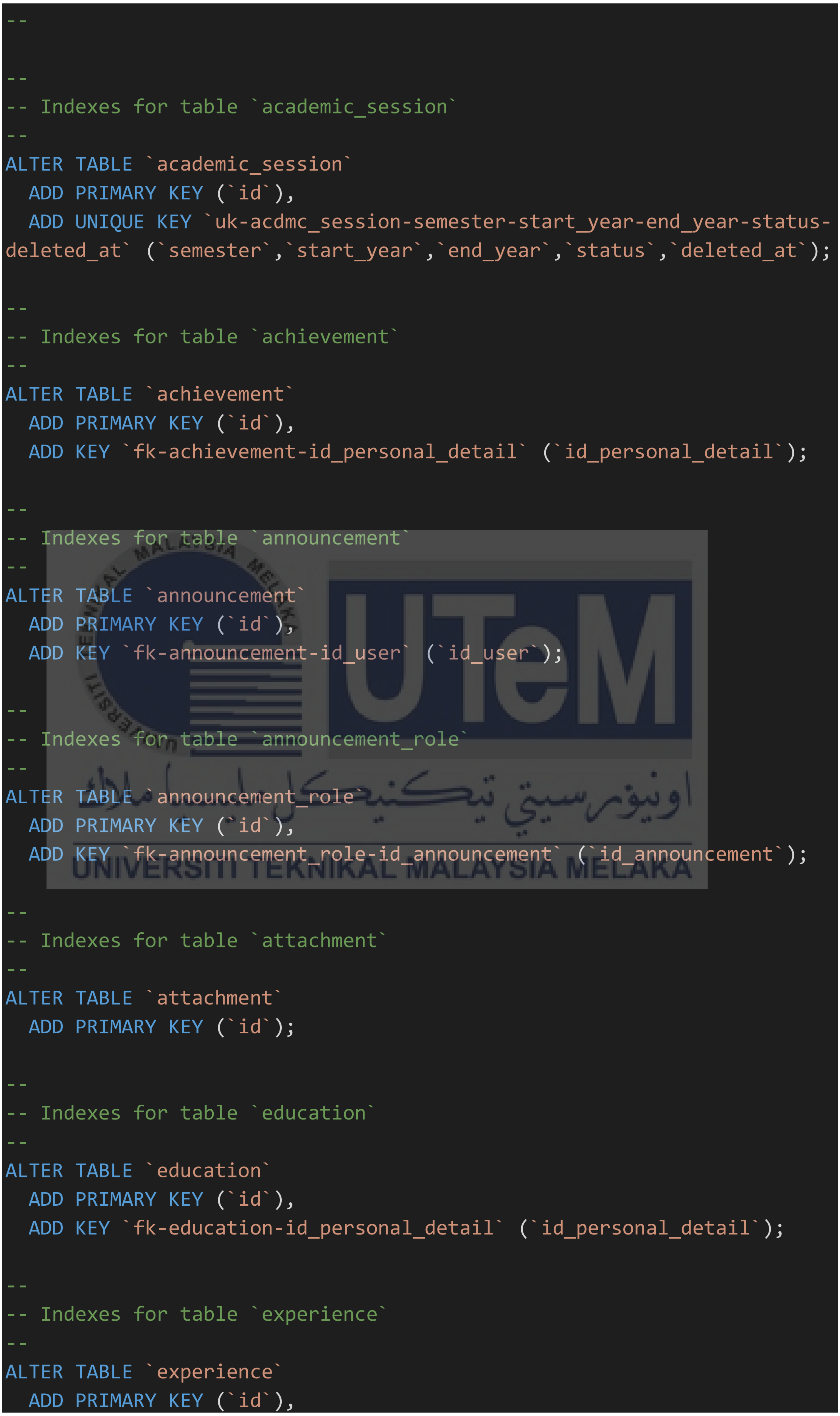
```
  `address` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_
ci NOT NULL,
  `tel_no` int NOT NULL,
  `start_day` smallint NOT NULL COMMENT '0 = Monday, 1 = Tuesday, 2 = W
ednesday, 3 = Thursday, 4 = Friday, 5 = Saturday, 6 = Sunday',
  `end_day` smallint NOT NULL COMMENT '0 = Monday, 1 = Tuesday, 2 = Wed
nesday, 3 = Thursday, 4 = Friday, 5 = Saturday, 6 = Sunday',
  `open_hour` time NOT NULL,
  `close_hour` time NOT NULL,
  `job_description` text NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `organization_detail`:
--   `id_attachment`
--       `attachment` -> `id`
--   `id_state`
--       `lookup_state` -> `id`
--   `id_user`
--       `user` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `personal_detail`
--

CREATE TABLE `personal_detail` (
  `id` int NOT NULL,
  `id_user` int NOT NULL,
  `id_programme` int NOT NULL,
  `id_state` int DEFAULT NULL,
  `id_attachment` int DEFAULT NULL COMMENT 'Category = Profile Picture'
,
  `nric_no` bigint DEFAULT NULL,
  `user_no` varchar(255) NOT NULL COMMENT 'Staff No. or Matric No.',
  `name` varchar(255) NOT NULL,
  `gender_no` smallint NOT NULL COMMENT '1 = Male, 2 = Female',
  `religion` varchar(255) DEFAULT NULL,
  `tel_no` int DEFAULT NULL,
  `address` varchar(255) DEFAULT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
```



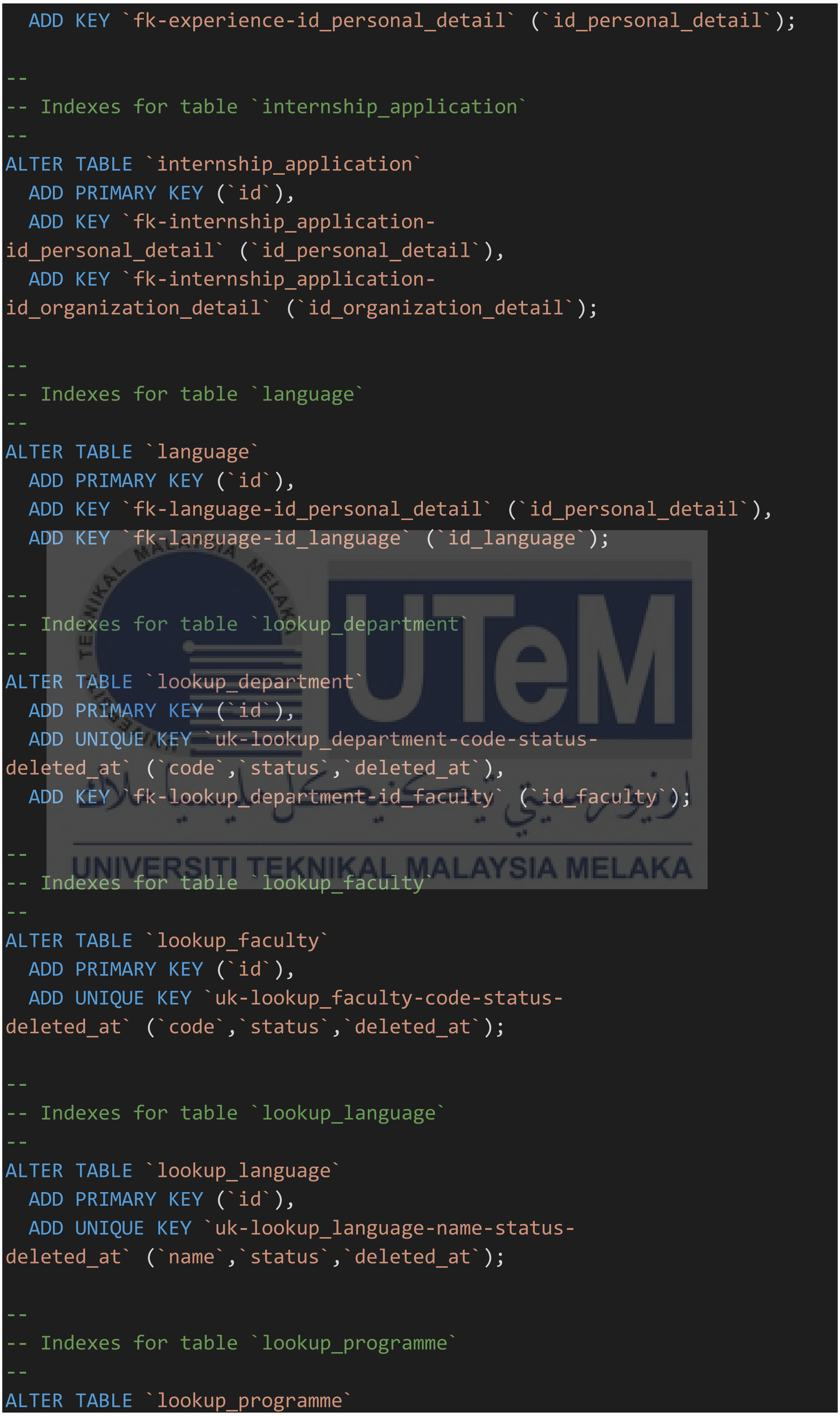
```
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `personal_detail`:
--   `id_attachment`
--       `attachment` -> `id`
--   `id_programme`
--       `lookup_programme` -> `id`
--   `id_state`
--       `lookup_state` -> `id`
--   `id_user`
--       `user` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `skill`
--

CREATE TABLE `skill` (
  `id` int NOT NULL,
  `id_personal_detail` int NOT NULL,
  `name` varchar(255) NOT NULL,
  `scale` smallint NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `skill`:
--   `id_personal_detail`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `supervision`
--

CREATE TABLE `supervision` (
  `id` int NOT NULL,
```



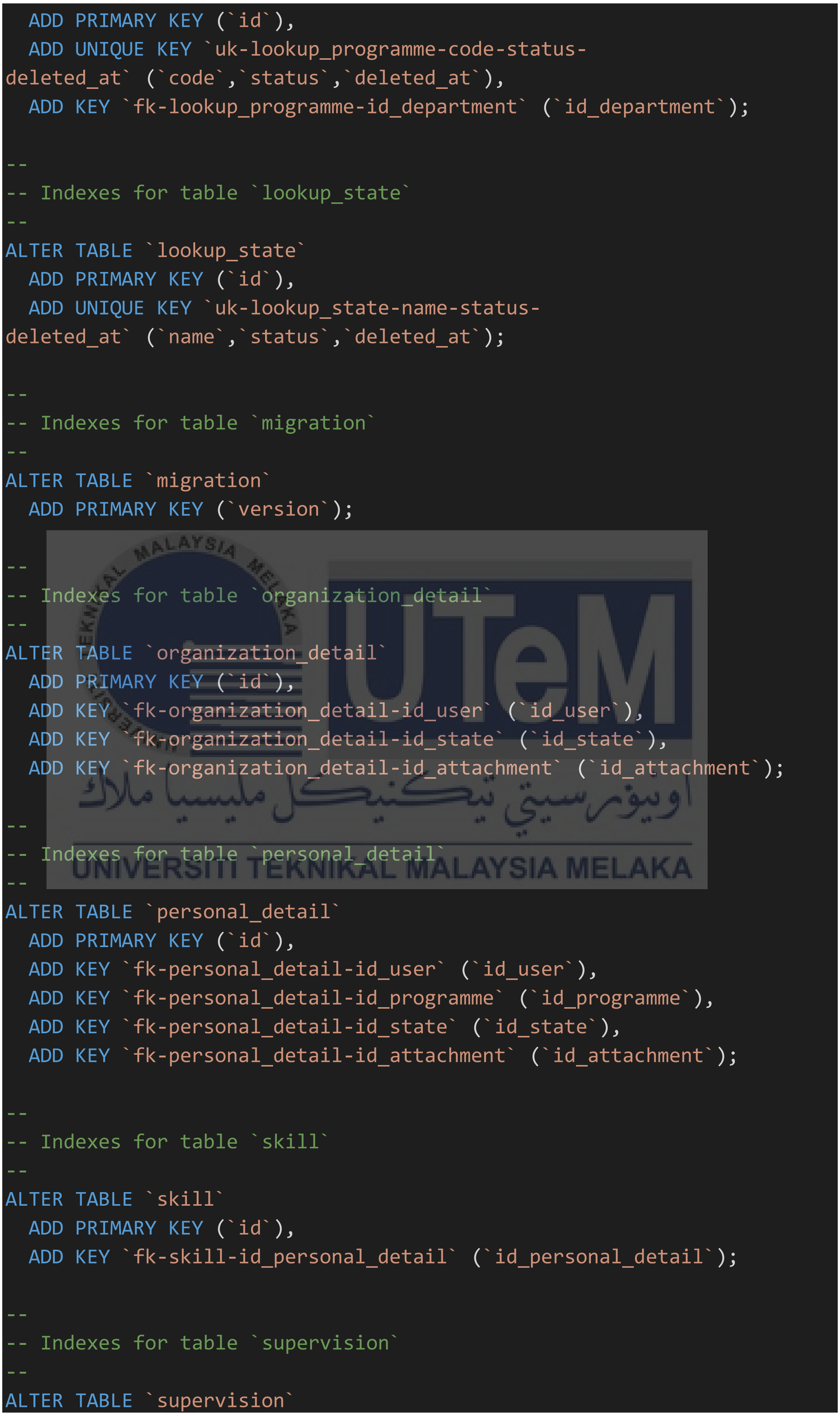
```
  `id_supervisor` int NOT NULL,
  `id_supervisee` int DEFAULT NULL,
  `id_academic_session` int NOT NULL,
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `supervision`:
--   `id_academic_session`
--       `academic_session` -> `id`
--   `id_supervisee`
--       `personal_detail` -> `id`
--   `id_supervisor`
--       `personal_detail` -> `id`
--

-- --------------------------------------------------------

--
-- Table structure for table `user`
--

CREATE TABLE `user` (
  `id` int NOT NULL,
  `username` varchar(255) NOT NULL,
  `auth_key` varchar(32) NOT NULL,
  `password_hash` varchar(255) NOT NULL,
  `password_reset_token` varchar(255) DEFAULT NULL,
  `verification_token` varchar(255) DEFAULT NULL,
  `email` varchar(255) NOT NULL,
  `role_no` smallint NOT NULL COMMENT '1 = Administrator, 2 = Coordinat
or, 3 = Supervisor, 4 = Student, 5 = Organization',
  `status` smallint NOT NULL DEFAULT '1' COMMENT '1 = Active, 0 = Inact
ive, -1 = Deleted',
  `created_at` int NOT NULL,
  `updated_at` int NOT NULL,
  `deleted_at` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- RELATIONSHIPS FOR TABLE `user`:
--

--
-- Indexes for dumped tables
```



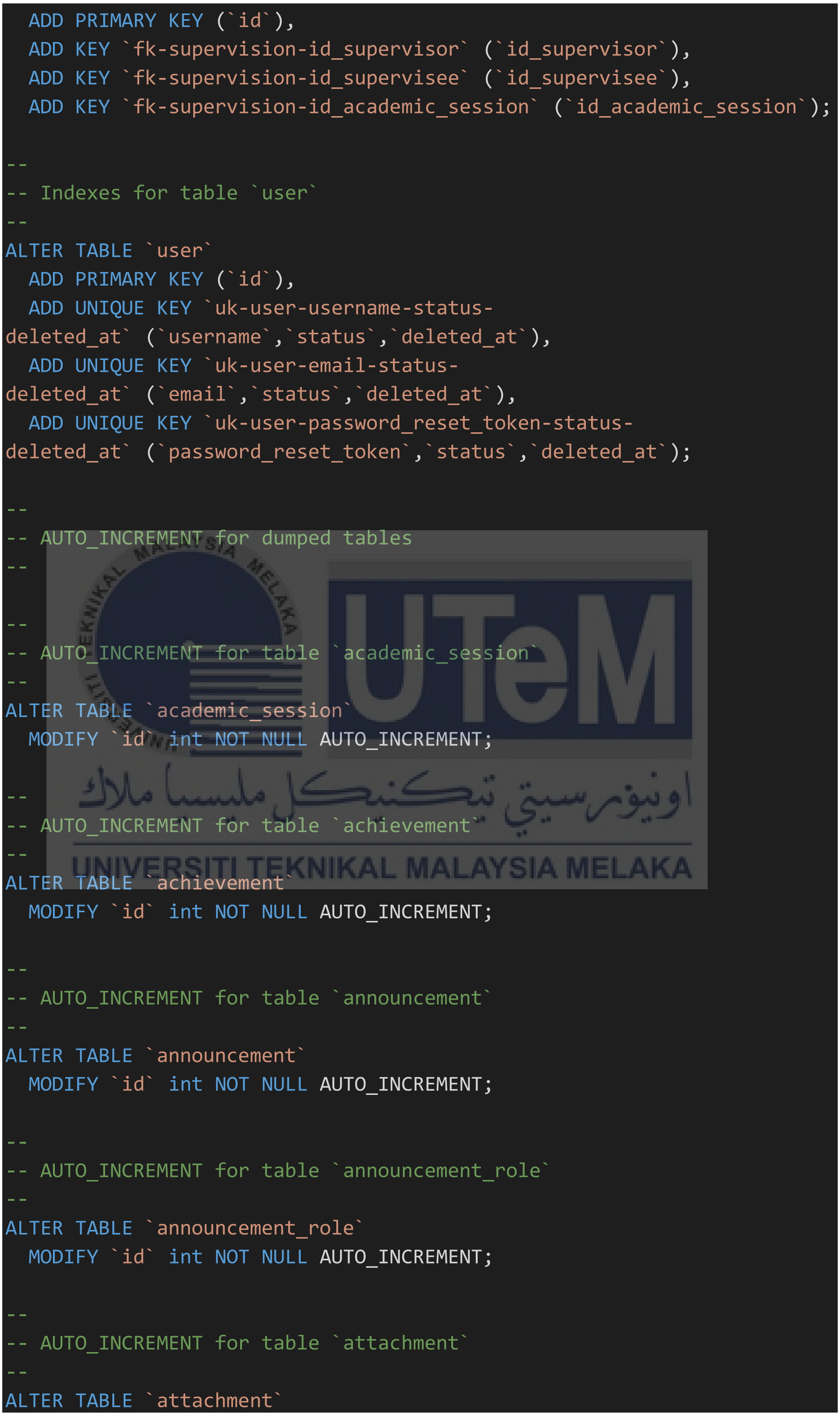
```
--

--
-- Indexes for table `academic_session`
--
ALTER TABLE `academic_session`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-acdmc_session-semester-start_year-end_year-status-
deleted_at` (`semester`,`start_year`,`end_year`,`status`,`deleted_at`);

--
-- Indexes for table `achievement`
--
ALTER TABLE `achievement`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-achievement-id_personal_detail` (`id_personal_detail`);

--
-- Indexes for table `announcement`
--
ALTER TABLE `announcement`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-announcement-id_user` (`id_user`);

--
-- Indexes for table `announcement_role`
--
ALTER TABLE `announcement_role`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-announcement_role-id_announcement` (`id_announcement`);

--
-- Indexes for table `attachment`
--
ALTER TABLE `attachment`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `education`
--
ALTER TABLE `education`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-education-id_personal_detail` (`id_personal_detail`);

--
-- Indexes for table `experience`
--
ALTER TABLE `experience`
  ADD PRIMARY KEY (`id`),
```



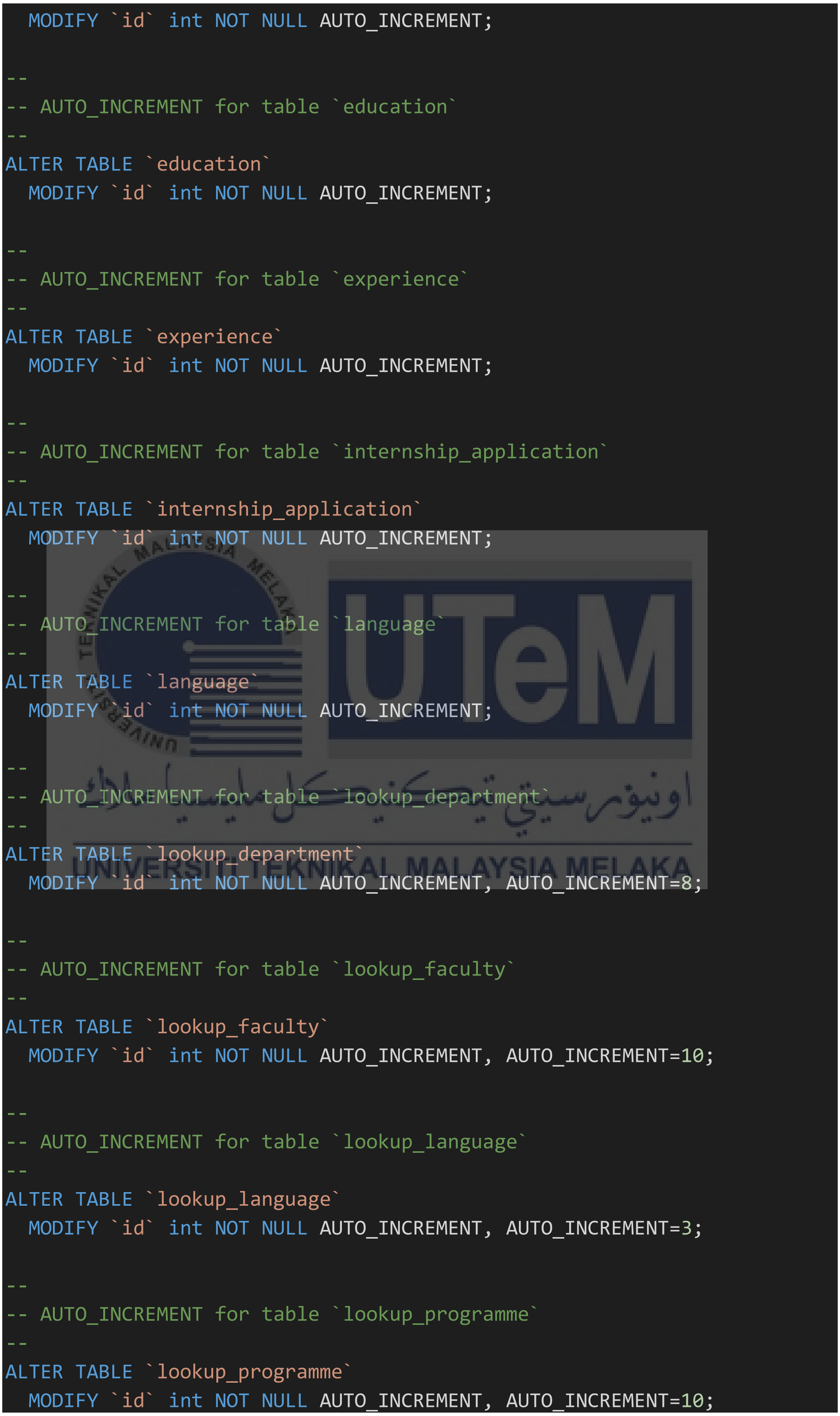
```
  ADD KEY `fk-experience-id_personal_detail` (`id_personal_detail`);

--
-- Indexes for table `internship_application`
--
ALTER TABLE `internship_application`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-internship_application-
id_personal_detail` (`id_personal_detail`),
  ADD KEY `fk-internship_application-
id_organization_detail` (`id_organization_detail`);

--
-- Indexes for table `language`
--
ALTER TABLE `language`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-language-id_personal_detail` (`id_personal_detail`),
  ADD KEY `fk-language-id_language` (`id_language`);

--
-- Indexes for table `lookup_department`
--
ALTER TABLE `lookup_department`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-lookup_department-code-status-
deleted_at` (`code`,`status`,`deleted_at`),
  ADD KEY `fk-lookup_department-id_faculty` (`id_faculty`);

--
-- Indexes for table `lookup_faculty`
--
ALTER TABLE `lookup_faculty`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-lookup_faculty-code-status-
deleted_at` (`code`,`status`,`deleted_at`);

--
-- Indexes for table `lookup_language`
--
ALTER TABLE `lookup_language`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-lookup_language-name-status-
deleted_at` (`name`,`status`,`deleted_at`);

--
-- Indexes for table `lookup_programme`
--
ALTER TABLE `lookup_programme`
```



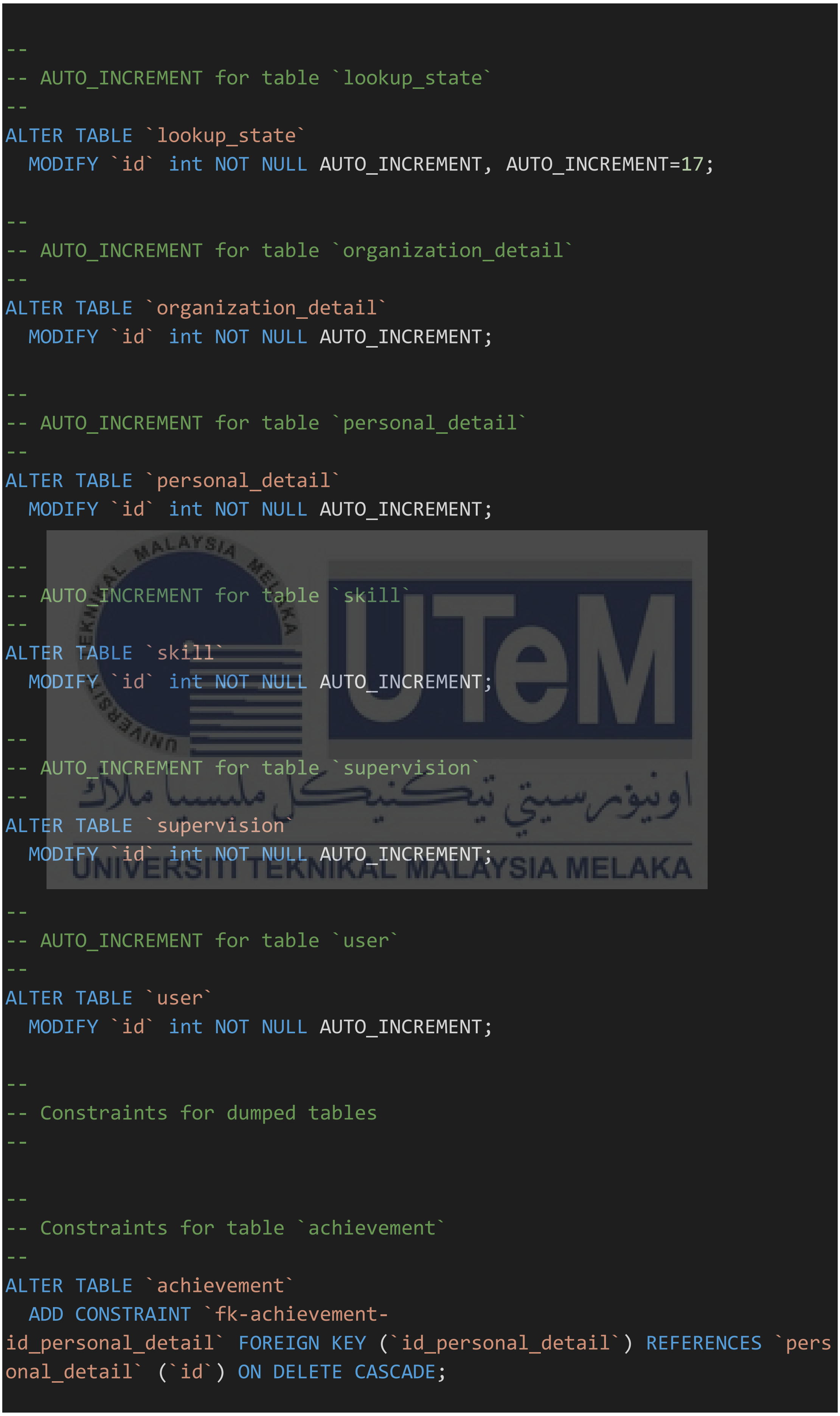
```
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-lookup_programme-code-status-
deleted_at` (`code`,`status`,`deleted_at`),
  ADD KEY `fk-lookup_programme-id_department` (`id_department`);

--
-- Indexes for table `lookup_state`
--
ALTER TABLE `lookup_state`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-lookup_state-name-status-
deleted_at` (`name`,`status`,`deleted_at`);

--
-- Indexes for table `migration`
--
ALTER TABLE `migration`
  ADD PRIMARY KEY (`version`);

--
-- Indexes for table `organization_detail`
--
ALTER TABLE `organization_detail`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-organization_detail-id_user` (`id_user`),
  ADD KEY `fk-organization_detail-id_state` (`id_state`),
  ADD KEY `fk-organization_detail-id_attachment` (`id_attachment`);

--
-- Indexes for table `personal_detail`
--
ALTER TABLE `personal_detail`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-personal_detail-id_user` (`id_user`),
  ADD KEY `fk-personal_detail-id_programme` (`id_programme`),
  ADD KEY `fk-personal_detail-id_state` (`id_state`),
  ADD KEY `fk-personal_detail-id_attachment` (`id_attachment`);

--
-- Indexes for table `skill`
--
ALTER TABLE `skill`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-skill-id_personal_detail` (`id_personal_detail`);

--
-- Indexes for table `supervision`
--
ALTER TABLE `supervision`
```



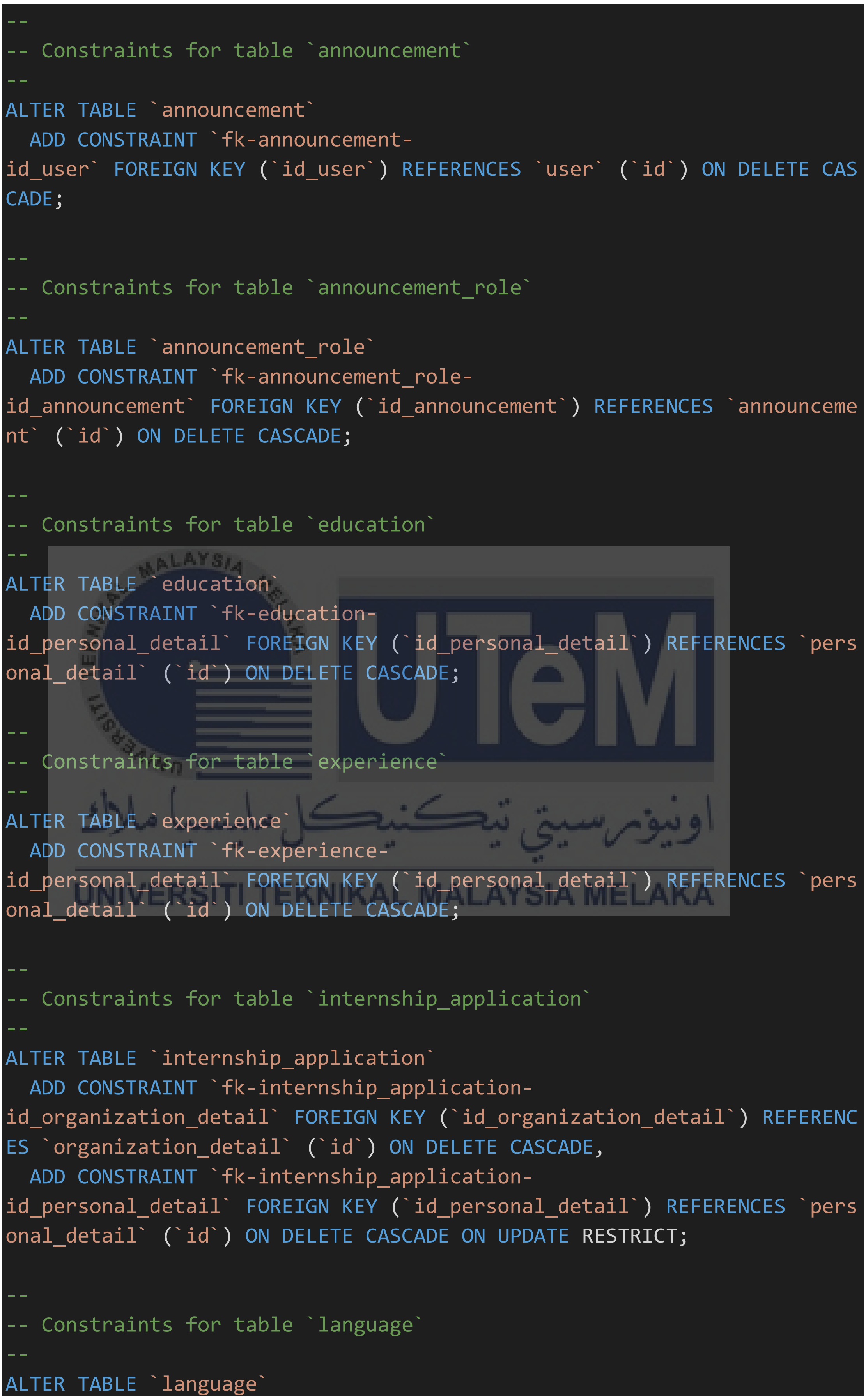
```
  ADD PRIMARY KEY (`id`),
  ADD KEY `fk-supervision-id_supervisor` (`id_supervisor`),
  ADD KEY `fk-supervision-id_supervisee` (`id_supervisee`),
  ADD KEY `fk-supervision-id_academic_session` (`id_academic_session`);

--
-- Indexes for table `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `uk-user-username-status-
deleted_at` (`username`,`status`,`deleted_at`),
  ADD UNIQUE KEY `uk-user-email-status-
deleted_at` (`email`,`status`,`deleted_at`),
  ADD UNIQUE KEY `uk-user-password_reset_token-status-
deleted_at` (`password_reset_token`,`status`,`deleted_at`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `academic_session`
--
ALTER TABLE `academic_session`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `achievement`
--
ALTER TABLE `achievement`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `announcement`
--
ALTER TABLE `announcement`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `announcement_role`
--
ALTER TABLE `announcement_role`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `attachment`
--
ALTER TABLE `attachment`
```



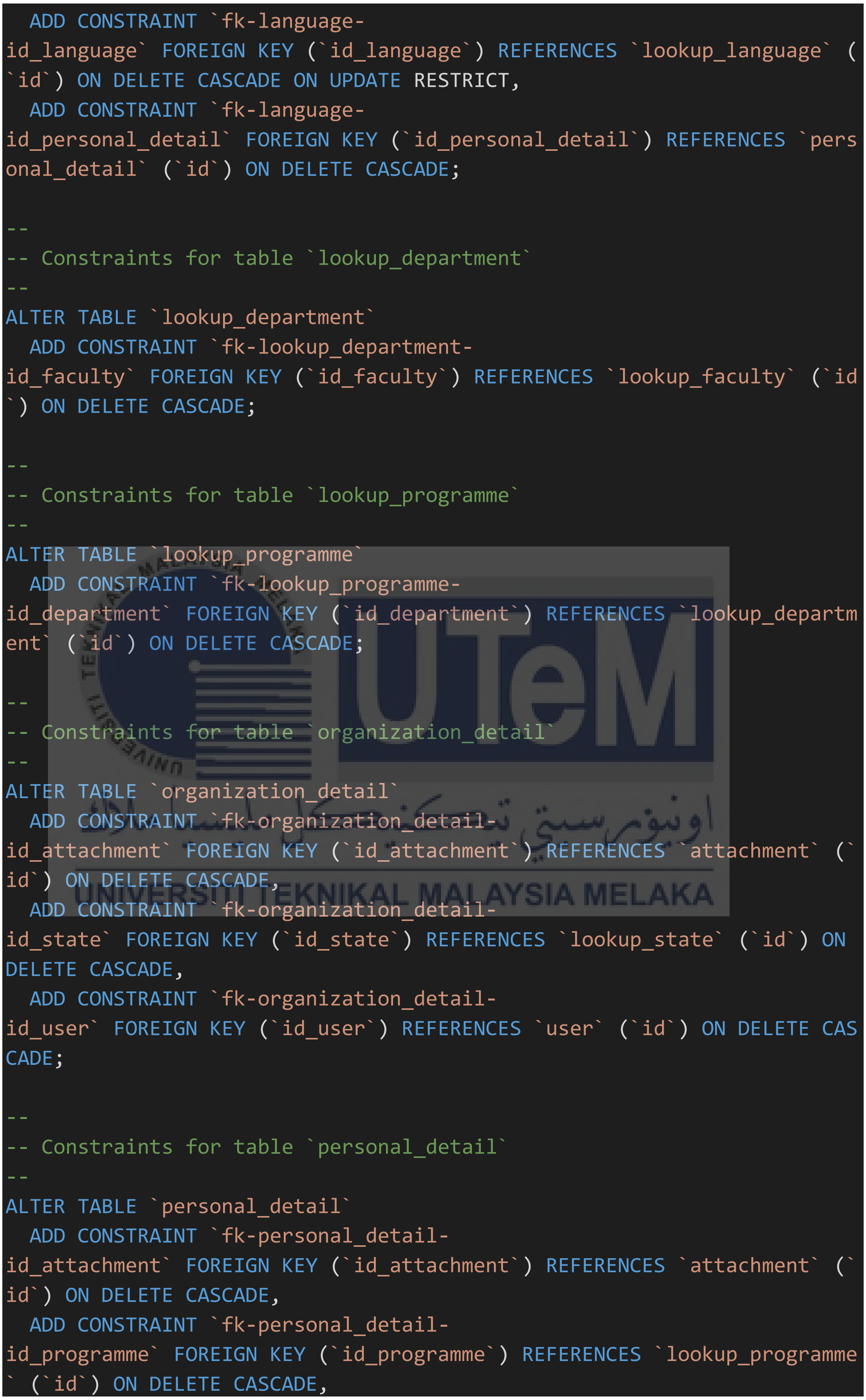
```
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `education`
--
ALTER TABLE `education`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `experience`
--
ALTER TABLE `experience`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `internship_application`
--
ALTER TABLE `internship_application`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `language`
--
ALTER TABLE `language`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `lookup_department`
--
ALTER TABLE `lookup_department`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;

--
-- AUTO_INCREMENT for table `lookup_faculty`
--
ALTER TABLE `lookup_faculty`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=10;

--
-- AUTO_INCREMENT for table `lookup_language`
--
ALTER TABLE `lookup_language`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

--
-- AUTO_INCREMENT for table `lookup_programme`
--
ALTER TABLE `lookup_programme`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=10;
```



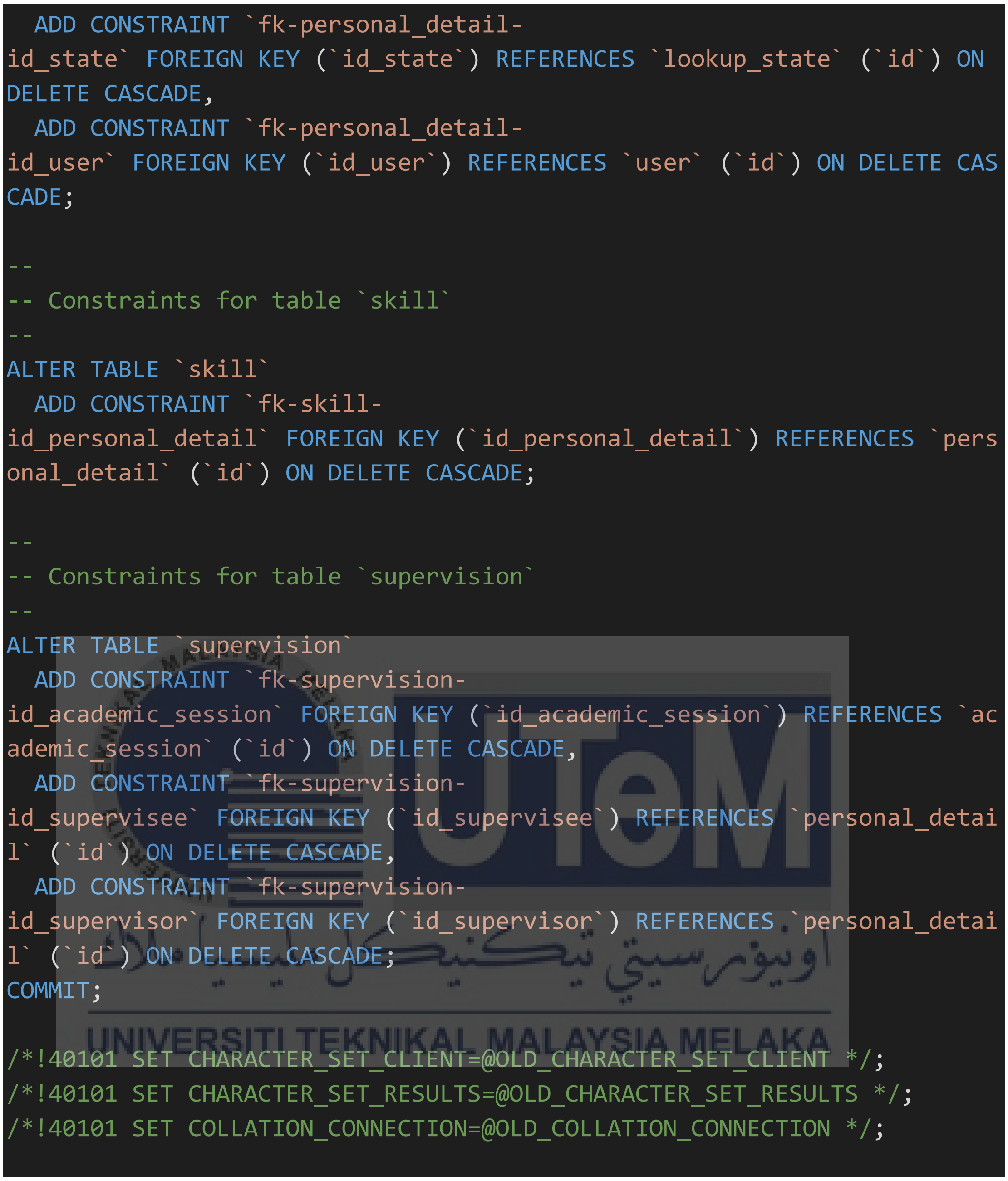
```
--
-- AUTO_INCREMENT for table `lookup_state`
--
ALTER TABLE `lookup_state`
  MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=17;

--
-- AUTO_INCREMENT for table `organization_detail`
--
ALTER TABLE `organization_detail`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `personal_detail`
--
ALTER TABLE `personal_detail`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `skill`
--
ALTER TABLE `skill`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `supervision`
--
ALTER TABLE `supervision`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `user`
--
ALTER TABLE `user`
  MODIFY `id` int NOT NULL AUTO_INCREMENT;

--
-- Constraints for dumped tables
--

--
-- Constraints for table `achievement`
--
ALTER TABLE `achievement`
  ADD CONSTRAINT `fk-achievement-id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `personal_detail` (`id`) ON DELETE CASCADE;
```



```
--
-- Constraints for table `announcement`
--
ALTER TABLE `announcement`
  ADD CONSTRAINT `fk-announcement-
id_user` FOREIGN KEY (`id_user`) REFERENCES `user` (`id`) ON DELETE CAS
CADE;

--
-- Constraints for table `announcement_role`
--
ALTER TABLE `announcement_role`
  ADD CONSTRAINT `fk-announcement_role-
id_announcement` FOREIGN KEY (`id_announcement`) REFERENCES `announceme
nt` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `education`
--
ALTER TABLE `education`
  ADD CONSTRAINT `fk-education-
id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `pers
onal_detail` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `experience`
--
ALTER TABLE `experience`
  ADD CONSTRAINT `fk-experience-
id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `pers
onal_detail` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `internship_application`
--
ALTER TABLE `internship_application`
  ADD CONSTRAINT `fk-internship_application-
id_organization_detail` FOREIGN KEY (`id_organization_detail`) REFERENC
ES `organization_detail` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-internship_application-
id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `pers
onal_detail` (`id`) ON DELETE CASCADE ON UPDATE RESTRICT;

--
-- Constraints for table `language`
--
ALTER TABLE `language`
```



```
  ADD CONSTRAINT `fk-language-id_language` FOREIGN KEY (`id_language`) REFERENCES `lookup_language` (`id`) ON DELETE CASCADE ON UPDATE RESTRICT,
  ADD CONSTRAINT `fk-language-id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `personal_detail` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `lookup_department`
--
ALTER TABLE `lookup_department`
  ADD CONSTRAINT `fk-lookup_department-id_faculty` FOREIGN KEY (`id_faculty`) REFERENCES `lookup_faculty` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `lookup_programme`
--
ALTER TABLE `lookup_programme`
  ADD CONSTRAINT `fk-lookup_programme-id_department` FOREIGN KEY (`id_department`) REFERENCES `lookup_department` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `organization_detail`
--
ALTER TABLE `organization_detail`
  ADD CONSTRAINT `fk-organization_detail-id_attachment` FOREIGN KEY (`id_attachment`) REFERENCES `attachment` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-organization_detail-id_state` FOREIGN KEY (`id_state`) REFERENCES `lookup_state` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-organization_detail-id_user` FOREIGN KEY (`id_user`) REFERENCES `user` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `personal_detail`
--
ALTER TABLE `personal_detail`
  ADD CONSTRAINT `fk-personal_detail-id_attachment` FOREIGN KEY (`id_attachment`) REFERENCES `attachment` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-personal_detail-id_programme` FOREIGN KEY (`id_programme`) REFERENCES `lookup_programme` (`id`) ON DELETE CASCADE,
```



```
  ADD CONSTRAINT `fk-personal_detail-
id_state` FOREIGN KEY (`id_state`) REFERENCES `lookup_state` (`id`) ON
DELETE CASCADE,
  ADD CONSTRAINT `fk-personal_detail-
id_user` FOREIGN KEY (`id_user`) REFERENCES `user` (`id`) ON DELETE CAS
CADE;

--
-- Constraints for table `skill`
--
ALTER TABLE `skill`
  ADD CONSTRAINT `fk-skill-
id_personal_detail` FOREIGN KEY (`id_personal_detail`) REFERENCES `pers
onal_detail` (`id`) ON DELETE CASCADE;

--
-- Constraints for table `supervision`
--
ALTER TABLE `supervision`
  ADD CONSTRAINT `fk-supervision-
id_academic_session` FOREIGN KEY (`id_academic_session`) REFERENCES `ac
ademic_session` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-supervision-
id_supervisee` FOREIGN KEY (`id_supervisee`) REFERENCES `personal_detai
l` (`id`) ON DELETE CASCADE,
  ADD CONSTRAINT `fk-supervision-
id_supervisor` FOREIGN KEY (`id_supervisor`) REFERENCES `personal_detai
l` (`id`) ON DELETE CASCADE;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

APPENDIX E: USER ACCEPTANCE TESTING - ORGANIZATION

Test Case ID:	FIPS_UAT_G00		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	User Authentication			
Test Description:	Verify Login with Valid Username and Password			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to User Login	Display Login Form		

2	Login using:
	Username = sitma
	Password = sitma0123
		Logged in into the System		

Overall Comments:				
Test Executed by:			**Name:**	
			Title:	

	Date:	

Test Case ID:	FIPS_UAT_G01		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Dashboard			
Test Description:	Display Dashboard			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Home (Dashboard)	Display Announcement and Dashboard		
Overall Comments:				
Test Executed by:			**Name:**	
			Title:	
			Date:	

Test Case ID:	FIPS_UAT_G02		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Organization Profile			
Test Description:	Display and Update Organization Profile			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Organization Profile	Display Organization Profile		
2	Click the Update Profile button	Display Update Profile Page		
3	Update the Profile Data	Profile Data Updated		
Overall Comments:				
Test Executed by:			**Name:**	
			Title:	
			Date:	

Test Case ID:	FIPS_UAT_G03		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Student Applicant			
Test Description:	Accept or Reject Student Applicant			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Student Applicant	Display Student Applicant List		
2	View Student ‘Zaki’	Display Zaki’s Resume		
3	Approve Applicant	Student Applicant Approved, the Status Will Be Changed to ‘Awaiting Student Reply’		
4	View Student ‘Puteh’	Display Puteh’s Resume		
5	Reject Applicant	Student Applicant Rejected, the Status		

		Will Be Changed to ‘Rejected		
			Overall Comments:	
		Test Executed by:	**Name:**	
			Title:	
			Date:	

APPENDIX F: USER ACCEPTANCE TESTING – STUDENT 1

Test Case ID:	FIPS_UAT_S10		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	User Authentication			
Test Description:	Verify Login with Valid Username and Password			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to User Login	Display Login Form	Pass	

2	Login using:
	Username = b031910230
	Password = amin0123
		Logged in into the System	Pass	

			Overall Comments:	Put announcement at first page for student alert
		Test Executed by:	**Name:**	Amin

	Faculty/Programme:	FTMK / MITS
	Date:	10 / 9 /2021

Test Case ID:	FIPS_UAT_S11		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Dashboard			
Test Description:	Display Dashboard			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Home (Dashboard)	Display Announcement and List of Application	Pass	
			Overall Comments:	
		Test Executed by:	**Name:**	Amin
			Faculty/Programme:	FTMK / MITS
			Date:	10 / 9 /2021

Test Case ID:	FIPS_UAT_S12		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Resume			
Test Description:	Display and Update Resume			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Resume	Display Resume	pass	
2	Click the Update Resume Button	Display Update Resume Page	Pass	Full must fill in all data. All required not suitable for user
3	Update Resume Information	Resume Information Updated	pass	
			Overall Comments:	
		Test Executed by:	**Name:**	Amin
			Faculty/Programme:	FTMK / MITS
			Date:	10 / 9 /2021

Test Case ID:	FIPS_UAT_S13		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Internship Application			
Test Description:	Apply Internship at Organization			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, List of Organization	Display List of Organization	Pass	
2	View Organization SITMA Solution	Display Organization Profile	Pass	
3	Click the Apply Internship Button	Internship Applied	Pass	
4	Navigate to Home, to View the List of Application	Display List of Application, and the Latest Application Status Will Be Shown as ‘In Review’	Pass	

5.	After the Organization Has Accept the Student Application, Accept the Organization Approval	The Status Will Be Updated to ‘Student Accepted’ for SITMA Solution, and All Other Pending Applications Will Be Rejected	Pass	Student can apply even the first company accepted
			Overall Comments:	
		Test Executed by:	**Name:**	Amin
			Faculty/Programme:	FTMK / MITS
			Date:	10 / 9 /2021

Test Case ID:	FIPS_UAT_S14		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Compare Organization			
Test Description:	Compare Organization Details			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, Compare Organization	Display Compare Organization Table	Pass	
2	Choose Organization in the Dropdown List	Display Organization Details	Pass	Student Cannot clear the selection
			Overall Comments:	
		Test Executed by:	**Name:**	Amin
			Faculty/Programme:	FTMK / MITS
			Date:	10 / 9 /2021

APPENDIX G: USER ACCEPTANCE TESTING – STUDENT 2

Test Case ID:	FIPS_UAT_S20		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	User Authentication			
Test Description:	Verify Login with Valid Username and Password			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to User Login	Display Login Form	Pass	

2	Login using:
	Username = b031910195
	Password = asyraf0123
		Logged in into the System	Pass	

			Overall Comments:	The result produce as expected.
		Test Executed by:	**Name:**	Mohd Asyraf

	Faculty/Programme:	FTMK / BITS
	Date:	10/09/2021

Test Case ID:	FIPS_UAT_S21		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Dashboard			
Test Description:	Display Dashboard			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Home (Dashboard)	Display Announcement and List of Application	Pass	
			Overall Comments:	Announcement and list of application displayed.
		Test Executed by:	**Name:**	Mohd Asyraf
			Faculty/Programme:	FTMK / BITS
			Date:	10/09/2021

Test Case ID:	FIPS_UAT_S22		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Resume			
Test Description:	Display and Update Resume			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Resume	Display Resume	Pass	
2	Click the Update Resume Button	Display Update Resume Page	Pass	
3	Update Resume Information	Resume Information Updated	Pass	Cannot select year when add new achievement
			Overall Comments:	Resume successfully updated.
		Test Executed by:	**Name:**	Mohd Asyraf
			Faculty/Programme:	FTMK / BITS
			Date:	10/09/2021

Test Case ID:	FIPS_UAT_S23		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Internship Application			
Test Description:	Apply Internship at Organization			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, List of Organization	Display List of Organization	Pass	
2	View Organization SITMA Solution	Display Organization Profile	Pass	
3	Click the Apply Internship Button	Internship Applied	Pass	
4	Navigate to Home, to View the List of Application	Display List of Application, and the Latest Application Status Will Be Shown as 'In Review'	Pass	

5.	After the Organization Has Accept the Student Application, Accept the Organization Approval	The Status Will Be Updated to 'Student Accepted' for SITMA Solution, and All Other Pending Applications Will Be Rejected	Pass	
			Overall Comments:	The result produce as expected
		Test Executed by:	**Name:**	Mohd Asyraf
			Faculty/Programme:	FTMK / BITS
			Date:	10/09/2021

Test Case ID:	FIPS_UAT_S24		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Compare Organization			
Test Description:	Compare Organization Details			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, Compare Organization	Display Compare Organization Table	Pass	
2	Choose Organization in the Dropdown List	Display Organization Details	Pass	
			Overall Comments:	The result produce as expected.
Test Executed by:			**Name:**	Mohd Asyraf
			Faculty/Programme:	FTMK / BITS
			Date:	10/09/2021

APPENDIX H: USER ACCEPTANCE TESTING – STUDENT 3

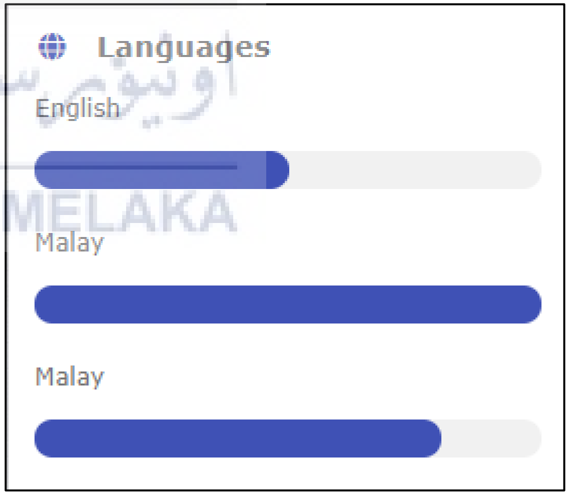
Test Case ID:	FIPS_UAT_S30		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	User Authentication			
Test Description:	Verify Login with Valid Username and Password			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to User Login	Display Login Form	**Pass**	

2	Login using:
	Username = b031910253
	Password = achikku0123
		Logged in into the System	**Pass**	

Overall Comments:				No comment
Test Executed by:			**Name:**	Muhammad Shahrul Nizam

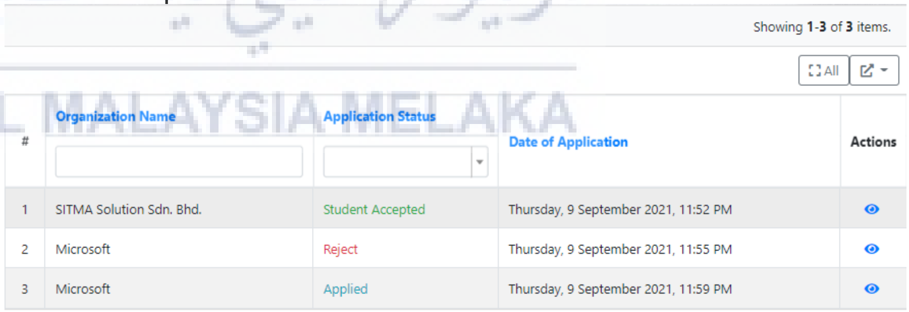
	Faculty/Programme:	FTMK / BITD
	Date:	9/9/2021

Test Case ID:	FIPS_UAT_S31		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Dashboard			
Test Description:	Display Dashboard			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Home (Dashboard)	Display Announcement and List of Application	**Pass**	
			Overall Comments:	No comment
		Test Executed by:	**Name:**	Muhammad Shahrul Nizam
			Faculty/Programme:	FTMK / BITD
			Date:	9/9/2021

Test Case ID:	FIPS_UAT_S32		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Resume			
Test Description:	Display and Update Resume			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Resume	Display Resume	**Pass**	
2	Click the Update Resume Button	Display Update Resume Page	**Pass**	Suggestion: - input skill use dropdown or suggestion with existing skill for make it easy for user Logic error: - language can insert multiple even exist before  Languages English Malay Malay

3	Update Resume Information	Resume Information Updated	**Pass**	
			Overall Comments:	Overall okey, need more improvement
		Test Executed by:	**Name:**	Muhammad Shahrul Nizam
			Faculty/Programme:	FTMK / BITD
			Date:	9/9/2021

Test Case ID:	FIPS_UAT_S33		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Internship Application			
Test Description:	Apply Internship at Organization			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, List of Organization	Display List of Organization	**Pass**	
2	View Organization SITMA Solution	Display Organization Profile	**Pass**	
3	Click the Apply	Internship Applied	**Pass**	- location button Apply Internship not suitable, make me confuse where the button

	Internship Button			
4	Navigate to Home, to View the List of Application	Display List of Application, and the Latest Application Status Will Be Shown as 'In Review'	**Pass**	
5.	After the Organization Has Accept the Student Application, Accept the Organization Approval	The Status Will Be Updated to 'Student Accepted' for SITMA Solution, and All Other	**Pass**	- as a student I still can apply other internship even I already accept another offer  Showing 1-3 of 3 items. # | Organization Name | Application Status | Date of Application | Actions 1 | SITMA Solution Sdn. Bhd. | Student Accepted | Thursday, 9 September 2021, 11:52 PM 2 | Microsoft | Reject | Thursday, 9 September 2021, 11:55 PM 3 | Microsoft | Applied | Thursday, 9 September 2021, 11:59 PM

		Pending Applications Will Be Rejected		
Overall Comments:				Overall okey, just test case sentences make me confuse
Test Executed by:			Name:	Muhammad Shahrul Nizam
			Faculty/Programme:	FTMK / BITD
			Date:	9/9/2021

Test Case ID:	FIPS_UAT_S34		**Test Designed by:**	Ameerul Zaki bin Azlan Raous
Test Title:	Compare Organization			
Test Description:	Compare Organization Details			
Step No.	**Test Steps**	**Expected Results**	**Status (Pass / Fail)**	**Notes**
1	Navigate to Application, Compare Organization	Display Compare Organization Table	**Pass**	
2	Choose Organization in the Dropdown List	Display Organization Details	**Pass**	
			Overall Comments:	No Comment
Test Executed by:			**Name:**	Muhammad Shahrul Nizam
			Faculty/Programme:	FTMK / BITD
			Date:	9/9/2021