

UNIVERSITI POLY-TECH MALAYSIA

TUITION MANAGEMENT SYSTEM

MOHAMAD DANIEL BIN AULADI

**BACHELOR OF INFORMATION
TECHNOLOGY (HONS) IN BUSINESS
COMPUTING**

UNIVERSITI POLY-TECH MALAYSIA
Faculty of Computing & Multimedia

TUITION MANAGEMENT SYSTEM

MOHAMAD DANIEL BIN AULADI
AM2311015219

FYP4105

MARCH 2026

Declaration of Originality

The given project is my original work and was not copied in part or in whole by some other source unless it was properly credited. In this regard, any use of prior work (book, journals, magazines, internet etc.) has been recognized in the main report as an item in the References or Bibliography lists.

I also accept that this project could be electronically copied and stored with the purpose of prevention and detection of plagiarism.

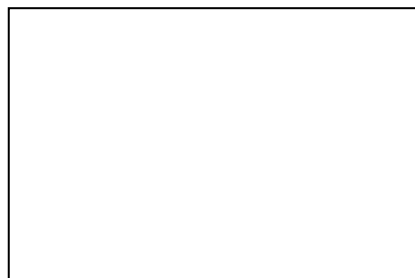
Copyright Acknowledgement

I also admit that the copyright of this project and report is vested in Universiti Poly-Tech Malaysia.

Signed:



Date: 13.11.2025



Office Stamp

Abstract

Tuition Management System (TMS) is an online solution that was created to increase the administrative effectiveness of the tuition centres by automating the main management processes. The project is aimed at resolving three significant issues that such institutions usually have: there is a manual process of working with student records, classes are not scheduled correctly, and payment tracking is not done correctly. The proposed TMS is a combination of academic and financial management in one centralized system so that administrators can handle the data of students, classes, and payments effectively. The system is built on Java, CSS, HTML and JavaScripts and MySQL which uses the Agile approach to guarantee user feedback and iterative development and improvements. The online registration system reduces the amount of data lost and the chances of human error as the student data is safely stored in a computer database. Automated scheduling will help avoid timetable conflicts as the availability of teachings and classrooms is verified in real-time and the fee management module allows automated billing, payments tracking, and financial reporting. Testing confirmed that the system performs accurately, securely, and reliably, fulfilling all functional and non-functional requirements. Overall, the Tuition Management System provides a cost-effective and user-friendly solution for small to medium-sized tuition centres, promoting transparency, operational efficiency, and data integrity in educational management.

Table of Contents

- 1 INTRODUCTION..... 15**
 - 1.1 Introduction15**
 - 1.2 Project Background.....15**
 - 1.3 Problem Statement.....16**
 - 1.3.1 Manual student record management lead to the risk of data loss and error 16
 - 1.3.2 Difficulty in managing class schedules without conflicts 16
 - 1.3.3 Traditional payment tracking makes it difficult to maintain accurate financial records 16
 - 1.4 Project Objectives17**
 - 1.4.1 To Investigate how an online registration module can reduce data loss and errors 17
 - 1.4.2 To develop a class scheduling feature that prevents timetable conflicts.....17
 - 1.4.3 To evaluate a fee management system with billing and payment tracking for accuracy and efficiency..... 17
 - 1.5 Scope and Target User.....18**
 - 1.5.1 Project Scope18
 - 1.5.2 Product Scope18
 - 1.5.3 Target User.....18
 - 1.6 Overview of This Report.....19**
- 2 LITERATURE REVIEW..... 20**
 - 2.1 Introduction20**
 - 2.2 Investigation20**
 - 2.2.1 Manual Student Record Management.....20
 - 2.2.2 Difficulty in Managing Class Schedules Without Conflicts21
 - 2.2.3 Traditional Payment Tracking and Financial Record Issues22
 - 2.3 Related Works.....23**
 - 2.3.1 FACTS Tuition Management System.....23
 - 2.3.2 Blackbaud Tuition Management System25
 - 2.3.3 openSIS (Open Source Student Information System)26
 - 2.4 Comparison28**
 - 2.5 Discussion29**
 - 2.6 Conclusion.....30**
- 3 METHODOLOGY..... 32**
 - 3.1 Introduction32**
 - 3.2 Agile Methodology.....32**
 - 3.3 Phases in Agile Methodology33**
 - 3.3.1 Requirement Analysis33
 - 3.3.2 System Design.....33
 - 3.3.3 System Development.....33
 - 3.3.4 Testing.....34
 - 3.3.5 Deployment34
 - 3.3.6 Review.....34

3.4	Conclusion	35
4	REQUIREMENTS	36
4.1	Introduction	36
4.2	Data Gathering Techniques	36
4.2.1	Study of Existing Project	36
4.2.2	Questionnaire	37
4.3	Functional Requirement	37
4.4	Non-Function Requirement	38
4.5	System Requirement	39
4.5.1	Software requirements.....	39
4.5.2	Hardware Requirements	41
4.6	Conclusion	41
5	ANALYSIS	42
5.1	Introduction	42
5.2	Data Gathering Analysis	42
5.2.1	Questionnaire Analysis	42
5.3	Use Case Model	52
5.4	BPMN (Business Process Modelling Notation)	53
5.6	Conclusion	54
6	DESIGN	55
6.1	Introduction	55
6.2	Interface Design	56
6.2.1	Log in Page	56
6.2.2	Admin – Dashboard	56
6.2.3	Admin – Schedule.....	57
6.2.4	Admin – Payment	57
6.2.5	Admin – Student	58
6.2.6	Admin – Teacher	58
6.2.7	Admin – Report.....	59
6.2.8	Teacher – Schedule.....	59
6.2.9	Teacher – Report.....	60
6.2.10	Teacher – Profile	60
6.3	Database Design	60
6.3.1	Data Dictionary	60
6.3.2	Data Flow Diagram (DFD)	64
6.3.3	Entity Relational Diagram (ERD).....	65
6.4	Flow of the System	66
6.5	Conclusion	67
7	IMPLEMENTATION	67
7.1	Introduction	68
7.2	Execution Platform	68
7.3	Implementation Tools	69

- 7.3.1 Visual Studio Code69
- 7.3.2 IntelliJ69
- 7.3.3 MySQL Database70
- 7.4 System Interface.....70**
 - 7.4.1 Log in Page70
 - 7.4.2 Admin – Dashboard71
 - 7.4.3 Admin – Schedule.....72
 - 7.4.4 Admin – Payment72
 - 7.4.5 Admin – Student73
 - 7.4.6 Admin – Teachers.....73
 - 7.4.7 Admin – Student Report.....74
 - 7.4.8 Teacher – Schedule.....74
 - 7.4.9 Admin – Payment Record75
 - 7.4.10 Teacher – Reports75
 - 7.4.11 Teacher – Profile76
- 7.5 Significant Functions76**
 - 7.5.1 Login.....76
 - 7.5.2 Forgot Password.....77
- 7.6 Conclusion.....77**
- 8 TESTING.....78**
 - 8.1 Introduction78**
 - 8.2 Unit Testing.....78**
 - 8.3 Integration Testing79**
 - 8.4 System Testing80**
 - 8.4.1 Functional Testing80
 - 8.4.2 Non-Functional Testing.....81
 - 8.5 Acceptance Testing.....82**
 - 8.6 Conclusion.....98**
- 9 PROJECT MANAGEMENT99**
 - 9.1 Introduction99**
 - 9.2 Project Schedule99**
 - 9.2.1 Work Breakdown Structure100
 - 9.2.2 Gantt Chart101
 - 9.3 Risk Management102**
 - 9.4 Conclusion.....103**
- 10 CONCLUSION104**
 - 10.1 Introduction104**
 - 10.2 Achievement104**
 - 10.2.1 First Point of the Objective104
 - 10.2.2 Second Point of the Objective.....104
 - 10.2.3 Third Point of the Objective.....104
 - 10.3 Constraint and Limitation105**
 - 10.4 Future Work and Recommendation105**

10.5 Conclusion.....	105
Appendix A – Requirements Specification Document	106
Appendix B – User manual	111
Appendix C – Turnitin Result.....	118
Appendix D – Log Book	119
References	124

List of Figures

Figure 2. 1: FACTS Tuition Management System Dashboard 23

Figure 2. 2: FACTS Tuition Payment Activity 23

Figure 2. 3: Blackbaud Tuition Management 25

Figure 2. 4: Admin Portal 25

Figure 2. 5: openSIS System Grading 26

Figure 2. 6: openSIS Dashboard 27

Figure 3. 1: Agile Methodology Diagram (Marketing91, n.d.) 32

Figure 4. 1: MySQL 39

Figure 4. 2: CSS, HTML, JavaScript 39

Figure 4. 3: Java 39

Figure 4. 4: Bootstrap 40

Figure 4. 5: Spring Boot 40

Figure 4. 6: Visual Studio Code 40

Figure 4. 7: IntelliJ 41

Figure 5. 1: Question 1 42

Figure 5. 2: Question 2 43

Figure 5. 3: Question 3 43

Figure 5. 4: Question 4 43

Figure 5. 5: Question 5 44

Figure 5. 6: Question 6 44

Figure 5. 7: Question 7 44

Figure 5. 8: Question 8 45

Figure 5. 9: Question 9 45

Figure 5. 10: Question 10 45

Figure 5. 11: Question 11 46

Figure 5. 12: Question 12 46

Figure 5. 13: Question 1 46

Figure 5. 14: Question 2 47

Figure 5. 15: Question 3 47

Figure 5. 16: Question 4 47

Figure 5. 17: Question 5 48

Figure 5. 18: Question 6 48

Figure 5. 19: Question 7 48

Figure 5. 20: Question 8 49

Figure 5. 21: Question 9 49

Figure 5. 22: Question 10 49

Figure 5. 23: Question 11 50

Figure 5. 24: Question 12 50

Figure 5. 25: Question 13 50

Figure 5. 26: Question 14 51

Figure 5. 27: Use Case Model 52

Figure 5. 28: BPMN 53

Figure 6. 1: Login Page 56

Figure 6. 2: Admin Dashboard 56

Figure 6. 3: Admin Schedule 57

Figure 6. 4: Admin Payment	57
Figure 6. 5: Admin Student	58
Figure 6. 6: Admin Teacher	58
Figure 6. 7: Admin Report	59
Figure 6. 8: Teacher Schedule	59
Figure 6. 9: Teacher Report	60
Figure 6. 10: Teacher Profile	60
Figure 6. 11: User Database	61
Figure 6. 12: Admin Database	61
Figure 6. 13: Teacher Database	61
Figure 6. 14: Student Database	62
Figure 6. 15: Subject Database	62
Figure 6. 16: Schedule Database	62
Figure 6. 17: Payment Database	62
Figure 6. 18: Teach Database	63
Figure 6. 19: Student_pay Database	63
Figure 6. 20: Data Flow Diagram	64
Figure 6. 21: Entity Relational Diagram	65
Figure 6. 22: Flow of the system	66
Figure 7. 1: Execution Platform	68
Figure 7. 2: Vs Code	69
Figure 7. 3: IntelliJ	69
Figure 7. 4: MySQL Database	70
Figure 7. 5: Login Page	70
Figure 7. 6: Admin Dashboard	71
Figure 7. 7: Schedule	72
Figure 7. 8: Payment	72
Figure 7. 9: Student	73
Figure 7. 10: Teachers	73
Figure 7. 11: Student Report	74
Figure 7. 12: Teacher Schedule	74
Figure 7. 13: Payment Record	75
Figure 7. 14: Teacher Report	75
Figure 7. 15: Teacher Profile	76
Figure 7. 16: Login	76
Figure 7. 17: Forgot Password	77
Figure 8. 1: Question 1	84
Figure 8. 2: Question 2	84
Figure 8. 3: Question 3	85
Figure 8. 4: Question 4	85
Figure 8. 5: Question 5	86
Figure 8. 6: Question 6	86
Figure 8. 7: Question 7	87
Figure 8. 8: Question 8	87
Figure 8. 9: Question 9	88
Figure 8. 10: Question 10	88
Figure 8. 11: Question 11	89
Figure 8. 12: Question 12	89
Figure 8. 13: Question 13	90

Figure 8. 14: Question 14	90
Figure 8. 15: Question 15	91
Figure 8. 16: Question 1	91
Figure 8. 17: Question 2	92
Figure 8. 18: Question 3	92
Figure 8. 19: Question 4	93
Figure 8. 20: Question 5	93
Figure 8. 21: Question 6	94
Figure 8. 22: Question 7	94
Figure 8. 23: Question 8	95
Figure 8. 24: Question 9	95
Figure 8. 25: Question 10	96
Figure 8. 26: Question 11	96
Figure 8. 27: Question 12	97
Figure 8. 28: Question 13	97
Figure 8. 29: Question 14	98
Figure 8. 30: Question 15	98
Figure 9. 1: Work Breakdown Structure	100
Figure 9. 2: Gantt Chart	101

List of Tables

Table 2. 1: Comparison of Existing Project.....	28
Table 4. 1: Functional Requirement of Administrator	37
Table 4. 2: Functional Requirement of Teacher.....	38
Table 4. 3: Non-Function Requirement.....	38
Table 4. 4: Hardware Requirements.....	41
Table 8. 1: Unit Testing	78
Table 8. 2: Integration Testing.....	79
Table 8. 3: Functional system testing	80
Table 8. 4: Non-Functional system testing.....	81
Table 8. 5: Acceptance Testing	82
Table 9. 1: Risk Management.....	102

Acknowledgements

First and foremost, I would like to express my deepest gratitude to Allah SWT for His blessings, guidance, and strength that enabled me to successfully complete this Final Year Project entitled Tuition Management System.

To my supervisor, Miss Nur Afini Natrah Binti Mohd Ashril, I am very grateful that she has been very supportive, offered a lot of guidance and encouragement during the process of developing this project. Her positive criticism and experience have played a significant role in determining the success of this work. I will also wish to express my thanks to the Faculty of Computing and Multimedia, Universiti Poly-Tech Malaysia (UPTM) which provided me with the resources, facilities, and knowledge that assisted me a lot in the completion of this project.

The administrators and teachers of Pusat Tuisyen Sinar Wangsa are also thanked as they provided good cooperation and useful input during the data collection and testing process. Their views and recommendations played a vital role towards making the system practical and relevant.

Lastly, I would like to show my sincere gratitude to my parents, family and friends who have given me their unconditional love, patience and moral support during this journey. This would not have been made possible without their prayers, motivation and understanding.

1 INTRODUCTION

1.1 Introduction

This chapter gives an introduction to the proposed Tuition Management System (TMS). It provides the background of the project, the problem statement, objectives, the scope and target users, and it gives the general structure of the report. The TMS will be used to enhance the operation of the tuition centres with problems regarding manual student registration, schedule of classes and tracking of payment. As the digital technology is incorporated, the system will strive to reduce human error, enhance efficiency, and improve a proper and transparent record keeping.

1.2 Project Background

The use of modern technology has been integrated in the educational and business processes, which allows the institutions to simplify the processes and maximize efficiency. Technology is becoming more critical in the management of administrative functions in tuition centres, including; enrolment, scheduling, attendance, and payment management (Arias, Macatulad, and Malaluan, 2021). In the absence of automated system implementation, tuition centres are prone to issues like loss of data, repetition, and lack of communication between administrators, teachers, parents, and students.

A tuition management system or a Tuition Management System (TMS) is a key to solving these problems since it will offer a centralized system to manage its daily activities. Tuition centres can also be automated to minimize errors, facilitate decision-making, and boost communication, not to mention that transparency in financial records is guaranteed (Arctiq, n.d.). To illustrate, internet-based enrolment will decrease the chances of lost or repeated student information, computerized class scheduling will minimise scheduling clashes and computerised payment systems will provide financial oversight the way it should be.

The systems in place show how digital platforms can work but also indicate some gaps. FACTS Tuition Management and Blackbaud Tuition Management are highly popular systems that are effective in automating financial operations like billing, flexible payment arrangements, and reminders. Nevertheless, they also use other tools to support the academic features (FACTS Management, n.d.; Blackbaud, n.d.). On the one hand, openSIS concentrates on student data, time-table, and check-in attendance, but the financial managing system is very weak (openSIS, n.d.). Such constraints raise the necessity to have a system that combines both financial and academic management in a manner that is feasible in small to medium-sized tuition centres.

Considering issues above, this project suggests development of web-based TMS in the Pusat Tuisyen Sinar Wangsa. There will be three key modules that the system will concentrate on; these modules include student registration, class scheduling and fees management. With these features, the TMS will not only streamline the administrative processes but also the entire process of teaching and learning by all the stakeholders.

1.3 Problem Statement

1.3.1 Manual student record management lead to the risk of data loss and error

The risks of loss and errors in data and the records of students are common in the case of manual student record management. In situations where the tuition centres use paper files, notebooks, or simple spreadsheet programmes, the student data is prone to losing or being damaged or even destroyed. Also, manual input leads to the risk of error, including wrong data entry and duplicate or irregular update of various records (e.g. attendance, payments, or performance). Such problems not only result in the inaccuracy and unreliability of the data that pertains to the students but also render the process of retrieving or updating records to be time-consuming and inefficient by the administrators and teachers. Consequently, the management of records manually may impede the seamless working processes and lower the overall efficiency of the tuition centre administration (Arias, Macatulad, & Malaluan, 2021).

1.3.2 Difficulty in managing class schedules without conflicts

Inability to handle class schedules without some conflict is a widespread issue in tuition centres which have applied manual scheduling procedures. With whiteboards, paper schedules and basic spreadsheet, the administrators cannot coordinate teacher availability, classroom capacity, and subject choices of students. This tends to cause overlapping time schedules; two classes can be placed in either the same teacher or same room at the same time. Manual scheduling is also more difficult to adapt to since oftentimes the teacher may miss the change or there may be last minute student calls. On top of disrupting the teaching process, these conflicts also lead to frustration among the students, parents and staff. A timetable that is smooth and efficient cannot be realized without an automated system capable of identifying and avoiding clashes, as the process takes time and is subject to human error (openSIS, n.d.).

1.3.3 Traditional payment tracking makes it difficult to maintain accurate financial records

The old method of tracking the payment is not easy to keep proper financial books. A large number of tuition centres are still using manual systems like paper receipts, logbooks or simple

spreadsheets to keep track of student payments. These methods are likely to create human error, repetition and absence of entries and particularly when two or more administrators are engaged in processing payments. The delay in updating the records can also create confusion on the students who have paid and those who are outstanding and this can create argument with parents. Moreover, tracking manually does not provide transparency and complicates the creation of correct financial reports to be audited or used in decision-making. The more students there are, the more complicated and inefficient the process is, which makes it necessary to implement a digital payment management system that guarantees the accuracy, transparency and up-to-date results (FACTS Management, n.d.).

1.4 Project Objectives

1.4.1 To Investigate how an online registration module can reduce data loss and errors

The first goal is aimed at correcting the issue of using manual student records management that usually results in the loss of data, its duplication, and human error. Through research and implementation of an online-based registration module, the system will enable administrators to store and manage student and teacher records in a centralized database safely. This online system will give more accuracy in data, less chance of missing information and an easier means of accessing and updating the records as compared to the old method of paper-based.

1.4.2 To develop a class scheduling feature that prevents timetable conflicts

The second goal will address the problem of time-related challenges that is common at tuition centres. The use of manual scheduling may lead to overlaps in classes, bookings in the same classes, or conflicts in the availability of teachers. In order to overcome this, a scheduling feature will be part of the system that will automate the process of assigning classes, teachers, and rooms. This will make sure that schedules are properly structured and conflict free and thus increase the teaching efficiency and student learning experiences.

1.4.3 To evaluate a fee management system with billing and payment tracking for accuracy and efficiency

The third goal is to assess how digital fee management system can boost the financial business of the tuition centre. Conventional ways of tracking payments usually cause inaccuracies in the process of recording payments, loss of transactions, and inability to generate the correct financial reports. The system will streamline financial management by adding an automated system of billing and tracking payments by giving accurate records and giving clear recap of pending and paid ones. The assessment of this system will prove its functionality in providing proper financial control and sustainability of the tuition centre in the long run.

1.5 Scope and Target User

The next phases are the description of the scope of the project and product, as well as a description of the target users of the system. This is to make sure that the objectives and deliverables of the Tuition Management System (TMS) are clear and in line with the requirements of the intended users.

1.5.1 Project Scope

This project will involve the design of a stable and interactive web-based Tuition Management System that will be available via the internet to be used by Pusat Tuisyen Sinar Wangsa. The primary goal is to computerize some of the most important processes in the administration like student registration, scheduling of classes, and the management of fees that are currently being performed with the help of traditional methods. The system will enhance the quality of data and efficiency in the operations by substituting the inaccurate procedures, such as paper records, whiteboard scheduling, and manual payment tracking. It will use a system development life cycle (SDLC) developed in an agile manner to make the system flexible and enhance it. It will emphasize on three major modules that includes student and teacher registration and record management, automated class scheduling to eliminate timetable conflicts and fee management with billing and payment tracking.

1.5.2 Product Scope

The proposed Tuition Management System is aimed at becoming an effective and convenient digital system that combines management of administration and finances into a single system. It will allow administrators to save and access the data about students safely and create error-free schedules and keep track of payments with no obstacles. The teachers will find it easier to check the attendance and monitor the developments of the students. The emphasis of the product is on simplicity, accuracy, and reliability with a cost-effective solution that can be applicable in small and medium-sized tuition centres. Although the initial version will be a strong feature-oriented system, more functions can be later added to the system like mobile application support, online learning materials and improved reporting.

1.5.3 Target User

a) Administrators

The system will be used by administrators like managers to deal with registering students, class schedules, and even tuition payment. They will enjoy less work, fewer mistakes, and quicker accessibility to data and reports.

b) Teachers

The system will be used to record attendance and check schedules and monitor student performance by teachers. This will make them concentrate more on teaching by cutting administrative burdens.

1.6 Overview of This Report

- **Chapter 1 – Introduction:** In this chapter, the general background of the project is available, which involves introduction, project background, problem statement, project objectives, scope, and target users. It is the basis and rationale of the TMS development.
- **Chapter 2 – Literature Review:** The existing studies and systems, including FACTS Tuition Management, Blackbaud Tuition Management, and openSIS, have been reviewed in this chapter. It has pointed out their strengths, weaknesses, and gaps as a way of demonstrating why the proposed system is needed. Discussed also are relevant academic research in the area of education management and digital transformation to justify the project.
- **Chapter 3 – Methodology:** The chapter provides the methodology and development approach applied to the research. It explains the SDLC with an Agile methodology presenting requirement analysis, system design, development, testing, deployment, and maintenance.
- **Chapter 4 – Requirements:** This chapter gives an overview on how it is done; how project requirements are obtained using selected tools of information extraction.
- **Chapter 5 – Analysis:** A comprehensive discussion is present in this chapter concerning the data gathered during requirements gathering to finalization stage and subsequently visualization of findings of use diagrams.

2 LITERATURE REVIEW

2.1 Introduction

This chapter covers the literature and systems available concerning the tuition management platforms. This is aimed at determining the existing practices, technologies, and system constraints with the aim of coming up with a solid ground upon which to build a better Tuition Management System (TMS). With most tuition centres relying on manual methods of keeping records, scheduling and tracking payments, this review indicates the need to automate, integrate and come up with user friendly web solutions that can enhance efficiency and minimize human errors.

2.2 Investigation

This part will undertake a comprehensive exploration of the significant business issues that are experienced by tuition centres especially where the current management practices are traditional or manual. The research problem is the following: manual patient record management, problems in arranging classes, and old-fashioned payment tracking. The awareness of these issues contributes to explaining why a detailed digital solution should be proposed, which is the proposed Tuition Management System (TMS) that strives to enhance data accuracy, scheduling efficiency, and financial transparency.

The results of this research are built upon the data obtained during the interviews with administrators and feedback provided by teachers and the analysis of the existing management systems applied in the education institutions nowadays. These lessons were used to determine system requirements and shape the design of the proposed solution.

2.2.1 Manual Student Record Management

The use of manual process of handling student information has been one of the biggest problems of most tuition centres. These approaches usually presuppose the use of handwritten data, spreadsheets, or print registration forms that are inefficient and prone to errors (Arias et al., 2021). Manual record system is also prone to losing, duplicating and not updating effectively due to lack of centralized storage to store information on students. With an increase in the size of tuition centres, the student information about attendance, performance, and fee status becomes increasingly difficult to maintain manually (Nair & Saini, 2020).

The search found out that administrators in Pusat Tuisyen Sinar Wangsa waste a lot of time to access student data, check attendance, and update contacts. Mistake is a common phenomenon

particularly where records are handled by the various staff members working separately at the same time. Also, paper-based records can be damaged or lost easily, which poses serious threats of information loss and disruption of operations (Mohanty & Dash, 2022).

Sahu et al. (2020) also state that student records are highly accurate and accessible with the aid of digital transformation in educational administration. The centralized online registration module with the help of relational database like MySQL will allow real time updating of data, avoid duplication as well as consistency throughout all the departments. Moreover, social media systems that have role-based authentication are more secure than those with unlimited data access as they allow only authorized personnel to access them (Gupta & Sharma, 2022).

According to these results, the given TMS will feature a secure online registration and record management module in order to reduce the level of data loss and human error. This module will make the process of storing, retrieving and updating all student information such as attendance and academic performance digital and more reliable and efficient as compared to the manual system.

2.2.2 Difficulty in Managing Class Schedules Without Conflicts

The other issue that has been found during the investigation is the inability to structure the timetable of the classes manually. The majority of tuition centres continue to use the whiteboard, printed schedules, or even spreadsheet to organize classes, teachers and rooms. These systems are not automated and real-time validated and usually cause schedule overlaps and clashes with rooms (Rahman et al., 2019). As an example, a teacher can be assigned two classes simultaneously, or two classes can be booked in the same room because of poor communication between the staff members.

Administrators at Pusat Tuisyen Sinar Wangsa affirmed that manual timetable management takes a lot of time particularly when unexpected incidences arise including the absence of teachers or rescheduling requests. In the absence of centralized scheduling system, every change will have to be verified by hand in several records, which will lead to higher chances of discrepancy and misunderstanding. Chavda and Solanki (2021) assert that manual scheduling cannot be effectively used in expanding institutions and interferes with the teaching continuity.

These difficulties can be solved successfully with the help of scheduling system. These systems rely on algorithms to assign the classes depending on established criteria, including the availability of a teacher, classroom capacity, and subject demand (Kaur and Kaur, 2021). Moreover, an online scheduling system helps administrators to identify and resolve conflicts in real-

time and before a timetable can be confirmed. Notifications and real-time updates will enable teachers and students to be aware of any changes.

The proposed TMS will thus have a dynamic class scheduling feature which will eliminate conflict, and it will be easily edited. This aspect not only improves the efficiency of operations but also decreases the workload of the administration and establishes better coordination between teachers, students and administrators.

2.2.3 Traditional Payment Tracking and Financial Record Issues

Another aspect that tuition centres frequently find difficult to cope with is the financial management because they still use manual operations. The administrators usually document payments by use of paper receipt, logbook, or excel sheets, which are prone to mistake in data entry, duplication, and late update (Alam and Ahmed, 2020). These paper methods pose a challenge to ensure transparency and generate the correct financial reports. Payment records are not always complete and consistent, which is why conflicts between parents and administrators are likely to arise (Yusof et al., 2021).

In Pusat Tuisyen Sinar Wangsa, the administrators have reported cases of misplaced payment receipts and unconfirmed balances to be very frequent. Follow up of the overdue payments involved consulting several logbooks and manual reconciliation of the transactions a process that was not only time-consuming but also unreliable. The findings of the research by Asim et al. (2021) confirm that the implementation of automated billing systems can help increase financial responsibility and minimize human error within educational institutions.

The use of a digital fee management system offers the following benefits; automated generation of invoices, real-time payment updates and generation of reports without complications. This type of automation does not only guarantee financial accuracy but also builds trust between the tuition centre and parents since there are now easy to trace records of payment (Patil and Dhamale, 2022).

Thus, the proposed TMS will consist of an integrated billing and payment tracking system that will help administrators to keep track of payments sufficiently and create correct financial statements. The feature also keeps the financial data clean, ready and convenient to audit and make decisions.

2.3 Related Works

The literature review is meant to examine the current tuition and academic management systems that have led to the formulation of the proposed Tuition Management System (TMS). This section will determine the strengths and weaknesses of these systems by examining the existing technologies, structure of the systems, and implementation results so as to know the strengths and limitations of such systems to guide the development of a more integrated and user-friendly platform.

Three systems that are well established FACTS Tuition Management, Blackbaud Tuition Management and an openSIS (open Source Student Information System) were reviewed. All these systems serve particular administrative purposes like financial management, enrolment and academic scheduling. But there is none that offer a complete on-board solution in the case of small to medium-sized tuition centres. Each system is discussed as follows.

2.3.1 FACTS Tuition Management System

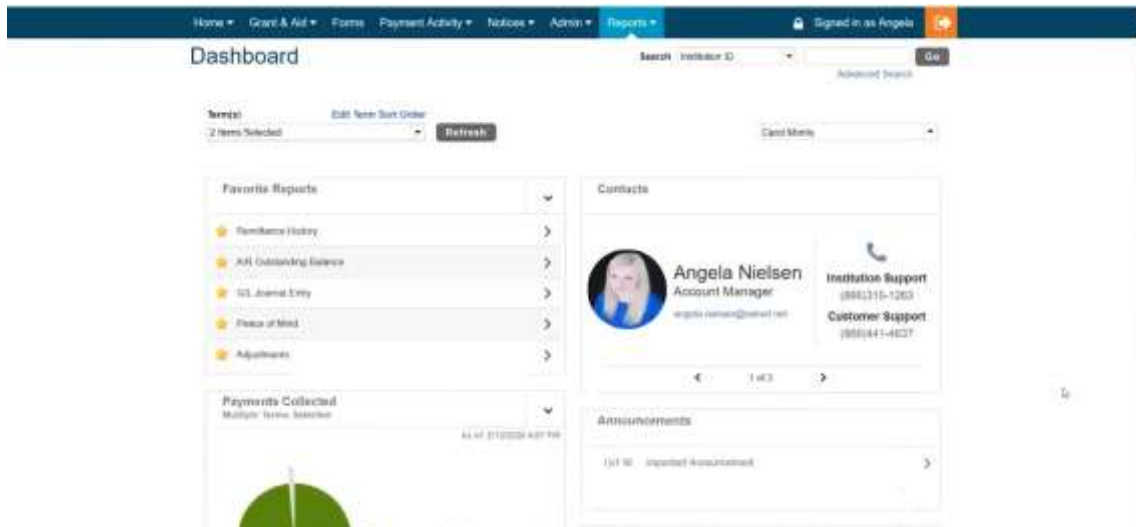


Figure 2. 1: FACTS Tuition Management System Dashboard

Agreement Number	Student Name	Student ID	Payment Method	Status	Remaining Amount Due	Remaining Credit Balance	Deductions	Unresolved Balance	Remaining Scheduled Payments
001 000 001	Adams, Abby	Adams, Abby	Credit Card	Active	1,000.00	0.00	0.00	0.00	0
001 000 002	Adams, Abby	Adams, Abby	Credit Card	Active	1,000.00	0.00	0.00	0.00	0
001 000 003	Adams, Abby	Adams, Abby	Credit Card	Completed	0.00	0.00	0.00	0.00	0
001 000 004	Adams, Abby	Adams, Abby	ACH	Active	1,000.00	0.00	0.00	0.00	0
001 000 005	Adams, Abby	Adams, Abby	ACH	Active	1,000.00	0.00	0.00	0.00	0
001 000 006	Adams, Abby	Adams, Abby	Credit Card	Active	1,000.00	0.00	1,125.00	0.00	0
001 000 007	Adams, Abby	Adams, Abby	ACH	Active	1,000.00	0.00	0.00	0.00	0
001 000 008	Adams, Abby	Adams, Abby	ACH	Active	1,000.00	0.00	0.00	0.00	0
001 000 009	Adams, Abby	Adams, Abby	Credit Card	Active	1,000.00	0.00	0.00	0.00	0

Figure 2. 2: FACTS Tuition Payment Activity

FACTS Tuition Management was a well-regarded software created by Nelnet, Inc. to automate the tuition payment system in learning institutions especially in the private schools and learning centres. It is aimed at streamlining the financial management, providing online billing, scheduling payments and automated reminders to parents and administrators (Nelnet, 2023).

FACTS system enables institutions to establish several payment options - full payment, monthly or semester payments - which provide parents with flexibility. The platform is connected to the online banking systems to enable the automatic deductions and transfer of funds electronically. FACTS is also useful in real time reporting and auditing where the administrators can access the history of transactions and come up with proper financial summaries.

Also, FACTS has other features like invoice management tool, a fee adjustment tool and a late payment notification tool, which considerably minimizes the administrative burden that comes with the manual billing. Schools can also design discounts, financial aid, or scholarships using the system itself (Adebayo & Ncube, 2021).

Strengths:

- High automation and transparency in financial processes.
- Flexible payment options to meet diverse parental preferences.
- Integrated notification and reporting system for financial tracking.
- Reduces administrative workload and ensures better accountability (Rahman et al., 2021).

Weaknesses:

FACTS has a weak academic management despite its robust financial management capabilities. It lacks student attendance, grading and class scheduling modules (Arias et al., 2021). That is why,

all the institutions that use the FACTS have to combine it with other learning management systems (LMS) in order to manage academic processes. This is why it is not as applicable to small tuition centres that need an all-in-one solution.

The advantage of FACTS in automating payment can be used as a reference on the fee management component of the TMS. Nonetheless, in comparison to FACTS, the suggested system will incorporate both academic and financial functionalities within a single platform and will not require third-party integrations.

2.3.2 Blackbaud Tuition Management System

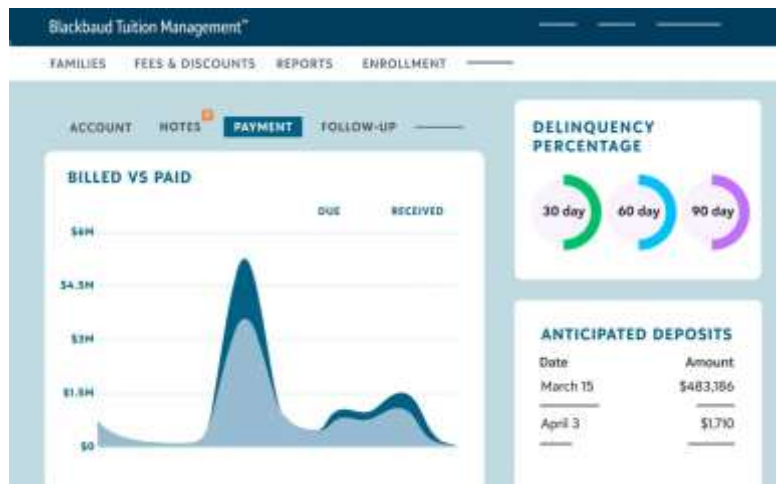


Figure 2. 3: Blackbaud Tuition Management



Figure 2. 4: Admin Portal

Blackbaud Tuition Management belongs to the Blackbaud K-12 Cloud Solutions Suite which is a complex system that assists schools to process tuition bills, student enrolment, and to communicate with families (Blackbaud, 2023). It aims at offering flexible financial operations and maintaining data protection and data integration with other educational systems.

Blackbaud system automates the whole billing process of generation of invoices and payment reconciliation. Parents are provided with the opportunity to use self-service portal to access their billing accounts and look at future bills and even pay them online using credit cards or direct bank transfers. It also provides automated email and SMS notifications about due dates, outstanding balances, and confirmation receipts (Patil and Dhamale, 2022).

Additionally, the system is connected with Blackbaud Student Information System (SIS) and blackbaud Learning Management System (LMS) to align student data, attendance, and grade data. This renders it scalable to large educational organizations that demand centralized information management (Gupta and Sharma, 2022).

Strengths:

- Fully automated and secure payment tracking system.
- Seamless integration with Blackbaud's suite of educational management tools.
- High-quality data analytics and financial reporting.
- Customizable user interface for different institution types (Sahu et al., 2020).

Weaknesses:

Though Blackbaud can be integrated to other Blackbaud products, it can be costly and resource-intensive, which is not readily available to small tuition centres with small budgets (Chavda & Solanki, 2021). Besides, the central orientation of the system is on financial processes, and, thus, academic modules (such as scheduling or attendance monitoring) are not fully developed without any integrations with other Blackbaud products.

The Blackbaud system offers useful information on creating an engaging and secure payment system. The suggested TMS is inspired by the data integration functionality of Blackbaud but is geared towards affordability and simplicity, enabling smaller tuition centres to operate all their business with one affordable and easy-to-use system.

2.3.3 openSIS (Open Source Student Information System)

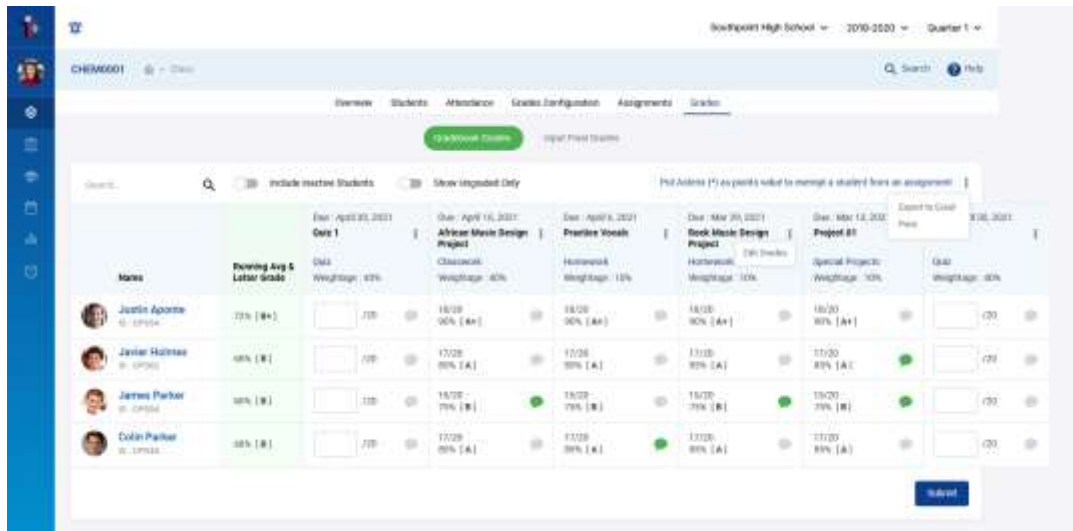


Figure 2. 5: openSIS System Grading

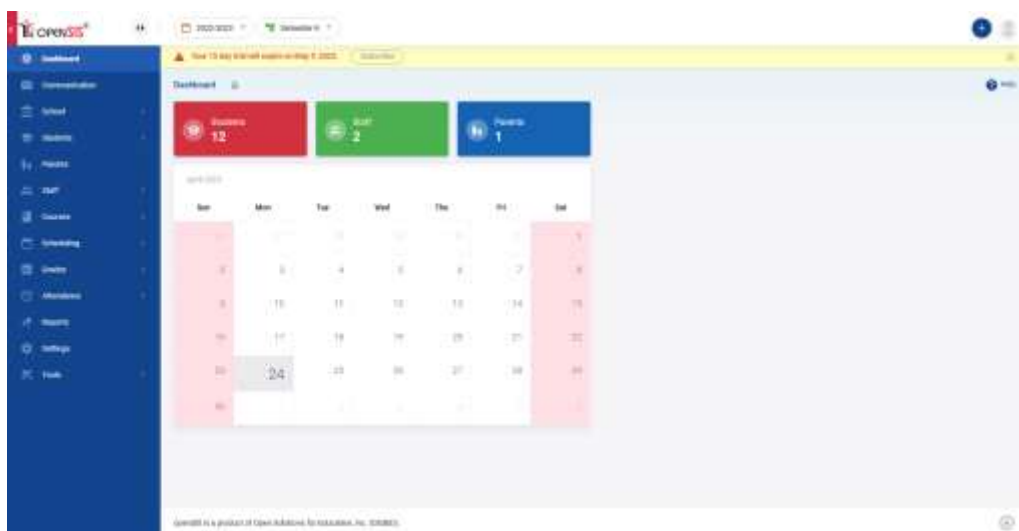


Figure 2. 6: openSIS Dashboard

OpenSIS is an open-source system aimed at dealing with school, college and training administrative and academic tasks. openSIS is developed by OS4ED and is aimed at offering basic academic services, including enrollment, attendance tracking, creating a timetable, and managing grades (OS4ED, 2023). It is open source, which means that it is free and can also be modified to fit the needs of the institution.

openSIS enables the institutions to store all the student information, such as demographics, attendance, academic achievements, and disciplinary data. It favors the use of role-based access control which guarantees different levels of access to the system by administrators, teachers, students and parents (Arias et al., 2021). The system also offers report generation features where schools can be able to create transcripts, attendance reports and performance summaries.

The other interesting aspect of openSIS is the class scheduling module, which automatically assigns teachers and classrooms depending on the availability and requirements of subjects (Kaur and Kaur,

2021). Nevertheless, openSIS is more advanced in academic management; however, it does not offer technical financial tracking and payment management (Nair and Saini, 2020).

Strengths:

- Open-source and customizable to fit specific institutional needs.
- Strong academic administration tools (attendance, grading, and scheduling).
- Supports multi-user access with role-based security controls.
- Generates comprehensive academic reports efficiently.

Weaknesses:

The system does not have any in-built financial management applications like fee tracking, invoicing or automated payment notification. It also demands technical skills to install, host, and maintain, which can be difficult when a small tuition centre has limited IT resources (Mohanty and Dash, 2022). openSIS offers inspiration to the academic component of the proposed Tuition Management System, especially in terms of running attendance and classroom scheduling. But in contrast to openSIS, the suggested TMS will merge academic and financial management into a single web-based application that will be much easier to operate and maintain.

2.4 Comparison

Having discussed three systems namely, FACTS Tuition Management, Blackbaud Tuition Management and openSIS it is apparent that each system has specific objectives as well as design strategies. FACTS and Blackbaud mainly focus on financial automation, whereas openSIS is concentrated on academic management. None of these systems however makes available a fully integrated platform that integrates both academic and financial features in a casual, low end and easily usable format that is meant to serve small to medium tuition centres. The following comparison shows their key features, strengths, and weaknesses.

Table 2. 1: Comparison of Existing Project

Feature/Criteria	FACTS TMS	Blackbaud TMS	openSIS	PTSW TMS
Main Focus	Automates tuition billing and payment tracking	Financial management with integration to student data	Academic management (scheduling, attendance, grading)	Integrated academic and financial management for tuition centres
Student Registration	Basic data storage	Integrated with SIS	Comprehensive student profiles	Online registration with secure database

Class Scheduling	Not available	Partial (via integration)	Automated timetable generation	Scheduling without conflicts
Attendance Tracking	Not included	Limited integration	Yes, real-time tracking	Yes, web-based attendance record and reports
Fee Management	Strong, automated billing and reminders	Advanced financial reporting and notifications	Not available	Payment tracking, and invoice generation
Reporting Tools	Financial reports only	Financial and analytics dashboards	Academic reports only	Financial and academic reports
Accessibility	Web-based (subscription)	Cloud-based (enterprise license)	Open-source, self-hosted	Web-based, responsive design for all devices
Cost	High (subscription)	High (enterprise license)	Free but needs technical skills	Affordable and easy to maintain
Customization	Vendor-controlled, limited	Moderate	High (open source)	High, designed for small institutions
Ease of Use	Simple for finance staff	Moderate (requires training)	Complex (requires IT knowledge)	Simple interface for admin and teachers
Ideal Users	Private schools, finance administrators	Large institutions with IT teams	Schools and colleges with technical staff	Tuition centres and teacher
Main Weakness	Lacks academic management	High cost, dependent on other systems	Lacks financial tools, requires technical setup	Currently under design

2.5 Discussion

Comparison of three of the existing systems FACTS Tuition Management, Blackbaud Tuition Management, and openSIS has revealed that all three have useful but partial features. FACTS and Blackbaud are primarily financial management solutions, and openSIS targets academic administration.

Nonetheless, none of them offers a wholesome and low-cost solution that consolidates both academic and financial undertakings of small tuition centres.

FACTS Tuition Management is also a good company that automates billing, payment tracking, and detailed financial reports (Adebayo and Ncube, 2021). It assists institutions to minimize the number of mistakes in manuals and enhance transparency in financial records. Irrespective of these advantages, FACTS does not have the necessary academic functionalities, including attendance management, grading, and schedule management (Arias et al., 2021). In the case of tuition centres, it would still require another system to manage student records and class records, and this will lead to increased work and costs.

Accordingly, Blackbaud Tuition Management will be a financial automation continuation of the other cloud-based services of the company, providing a wide range of payment options, automatic reminders, and reporting tools (Blackbaud, 2023). Although this is suitable in large schools with a dedicated IT staff, it is not feasible in small tuition centres due to its high price and complicated setup (Rahman et al., 2021). Academic functions are only available to institutions by purchasing additional modules and this makes the system resource intensive and inappropriate to centres with limited budgets.

Contrarily, openSIS offers a wide variety of academic management solutions, including student registration, attendance tracking, timetable scheduling, and performance reporting all in a central database (OS4ED, 2023). It is open source whereby institutions can make changes. Nevertheless, a payment or billing module is absent in the system, and administrators have to handle financial records manually or with the help of third-party apps (Nair & Saini, 2020). Furthermore, openSIS needs maintenance and installation skills that might be hard on smaller companies that lack IT skills (Mohanty and Dash, 2022).

The Tuition Management System (TMS) is a solution to these shortcomings with all the advantages of the three systems incorporated into a single, web-based system. Similar to FACTS and Blackbaud, it automates the payment tracking, invoice generation, and financial reporting. Simultaneously, it is openSIS-inspired, with open-source academic capabilities like web-based registration, no-show-scrutiny, and automatic schedule creation. The integration also guarantees that administrators will handle both academic and financial records under the same system thus saving the use of many platforms.

Moreover, the proposed TMS will be rather affordable and user-friendly, created with the usage of the popular web technologies like HTML, CSS, JavaScript, PHP (Laravel) or Node.js and MySQL. It may be hosted locally or online providing flexibility to tuition centres without spending a lot of money on subscription fee. Such ease of use and availability make it applicable to any institution, such as Pusat Tuisyen Sinar Wangsa that has a limited budget and staff but needs effective digital management.

To sum up, it has been seen during the discussion that FACTS and Blackbaud are both partial solutions to finance and academics respectively and they do not offer the full integration that tuition centres require. The proposed TMS will have a combination of academic and financial functions into one user-friendly system which will facilitate proper data management, eliminating any possible conflict of schedules, automated billing and transparency. It thus is a moderate, low cost and holistic solution that bridges the gap that has been established in the comparison.

2.6 Conclusion

In summary, this chapter has been able to review the past literature, existing systems and research work on development of the Tuition Management System (TMS). The research has revealed that tuition centres continue to encounter issues like manual records keeping, conflict in class schedules and poor tracking of payment. The analysis of the existing FACTS, Blackbaud, and openSIS systems revealed that all the systems have handy functionality yet have apparent limitations. FACTS and blackbaud perform well in the area of finance management but lacking in academic functions, and openSIS is good in academic records, but it lacks payment features. Hence, the suggested TMS will aim at integrating academic and financial characteristics into a single system. It will be easy, cheap and acceptable to small tuition centres. The results in this chapter are the basis and guidelines in the system design and development that will be discussed in the following chapter.

3 METHODOLOGY

3.1 Introduction

This chapter discusses the procedures and steps taken to come up with the Tuition Management System (TMS). This methodology is meant to make sure that there is a systematic, efficient and development of the system based on the needs of the user as analysed during the investigation phase.

The need to have a clear methodology that helps in steering every step of development, mitigate risks as well as ensure that the system achieves its intended objectives (Pressman and Maxim, 2020). It even enables a developer to utilize the resources well and create a system that is functional, easy to use, and stable. This project has a structured approach that makes sure all the activities undertaken during planning and implementation will be in line with the system objectives.

3.2 Agile Methodology



Figure 3. 1: Agile Methodology Diagram (Marketing91, n.d.)

In this project, the system development life cycle (SDLC) with agile will be used as a methodology. The SDLC offers a systematic approach to the creation of information systems, which is divided into phases known as requirement analysis, design, development, testing, deployment and review (Sommerville, 2020). It makes sure that development is carried out in a systematic and rational sequence that enhances quality and the accuracy of the system.

Nevertheless, Agile approach is an improvement on the SDLC as it brings in flexibility and adaptability. Agile enables the development to occur in small, iterative cycles called sprints instead of waiting to have the whole system developed and then getting feedback on the development (Beck et al., 2001). Individual modules or features are being covered with each sprint reviewed and enhanced according to the user feedback.

This SDLC and Agile combination is the best to apply to Tuition Management System as it enables the continuous improvement, accelerated delivery, and enhanced communication with users throughout the project (Highsmith, 2020). The strategy guarantees that the system is efficient in adapting to changes and needs of the users.

3.3 Phases in Agile Methodology

The System development life cycle (SDLC) integrated with the agile approach has six key phases namely requirement analysis, system design, system development, testing, system deployment and system review. The significance of each of the phases is to make sure that the Tuition Management System is constructed in a structured and efficient way. Agile concept is implemented in the whole cycles and makes it possible to get continuous feedback and improvement upon completion of every sprint (Beck et al., 2001; Pressman and Maxim, 2020).

3.3.1 Requirement Analysis

This stage will involve collecting data with stakeholders of the tuition-centre, including administrators and teachers. The primary goal is to be aware of the existing issues and decide what the system should accomplish. Based on observation and informal interviews, some problems were observed such as manual maintenance of student records, timetable clash in classes, and the time-consuming nature of payment tracking. Based on them, both functional and non-functional requirements have been identified. Some of the functional requirements are student registration, attendance, classes, and payment. The non-functional requirements are the usability, reliability and security requirements. By stating such requirements in writing, the development team will be able to make sure that each feature meets the expectations of the users.

3.3.2 System Design

The design phase will convert the user requirements into a full system blueprint. This phase entails coming up with the system architecture, database design and user interface (UI) design. The database is projected based on MySQL with the specified entities students, teachers, subjects, classes, and payments being connected by primary and foreign keys to ensure data integrity. Simultaneously, the UI design is provided with the help of such tools as Figma to see how customers will use the system. Regarding the interface, there are registration, scheduling, and fee-tracking screens illustrated in interface mock-up. A well-designed design assists developers and users to have the same understanding prior to the coding process and makes the system user-friendly and visually consistent.

3.3.3 System Development

This stage entails the development of the Tuition management system. Front end is designed in HTML, CSS and JavaScript to design responsive and interactive pages and the back end is designed in Java. MySQL is used in the implementation of the database to manage data effectively. The Agile iterative process of development entails the development of features in very small segments or sprints. As an illustration, the initial sprint can be concerned with registering the students, the second term can be concerned with scheduling the classes, and the third term can be concerned with the fees management. System is tested after every sprint and refined according to the feedback of the users. This method will enable the rapid delivery, debugging and constant enhancement of the development.

3.3.4 Testing

Testing is required to ensure that the system operates as expected and according to the quality standards as stipulated by the requirements. It is run on an ongoing basis throughout every sprint and not just at the conclusion of development.

A number of different types of testing are conducted:

- Functional Testing: Ensures that the functionality, including registration and payment tracking, is correct.
- Usability Testing: Ensure the interface is not complex, ambiguous or too complicated to use by all of the user roles (administrators, teachers).
- Security Testing: Makes sure that the sensitive information including personal information and transactions is well-protected against unauthorized access.

Multiple testing cycles help to identify errors at an early stage and minimize failures of the system after its deployment.

3.3.5 Deployment

After going through all the test phases, the system is implemented into the real world. Deployment refers to the installation of the application on a local server or cloud hosting application in order to have end-users accessing the application over the Internet. Training of the users is also done at this stage so that the administrators and teachers will be able to use the system effectively. It is first launched on a pilot basis, and any form of feedback is first obtained by the users which will assist in making the necessary small adjustments before the release. This is important as this is a step that is undertaken to leave the system in the development environment to the real working environment in Pusat Tuisyen Sinar Wangsa.

3.3.6 Review

The review phase substitutes the conventional maintenance phase in order to focus on constant assessment and enhancement instead of just remedying problems. During this stage, feedback will be collected continuously by the users like administrators and teachers, post deployment. Their remarks are evaluated to find out the areas, which the system may be enhanced, or some main features may be added. Review process makes the system remain relevant, efficient and user centred. It can be refining reports, better scheduling algorithms or better security features depending on user experience. Occasionally scheduled review meetings allow the development team to monitor performance and make plans towards later versions.

3.4 Conclusion

In conclusion, this chapter has talked about the methodology involved in the development of the Tuition Management System (TMS). The project will implement the System Development Life Cycle (SDLC) coupled with the Agile approach that balances between structured planning and flexibility in development. This combination approach makes sure that the system is created in a structured way and yet can be changed and improved according to the feedback of the users.

All stages of the methodology, requirement analysis to review are significant steps in steering the success of the project. The user needs were determined during the requirement analysis phase to make sure that the features of the system actually address the issues the tuition centres have been experiencing which include manual record keeping and poor payment tracking. These requirements were then converted into explicit system models and database designs during the design phase to guarantee logical arrangement and the user-friendly interfaces.

The development stage involved actual coding of the system and Agile iterations were applied where new modules (registration, scheduling, attendance and payment) could be developed, tested and enhanced in steps. Constant testing during every sprint also allowed to detect and address some mistakes early and created a more reliable system. The deployment also presented the system to actual people under a controlled environment giving the opportunity to collect feedback before the entire rollout. Lastly, the review phase will facilitate continuous enhancement by conducting frequent assessments and user feedback to ensure that the system can improve the systems needs in line with the requirements of the tuition centre.

In general, the given methodology gives a clear path when it comes to the creation of an effective, convenient, and efficient Tuition Management System. When the SDLC process is combined with the flexibility of Agile, the process of development will be more user oriented and adaptive. This will make sure

that the end system does not just serve the technical requirements, but it also brings real value to the administrators, teachers and students to facilitate easier operations and management in the tuition centre.

4 REQUIREMENTS

4.1 Introduction

This chapter shows the specifications in the development of Tuition Management System (TMS). One of the most crucial phases of software development is the requirements stage as it dictates what the system is supposed to accomplish and how it is going to work in fulfilling the needs of the users (Sommerville, 2020). It guarantees that the end system meets its purpose of ensuring the real issues of the staff and students at the tuition center are solved. Completely defining requirements will ensure that the development team does not make any assumptions that may lead to an unknown situation, minimize errors in the design process, and that the system will be efficient (Pressman and Maxim, 2020).

In this project, the system requirements will be classified into three, namely functional requirements, non-functional requirements, and system requirements. Functional requirements outline tasks that the system should execute, including handling the student records, classes, attendance, and payments. The performance, usability, and security standards are defined by non-functional requirements, and the hardware and software required in the process of development are determined by the system requirements. In order to collect these requirements, two key methods were employed, observation and questionnaire, in order to obtain real feedback of users, and examine the existing manual processes. These approaches made sure that the final requirements are based on the reality of needs and expectations of users at the tuition center (Dennis et al., 2021).

4.2 Data Gathering Techniques

In order to gather correct and valid requirements to develop the Tuition Management System (TMS), two primary methods of data collection were utilized research of previous projects and survey. The reason behind the selection of these techniques is that they do not only give the technical feedback but also the user-based feedback, the system is constructed following the best practices and operational requirements.

4.2.1 Study of Existing Project

Existing projects were studied by analysing the existing tuition management systems and other platforms of this sort to learn their characteristics, work patterns and the structure of the systems. This method served to find out the merits and flaws of the current systems in terms of student registration, schedule of classes, attendance tracking, and fee tracking. The developer learned about good user interface design,

the ways to manage the data and the automations by examining the work of other systems. The results of this research influenced the development of better features and prevented the possible pitfalls of design that are often found in other systems.

4.2.2 Questionnaire

Direct feedback in the form of questionnaires was collected using the questionnaires, which were distributed to the specific users, in this case, administrators and teachers, who are primarily the people in charge of handling student data, class schedules and tracking payments at the tuition centre. The questionnaire was created to gather data on the existing issues in their day to day running, the preferences of the user on features of the system and the expectations of the new system. This approach gave important feedback on the practical needs of the users, allowing to see which processes needed to be automated and which features could enhance efficiency and accuracy of management tasks.

The questionnaires are a good method of gathering information about numerous respondents in an efficient way, particularly in a project that involves a variety of users. In this research, the answers of administrators and teachers were summarized, and the analysis was undertaken to identify the most demanded system functions. This was then employed to establish and rank the functional requirements of the Tuition Management System (TMS) so that the end design can match user needs and expectations.

4.3 Functional Requirement

Functional requirements explain the particular behaviours, actions or activities that the system is to execute. They are founded on the needs that were determined in the process of collecting data and how the Tuition Management System will work.

Table 4. 1: Functional Requirement of Administrator

Function	Result
Sign up	Admin will sign up to the system by creating a user account
Log in	Admin will be able to access the system by logging in
Student Registration	The system must allow administrators to add, update, and delete student profiles, including personal and contact details.
Teacher Management	The system must enable administrators to manage teacher information and assign them to subjects or classes.
Class Scheduling	The system must allow administrators to create and edit class timetables, ensuring no conflicts between teachers, rooms, or time slots.

Financial Tracking	The system must allow administrators to generate invoices, record payments, and track outstanding balances.
Report	The system must generate reports for student attendance, student performance, and class schedules for administrative review.

Table 4. 2: Functional Requirement of Teacher

Function	Result
Sign up	Teacher will sign up to the system by creating a user account
Log in	Teacher will be able to access the system by logging in
Class Scheduling	The system will allow teacher to see class timetables, rooms and time slot.
Attendance Tracking	The system must allow teachers to record and update student attendance for each class.
Report Generation	The system allow teacher to update student performance and student attendance for admin review.

4.4 Non-Function Requirement

Non-functional requirements describe the quality attributes and operational standards of the system. They determine how well the system performs its tasks rather than what it does.

Table 4. 3: Non-Function Requirement

Non-Functional	Description
Usability	The interface must be user-friendly and easy to navigate for users with basic computer knowledge.
Performance	The system should respond to user requests (such as login, data retrieval, and saving) within 3 seconds.
Security	All user data, including passwords and payment records, must be stored securely using encryption.
Availability	The system should be available at least 99% of the time when hosted online.
Reliability	The system must perform its operations consistently without data loss or unexpected errors

Privacy	The system must comply with privacy principles by restricting unauthorized access to personal information.
---------	--

4.5 System Requirement

System requirements define the hardware and software requirements that are required to develop and run the Tuition Management System (TMS) successfully. Such needs are focused on making the system operate effectively, efficiently and be able to accommodate all the necessary functions. System requirements will be in two categories including hardware requirements and software requirements.

4.5.1 Software requirements

I. Database - MySQL



Figure 4. 1: MySQL

MySQL was chosen as the database management system due to its reliability, scalability and because it is easy to integrate with web applications that use Java. It has been able to effectively handle relational data including student records, payment information and attendance data.

II. Programming Language - Java, JavaScript, HTML, CSS



Figure 4. 2: CSS, HTML, JavaScript



Figure 4. 3: Java

Java is employed in the backend logic because of its security, stability and robust object-oriented design. The front-end interface is created on JavaScript, HTML, and CSS and allows the creation of responsive and interactive web pages to both the administration and teachers.

III. Framework – Bootstrap, Spring Boot



Figure 4. 4: Bootstrap



Figure 4. 5: Spring Boot

The spring boot framework streamlines the process of the backend development and offers powerful web security, REST API and data management features. Bootstrap can be used in responsive web design, meaning that the interface scales well across the various devices, like desktops, tablets, and smartphones.

IV. IDE - Visual Studio Code, IntelliJ



Figure 4. 6: Visual Studio Code



Figure 4. 7: IntelliJ

They were selected since these integrated development environments are flexible, offer smart code support, and support Java, HTML, and CSS. IntelliJ is efficient in terms of Java development, and Visual Studio Code offers a good front-end development environment.

4.5.2 Hardware Requirements

Table 4. 4: Hardware Requirements

Processor	AMD Ryzen 5 4600H 3.00GHz
RAM	8GB 3200MHz
GPU	GTX1650 4GB
Storage	512GB
Operating System	Windows 11

4.6 Conclusion

In conclusion the critical requirements to design the Tuition Management System (TMS) were presented in this chapter. Both technical and user-oriented requirements were determined through the systematic data-gathering methodological tools like the study of existing projects, and the questionnaires. Such techniques have been used in assuring that the system design is precise to the actual requirements of administrators and teachers especially in matters like student management, allocation of classes, attendance, and records related to payments. The requirements were divided into functional, non-functional and system requirements to give the requirements a clear structure to develop. Non-functional requirements outline

the fundamental operations of the system whereas functional requirements warrant that the system operates safely, reliably, and efficiently including protection of privacy of data used by the users. The hardware and software requirements specify the technical environment that is required to be implemented and perform efficiently. These factors combined form a solid basis to the design and development stages of this system whereby the TMS will bring efficiency, higher accuracy of data, and user satisfaction in the tuition centre.

5 ANALYSIS

5.1 Introduction

It is a chapter that deals with the analysis stage of a Tuition Management System (TMS). System analysis entails the perception and analysis of the problems, processes, and requirements that are already in existence or have been compiled previously to identify the manner in which the new system is to operate. Sommerville (2020) discusses that system analysis is supposed to convert user requirements into a systematic model, which recognises the system behaviour, processes and data flow.

During this stage, Data Gathering Analysis, Use Case Models, Flowcharts and Business Process Model and Notation (BPMN) diagrams are used to analyse functions and interaction between components of the system. They are analytical tools that assist in visualizing the way the system works, the flow of data through various processes and interaction of the system with users like administrators and teachers. The finding of this analysis is used as the basis of designing and implementing a system in the subsequent phases.

5.2 Data Gathering Analysis

5.2.1 Questionnaire Analysis

i. Teachers

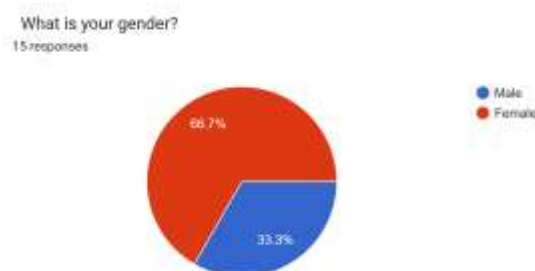


Figure 5. 1: Question 1

According to the answers, 66.7 percent of the teacher population is female, and 33.3 per cent are male. Here, it is apparent that majority of the teaching staffs are women.

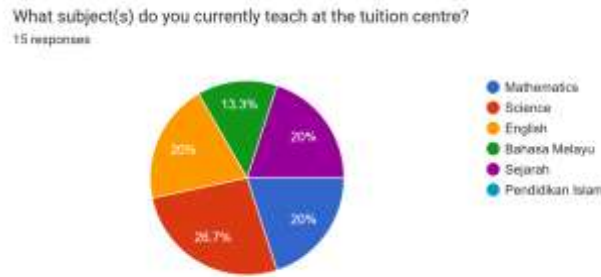


Figure 5. 2: Question 2

The statistics indicate that 26.7 percent of teachers teach in science, and 20 percent of teachers are in Mathematics, English, and Sejarah. The percentage of Bahasa Melayu is 13.3 and there is no teacher teaching Pendidikan Islam. This implies that Science is the most widely taught subject in the centre.

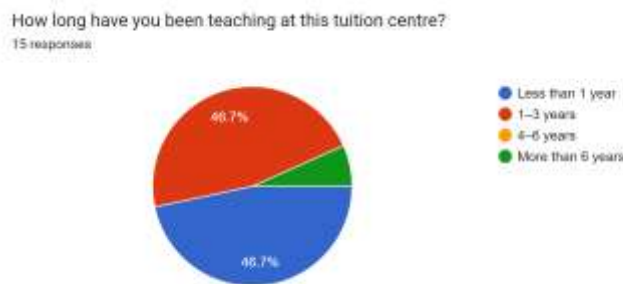


Figure 5. 3: Question 3

A total of 46.7% of teachers have been teaching for less than one year, another 46.7% have between one to three years of experience, and only 6.6% have more than six years. This means most teachers are still relatively new to the centre.

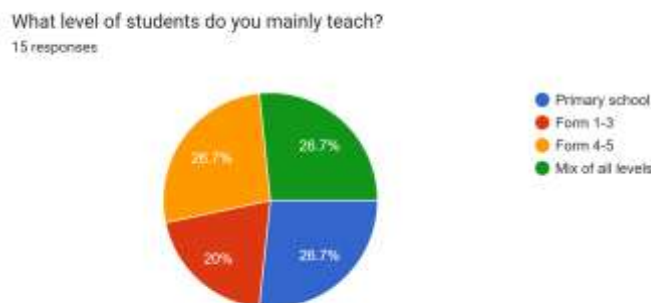


Figure 5. 4: Question 4

The responses are quite balanced, with 26.7% teaching primary school students, 20% teaching Form 1–3, 26.7% teaching Form 4–5, and another 26.7% teaching mixed levels. This shows that teachers handle a wide range of student levels.

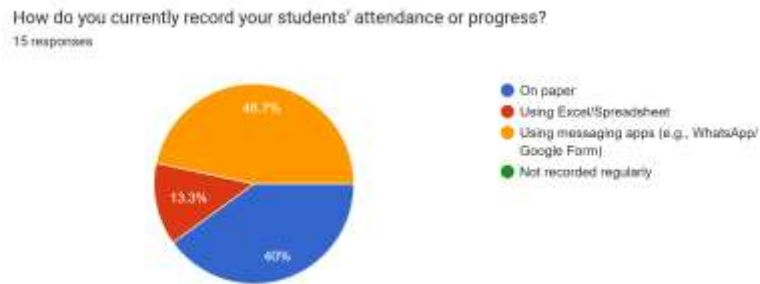


Figure 5. 5: Question 5

Most teachers record attendance and progress using WhatsApp or Google Forms (46.7%), followed by paper (40%) and Excel (13.3%). This shows that most teachers still use manual or informal methods instead of a proper system.

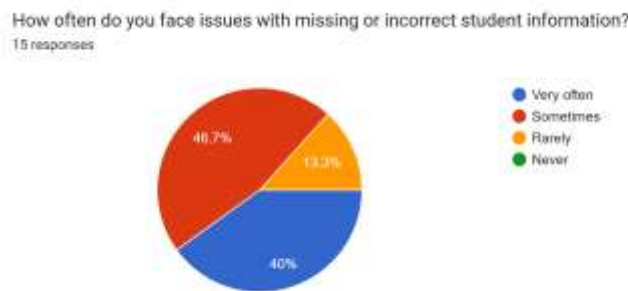


Figure 5. 6: Question 6

About 40% of teachers face this problem very often, 46.7% sometimes, and 13.3% rarely. This means that almost all teachers experience issues with inaccurate or missing data at some point.

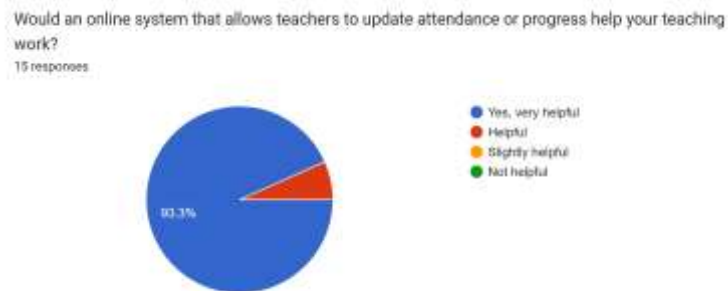


Figure 5. 7: Question 7

Nearly all teachers think that such a system would be helpful, 93.3% of them think that it would help a lot and 6.7% think that it would help. This is a good indication of the need to adopt an online system.

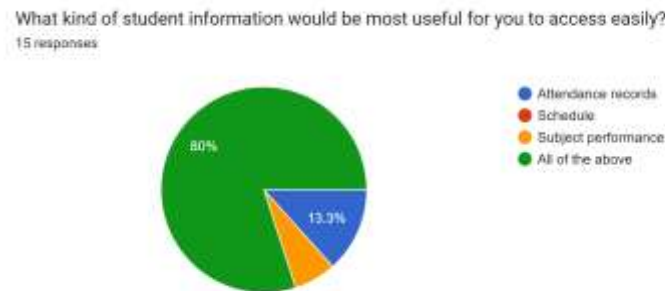


Figure 5. 8: Question 8

Majority of teachers (80) would like to have access to any form of information such as attendance records, schedules and student performance. This implies that we need a fully integrated system.



Figure 5. 9: Question 9

It has been revealed that 60 percent of teachers use printed timetables and 40 percent use WhatsApp groups or shared files. This implies that it is still manual in terms of scheduling and it can be enhanced digitally.

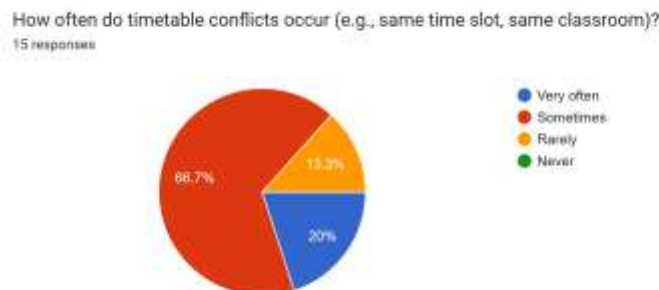


Figure 5. 10: Question 10

Twenty percent of teachers report frequent conflicts in their timetables (very often), 66.7% report type of conflicts rarely (sometimes), and 13.3% rarely (never). This demonstrates that scheduling problems are very common.

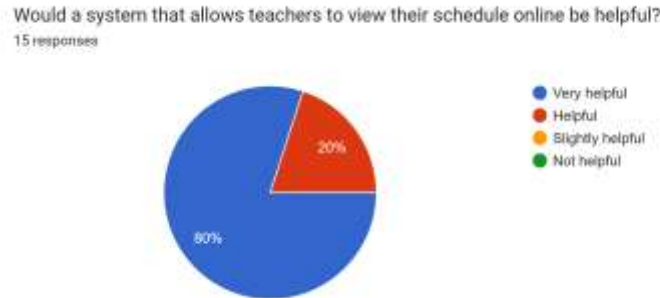


Figure 5. 11: Question 11

Majority of the teachers (80) reported that it will be very helpful and 20% reported that it will be helpful. This demonstrates the necessity of an online scheduling feature.

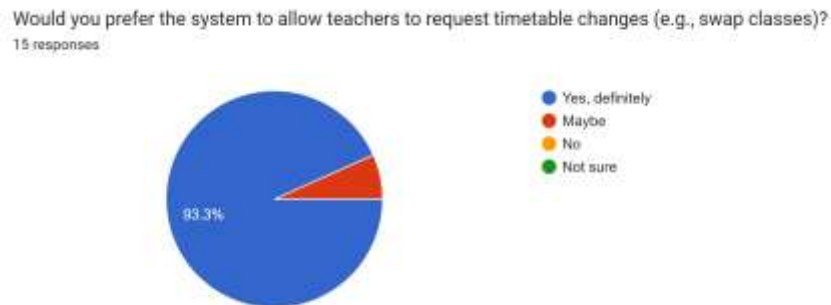


Figure 5. 12: Question 12

A big proportion (93.3) of the respondents responded yes, definitely, and 6.7% responded may be. This indicates that instructors desire flexibility to make or demand adjustments of their schedules with ease.

ii. Admin

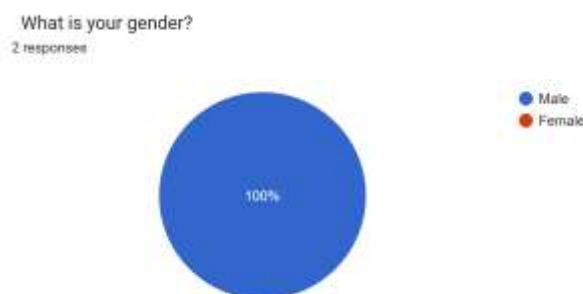


Figure 5. 13: Question 1

Both participants are males implying that 100% of the administrative personnel that took part in this survey are men.

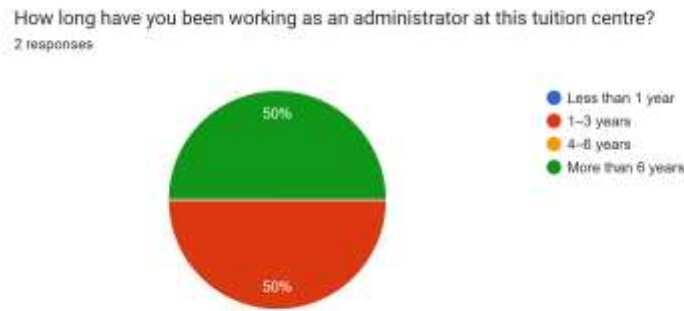


Figure 5. 14: Question 2

Fifty percent (50) of the respondents have a working life experience of 1-3 years, and the other half (50) have a work experience of over six years. This demonstrates the presence of newer and experienced administrators in the centre.

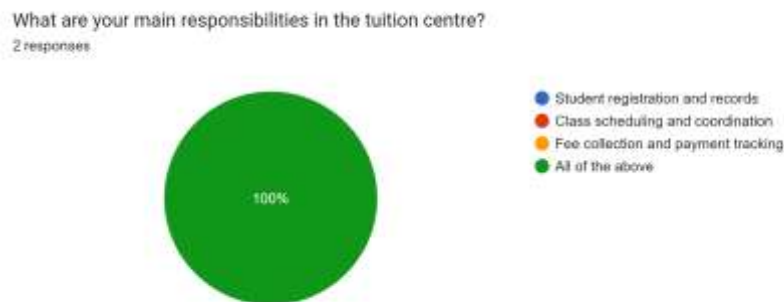


Figure 5. 15: Question 3

A hundred percent (100) of the respondents chose all of the above, which comprised of student registration and records, class scheduling and coordination, and fee collection and payment tracking. This demonstrates that administrators deal with several tasks and must have a system that will enable them cope with all tasks effectively.



Figure 5. 16: Question 4

The overall student population under the management of both respondents (100 percent) comprises over 200 students, meaning that both administrators have to deal with a vast number of student files and a computerized system capable of processing large amounts of data would be beneficial to them.

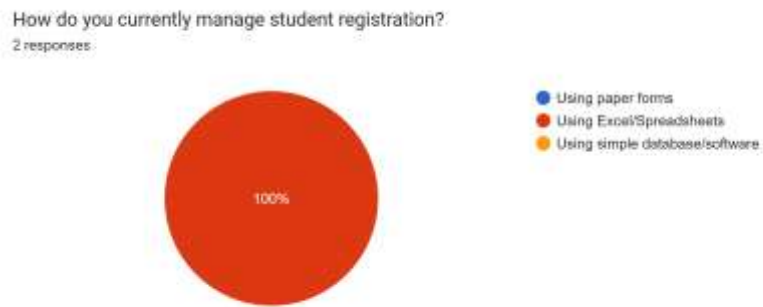


Figure 5. 17: Question 5

Student registration is done by all the administrators (100 percent) with Excel or spreadsheets. This indicates that the process is manual and can be time-consuming or can be subject to human error.

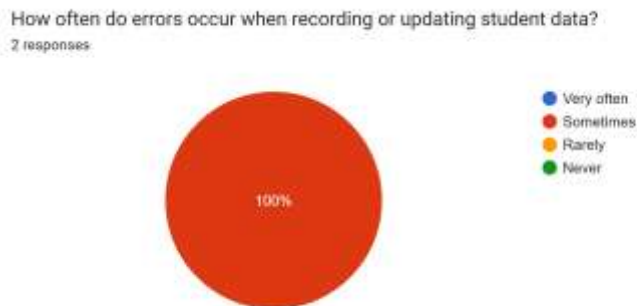


Figure 5. 18: Question 6

Both respondents (100%) responded to sometimes, which means that they make errors occasionally when updating the student information. This shows the necessity of a more precise and automated system.

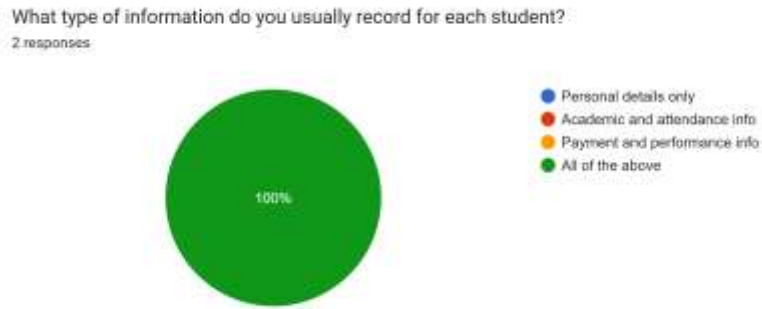


Figure 5. 19: Question 7

All administrators (100 percent) document all the information about the students such as personal information, academic information, attendance, payment information, and the performance information. This implies that the system must be capable of handling various types of data and that they are well organised.

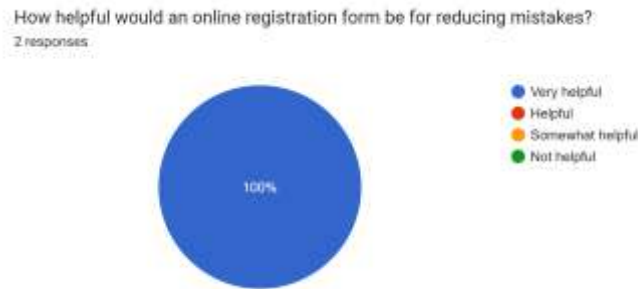


Figure 5. 20: Question 8

All the respondents (100 percent) indicated that online registration form would come in handy. This demonstrates a high level of support of digital registration to minimize manual error to enhance data accuracy.

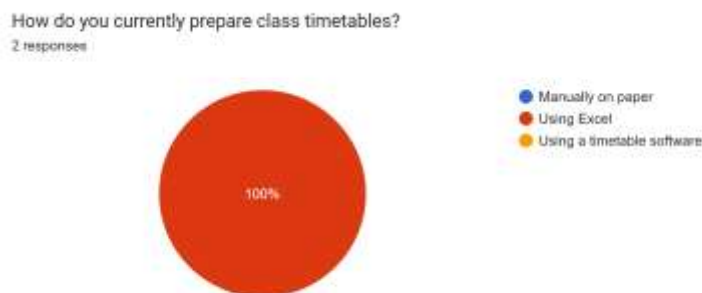


Figure 5. 21: Question 9

Both administrators (100%) are involved in the preparation of class timetables manually using a timetable template. This indicates that maybe the process can be automated to save time and reduce the scheduling errors.

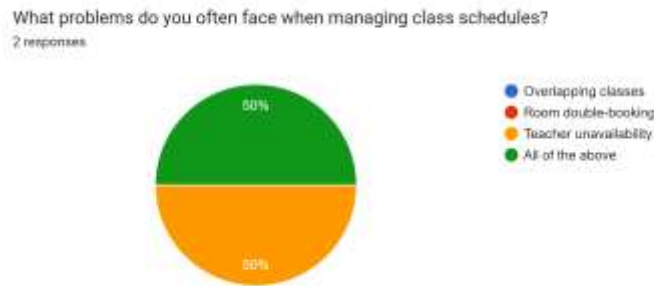


Figure 5. 22: Question 10

Exceptionally 50% of the respondents said that one of their problems is teacher unavailability and the other 50 percent said that they have all the issues listed like overlapping classes and room booking. This shows that scheduling is a common problem and requires more effective management tools.



Figure 5. 23: Question 11

The entire administrator population (100 percent) was in support of such a feature. This demonstrates that both teachers and administrators would have a connected system in which they can view schedules and ensure their accuracy online.

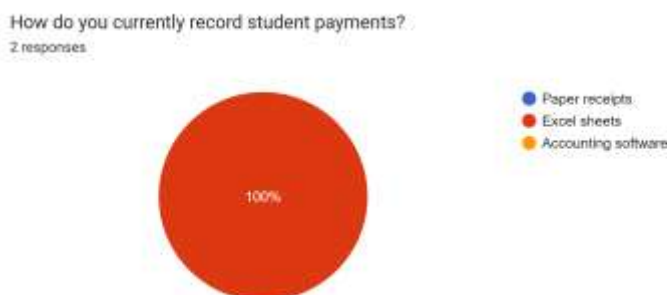


Figure 5. 24: Question 12

Administrators (100%) use Excel sheets to track the payments indicating that payment tracking is still manually done and can be likely to miss some data.



Figure 5. 25: Question 13

The problem of the absence of records is encountered by all respondents (100%). This establishes that payment tracking using manual methods is ineffective and inefficient, thus the necessity to use an automated payment system.

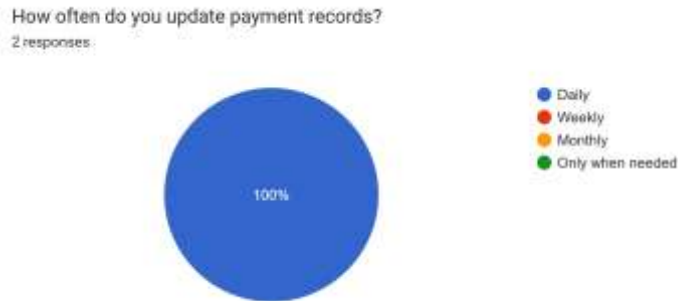


Figure 5. 26: Question 14

Both administrators (100 percent) reconcile payment records on a daily basis. It will imply that regular updates are frequent, but with manual processes, it is hard to ensure full accuracy and consistency over time.

5.3 Use Case Model

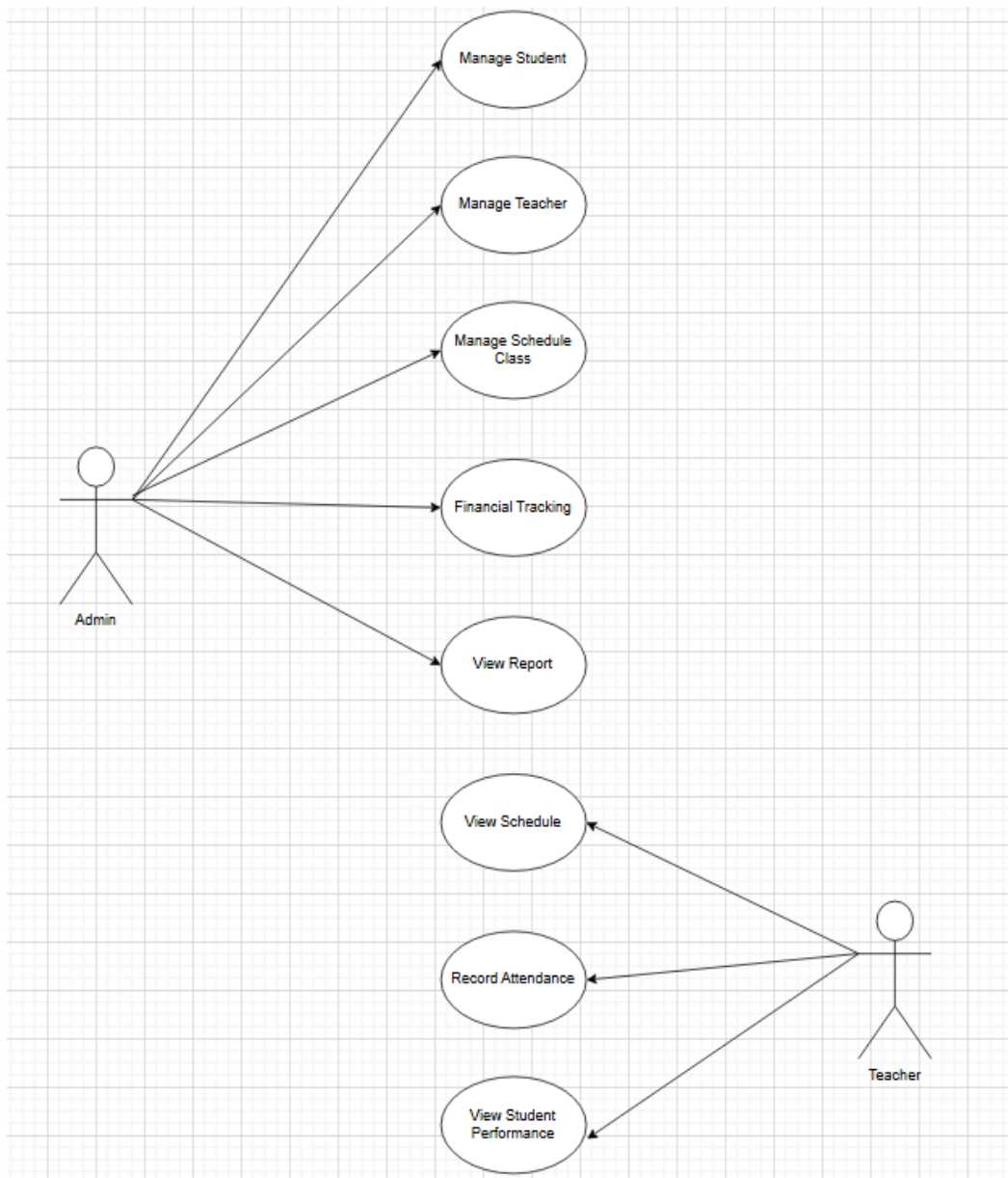


Figure 5. 27: Use Case Model

The use case model highlights that the two actors of Admin and Teacher require the system to serve various purposes. The Teacher is concerned with classroom and performance data, whereas the admin is concerned with management and administrative control. Both share common data using the system and there is a free flow of information and manual work is minimized. The model assists developers and other interested parties to comprehend the magnitude of user interrelationship, data relationship, and functional context of the system, which is a clear basis later in design and implementation.

5.4 BPMN (Business Process Modelling Notation)

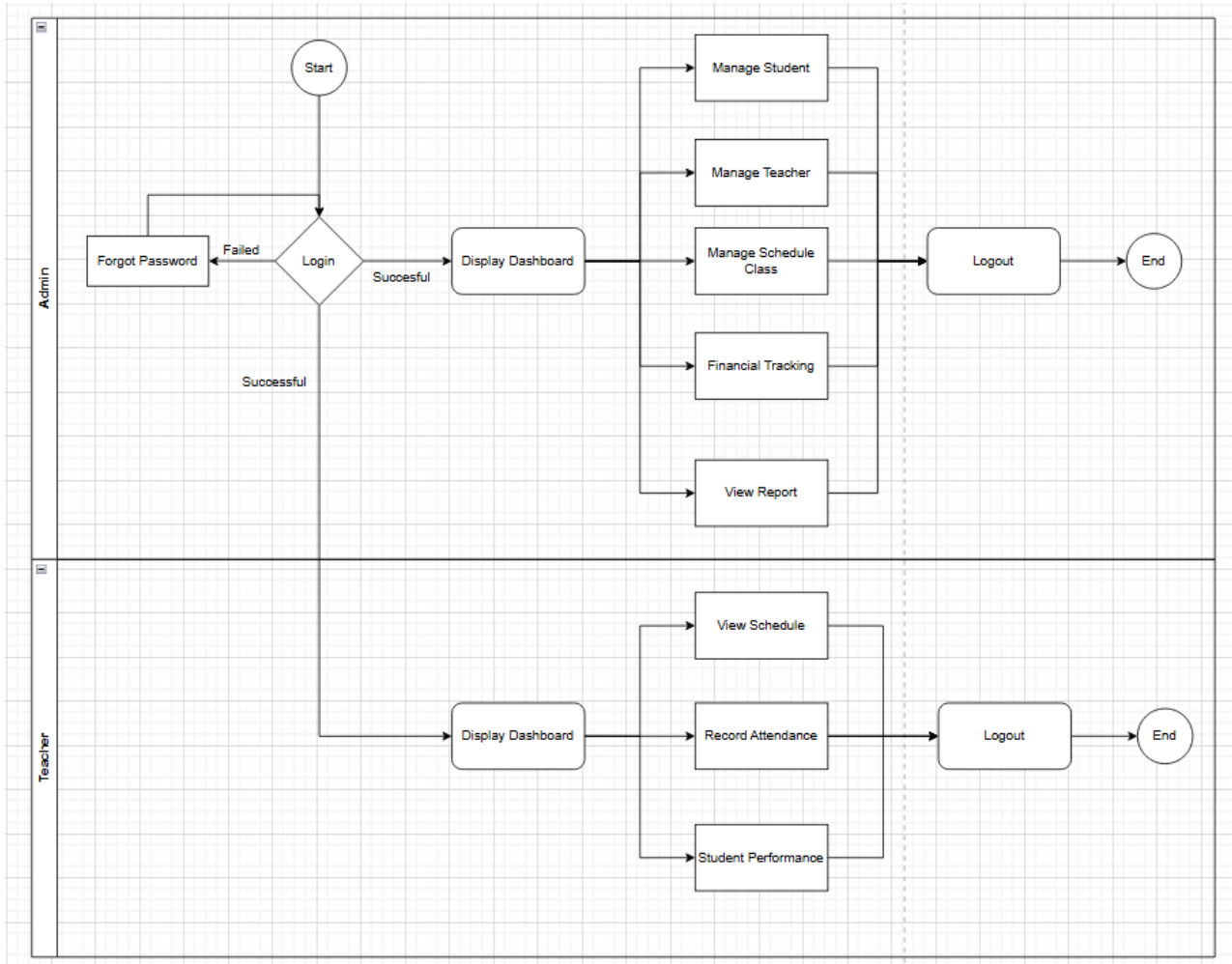


Figure 5. 28: BPMN

The above flowchart shows the flow of the processes of the Admin and Teacher in the Tuition Management System (TMS) with reference to the overall process. The first step is the stage of login where users are required to key in valid credentials so as to gain access to the system. In case the login fails, the user can select the Forgot Password option, which will allow recovery in case of success, they will be redirected to the main dashboard. To the Admin, the dashboard will offer some of the most important functions such as student and teacher management, scheduling classes, monitoring financial processes and system reports. These can enable the administrator to manage all the administrative and financial works within a single system. Once they have finished with their activities, the admin logs out to bring to an end the session safely.

At the same time, the Teacher also logs into the system and is redirected to his/her personal dashboard. The Teacher can use it to look at their schedule, take student attendance, and make updates to student performance records. The functions assist teachers in handling their academic duties better and lessening the paperwork. After finishing their work, the Teacher logs out so that the information could be secure. On balance, the flowchart is a clear indication of how the two user roles are in contact with the

system as there is easy navigation, assured access as well as good management of operations of the tuition centre.

5.6 Conclusion

In conclusion, this chapter provided the system analysis of the Tuition Management System (TMS) in terms of how various users communicate with the system and how the system processes the information. The Use Case Model has been able to identify two major actors, namely the Administrator and the Teacher and the various roles and functions they play in the system. This assisted in defining the scope of users, the tasks of the system, and also the connection among different processes like student management, attendance recording, and financial tracking.

The BPMN process also presented an understandable and systematic illustration of the workflow between work role Administrator and Teacher. It also depicted the interdependence between the actions of each user on the basis of the automated system operation like the validation of the logins, data storage, and the generation of reports. The visualization of these interactions allowed the BPMN to reveal possible process enhancements and make sure that the system design helps to facilitate a clear and effective operation.

This chapter generally gives a good groundwork towards the next phase of development, as per the analysis. It makes sure that the structure of the system is clearly understood, the requirements are properly captured and the processes are well linked to serve the objective of the Tuition Management System.

6 DESIGN

6.1 Introduction

This chapter presents the design phase of the proposed Tuition Management System (TMS). The design stage plays a crucial role in translating the system requirements and analysis into a structured blueprint that guides the system development process. It focuses on defining how the system will look, how it will function, and how different components will interact with each other (Sommerville, 2020).

In this chapter, the overall system architecture, user interface design, database design, and system workflow are discussed in detail. The design aims to ensure that the system is user-friendly, efficient, secure, and capable of meeting both functional and non-functional requirements identified in the previous chapters. A well-planned design helps reduce development risks, minimize errors during implementation, and improve system maintainability (Pressman & Maxim, 2020).

The chapter covers several key design aspects, including interface design to enhance user experience, database design to ensure data integrity and consistency, security framework to protect sensitive information, and system flow to illustrate how users interact with the system. These design components provide a solid foundation for the implementation phase, which will be discussed in the next chapter (Dennis et al., 2021).

6.2 Interface Design

6.2.1 Log in Page

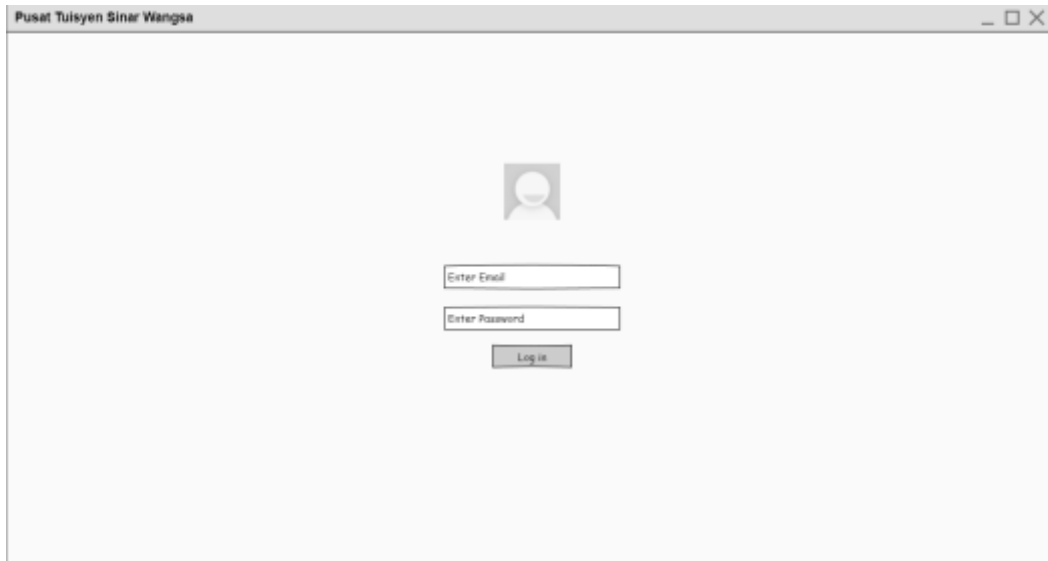


Figure 6. 1: Login Page

Figure 6.1 shows the login interface of the Tuition Management System (TMS). This page allows authorized users, including administrators and teachers, to access the system by entering their username and password. The login function ensures that only valid users can enter the system, providing basic security and user authentication before accessing system features.

6.2.2 Admin – Dashboard



Figure 6. 2: Admin Dashboard

Figure 6.2 illustrates the admin dashboard, which serves as the main control panel for the administrator. From this dashboard, the admin can navigate to different system modules such as student management, teacher management, schedule management, payment records, and reports. It provides a centralized view of system activities and quick access to important functions.

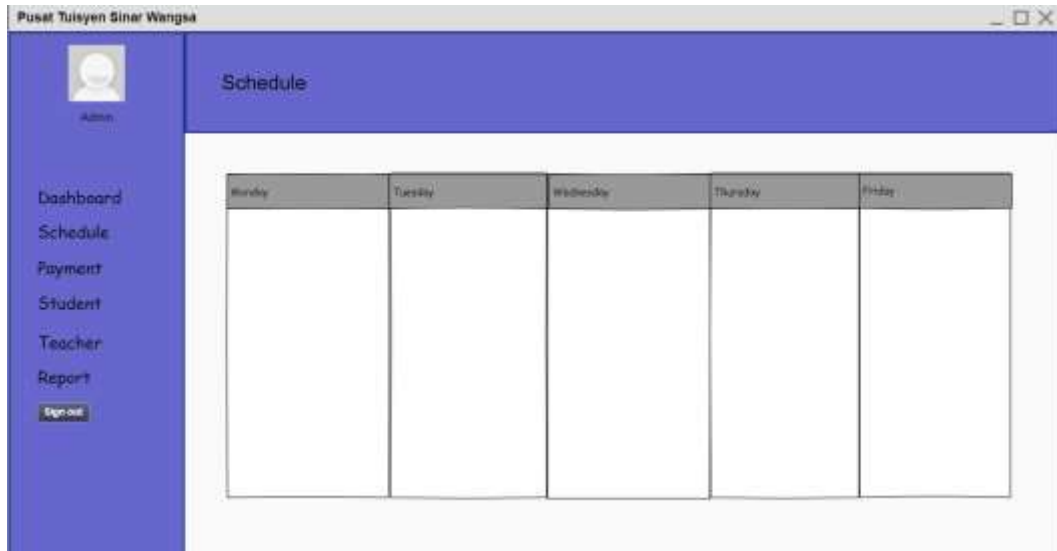


Figure 6. 3: Admin Schedule

Figure 6.3 displays the schedule management interface used by the administrator. This page allows the admin to create, edit, and manage tuition class schedules. The administrator can assign teachers, set class dates and times, and ensure that there are no scheduling conflicts.

6.2.4 Admin – Payment

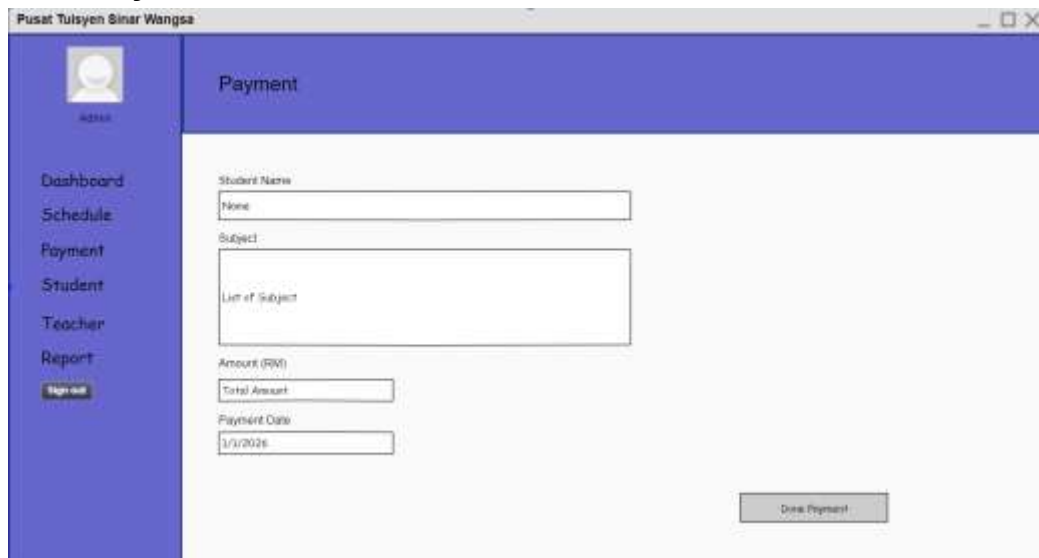


Figure 6. 4: Admin Payment

Figure 6.4 presents the payment management interface for the administrator. Through this page, the admin can record tuition fee payments made by students. The system stores payment information such as student name, payment date, and payment status for monitoring financial records.

6.2.5 Admin – Student

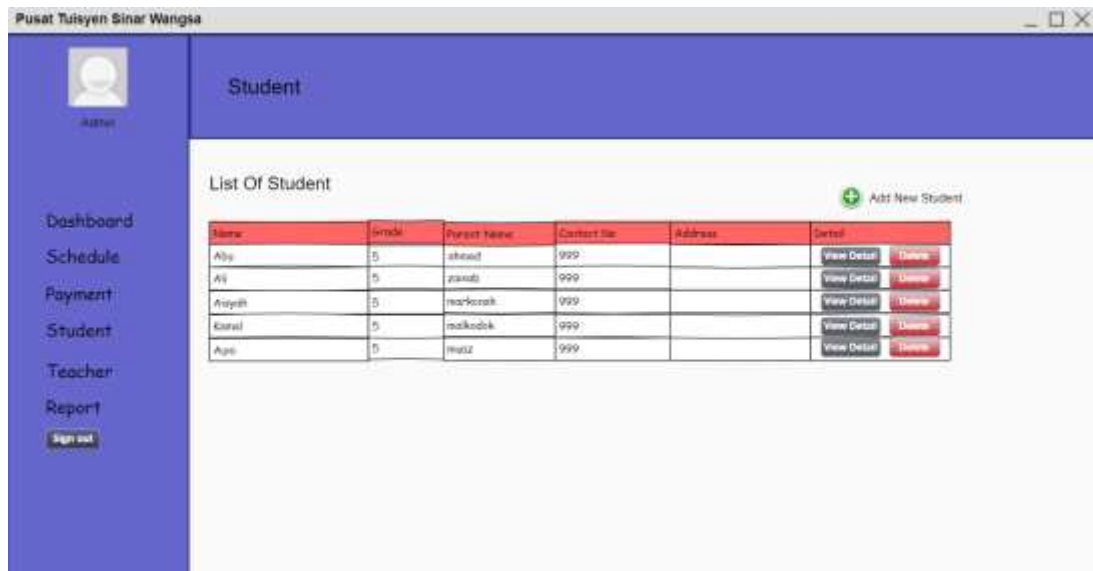


Figure 6. 5: Admin Student

Figure 6.5 shows the student management interface. This page enables the administrator to add, update, or delete student information. The system stores details such as student name, contact information, address, and enrolled subject.

6.2.6 Admin – Teacher



Figure 6. 6: Admin Teacher

Figure 6.6 illustrates the teacher management page. The administrator can manage teacher information including name, contact details, subject specialization, and other related data. This helps maintain organized records of teachers in the tuition center.

6.2.7 Admin – Report

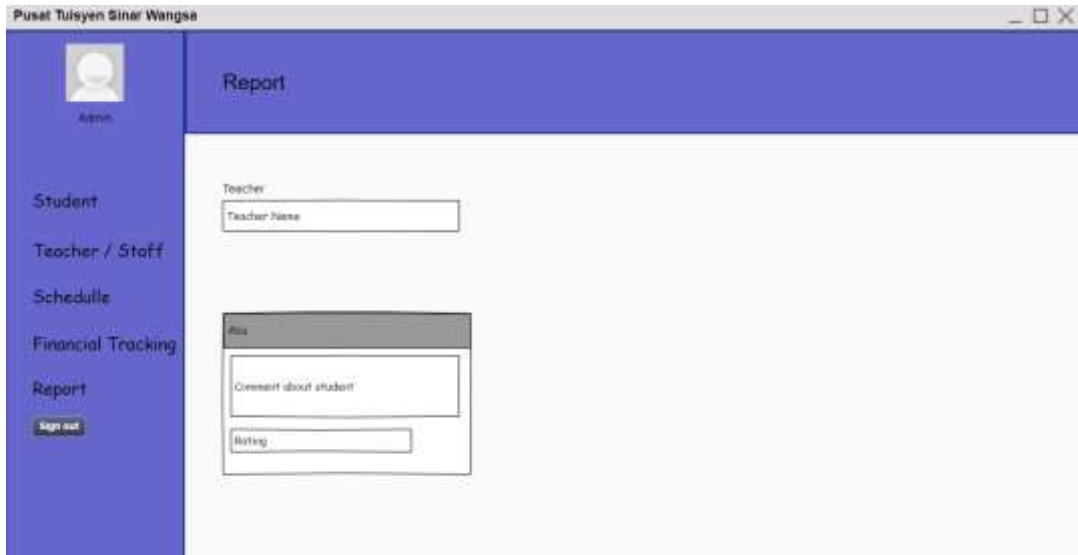


Figure 6. 7: Admin Report

Figure 6.7 presents the reporting interface for the administrator. This feature allows the admin to generate reports related to students' performance. These reports help in monitoring student activities and analysing student performance.

6.2.8 Teacher – Schedule

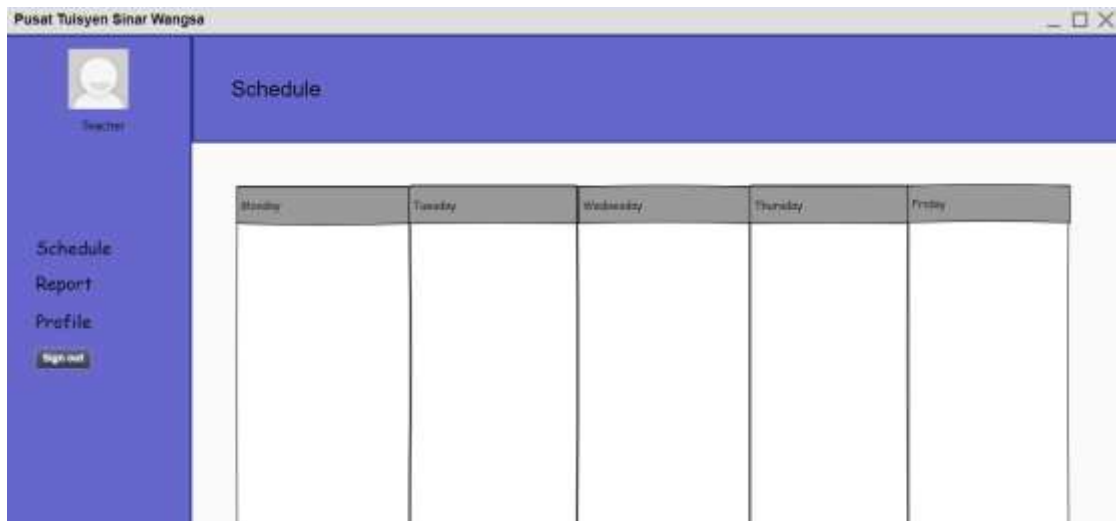


Figure 6. 8: Teacher Schedule

Figure 6.8 shows the schedule page available to teachers. Teachers can view their assigned teaching schedules, including class time, date, and subject. This allows teachers to easily manage their teaching timetable.

6.2.9 Teacher – Report

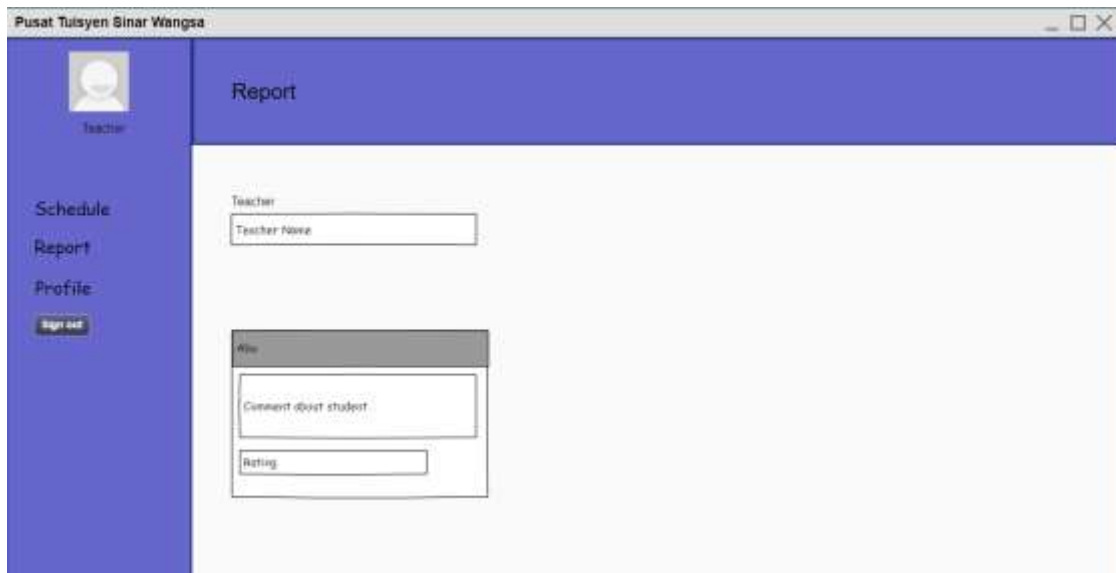


Figure 6. 9: Teacher Report

Figure 6.9 displays the report interface for teachers. Teachers can record or view student performance reports based on their class activities. This helps track student progress and learning outcomes.

6.2.10 Teacher – Profile

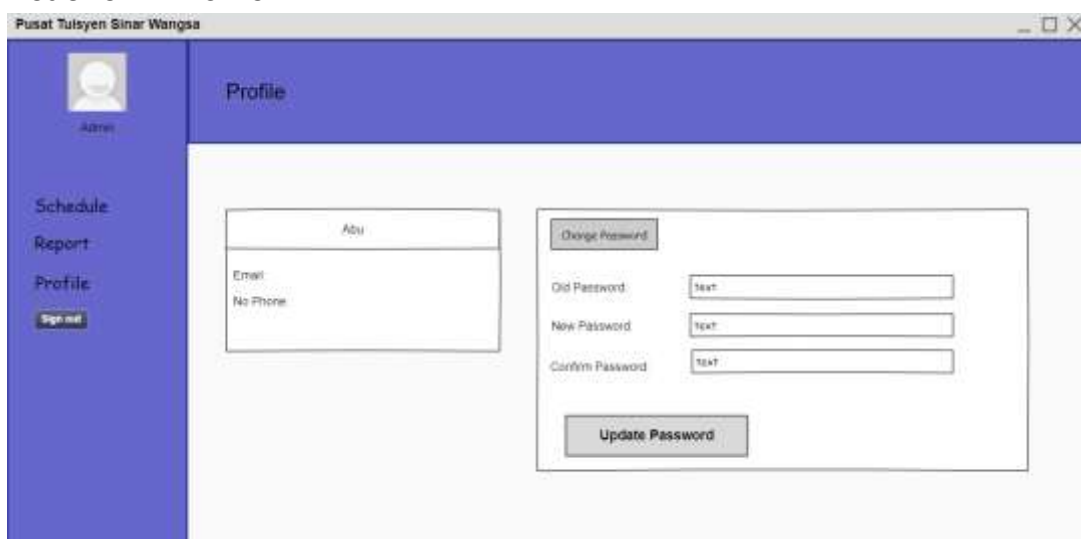


Figure 6. 10: Teacher Profile

Figure 6.10 illustrates the teacher profile page. This page allows teachers to view and update their personal information such as name, email, and contact details.

6.3 Database Design

6.3.1 Data Dictionary

A data dictionary is a structured reference that describes all data elements used in a system's database, including their names, data types, sizes, constraints, and meanings. In the Tuition Management System (TMS), the data dictionary provides clear definitions for each table and attribute

such as student records, class information, attendance, and payment details. This documentation ensures consistency, accuracy, and a shared understanding of the database structure among developers and stakeholders, while also helping to reduce data redundancy and errors. Additionally, the data dictionary supports easier system maintenance and future enhancements by serving as a reliable guide to the database design.

User Database

	A-Z Field	A-Z Type	A-Z Null	A-Z Key	A-Z Default	A-Z Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	email	varchar(255)	NO	UNI	[NULL]	
3	password	varchar(255)	NO		[NULL]	

Figure 6. 11: User Database

Figure 6.11 shows the structure of the user database table. This table stores user login credentials and related information required for authentication in the system.

Admin Database

	A-Z Field	A-Z Type	A-Z Null	A-Z Key	A-Z Default	A-Z Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	user_id	bigint	NO	UNI	[NULL]	

Figure 6. 12: Admin Database

Figure 6.12 presents the admin database table which stores information related to system administrators. This includes admin identification details used to manage system access and responsibilities.

Teacher Database

	A-Z Field	A-Z Type	A-Z Null	A-Z Key	A-Z Default	A-Z Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	user_id	bigint	NO	UNI	[NULL]	
3	address	varchar(255)	NO		[NULL]	
4	name	varchar(255)	NO		[NULL]	
5	phone_number	varchar(255)	NO		[NULL]	
6	created_at	datetime(6)	YES		[NULL]	

Figure 6. 13: Teacher Database

Figure 6.13 illustrates the teacher database table. This table stores teacher information such as name, email, contact details, and subject specialization.

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	address	varchar(255)	NO		[NULL]	
3	contact	varchar(255)	NO		[NULL]	
4	form_or_standard	varchar(255)	NO		[NULL]	
5	name	varchar(255)	NO		[NULL]	
6	parents_name	varchar(255)	NO		[NULL]	
7	created_at	datetime(6)	YES		[NULL]	

Figure 6. 14: Student Database

Figure 6.14 shows the student database structure. This table contains student-related information including student name, contact details, address, and enrolled subject.

Subject Database

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	teacher_id	bigint	NO	UNI	[NULL]	
3	name_subject	varchar(255)	NO		[NULL]	

Figure 6. 15: Subject Database

Figure 6.15 presents the subject database table. This table stores information about subjects offered by the tuition center, which are later linked to teachers and students.

Schedule Database

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	day	varchar(255)	YES		[NULL]	
3	end_time	varchar(255)	YES		[NULL]	
4	start_time	varchar(255)	YES		[NULL]	
5	subject_id	bigint	YES	MUL	[NULL]	

Figure 6. 16: Schedule Database

Figure 6.16 shows the schedule database table. This table stores class scheduling information including date, time, subject, and assigned teacher.

Payment Database

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	student_id	bigint	NO	MUL	[NULL]	
3	date	varchar(255)	NO		[NULL]	
4	status	varchar(255)	NO		[NULL]	
5	amount	decimal(38,0)	NO		[NULL]	

Figure 6. 17: Payment Database

Figure 6.17 illustrates the payment database table. It records payment transactions made by students, including payment date, amount, and related student information.

Teach Database

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	star	int	YES		[NULL]	
2	id	bigint	NO	PRI	[NULL]	auto_increment
3	student_id	bigint	NO	MUL	[NULL]	
4	teacher_id	bigint	NO	MUL	[NULL]	
5	comment	varchar(255)	YES		[NULL]	

Figure 6. 18: Teach Database

Figure 6.18 displays the teach database table. This table represents the relationship between teachers and subjects, indicating which teacher is responsible for teaching a particular subject.

Student_pay Database

	AZ Field	AZ Type	AZ Null	AZ Key	AZ Default	AZ Extra
1	id	bigint	NO	PRI	[NULL]	auto_increment
2	payment_id	bigint	NO	MUL	[NULL]	
3	student_id	bigint	NO	MUL	[NULL]	

Figure 6. 19: Student_pay Database

Figure 6.19 shows the student payment database table. This table records payment details associated with each student to track tuition fee payments.

6.3.2 Data Flow Diagram (DFD)

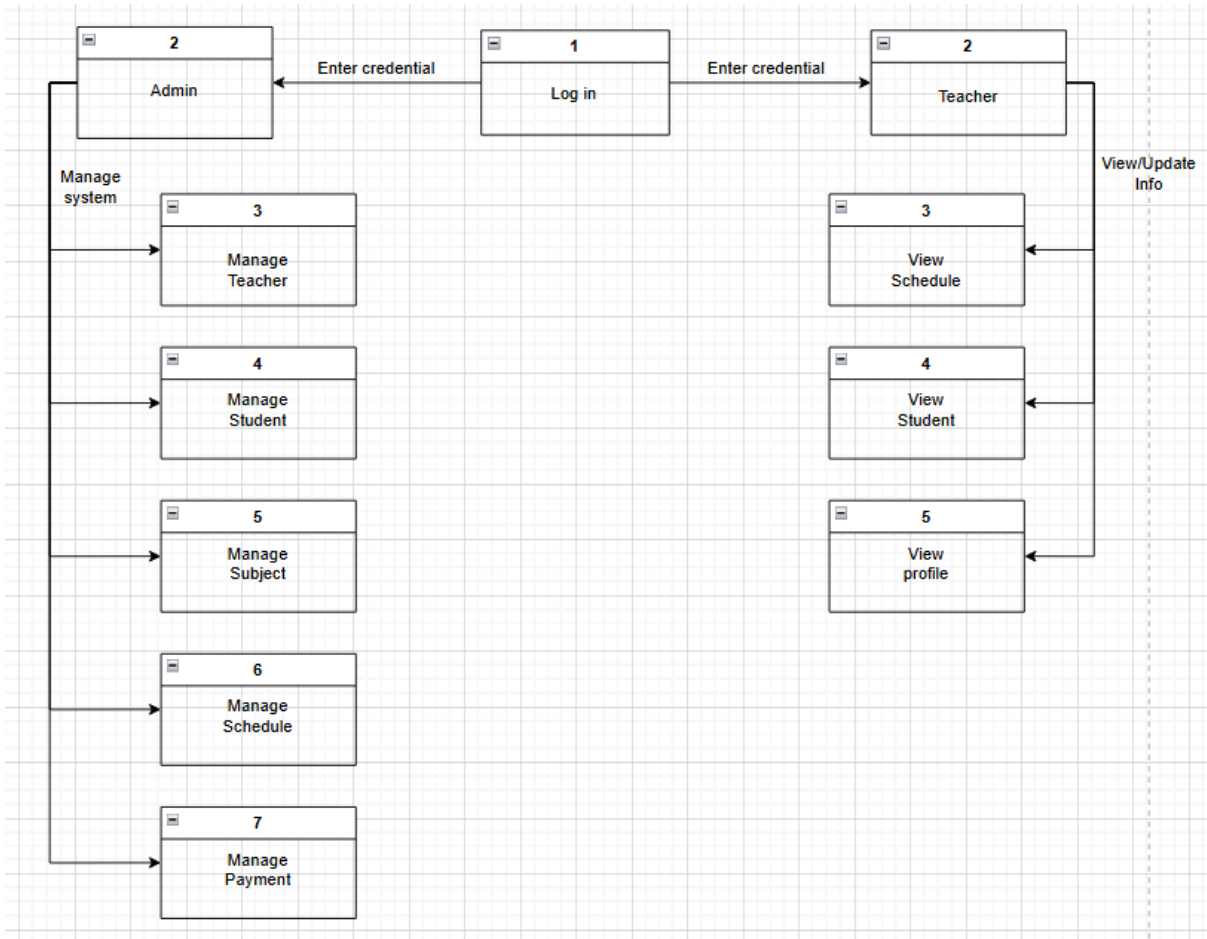


Figure 6. 20: Data Flow Diagram

Figure 6.20 illustrates the Data Flow Diagram (DFD) of the Tuition Management System (TMS). The diagram shows how data moves between users, processes, and the system database. In this system, the main external entities are the administrator and teacher, who interact with the system to manage various tuition-related activities. The administrator can manage student information, teacher records, class schedules, and payment data, while teachers can view their schedules and record student performance. These activities are processed by the system and stored in the database for future reference. The DFD helps visualize how data is processed and transferred within the system, ensuring that information flows efficiently between different system components.

6.3.3 Entity Relational Diagram (ERD)

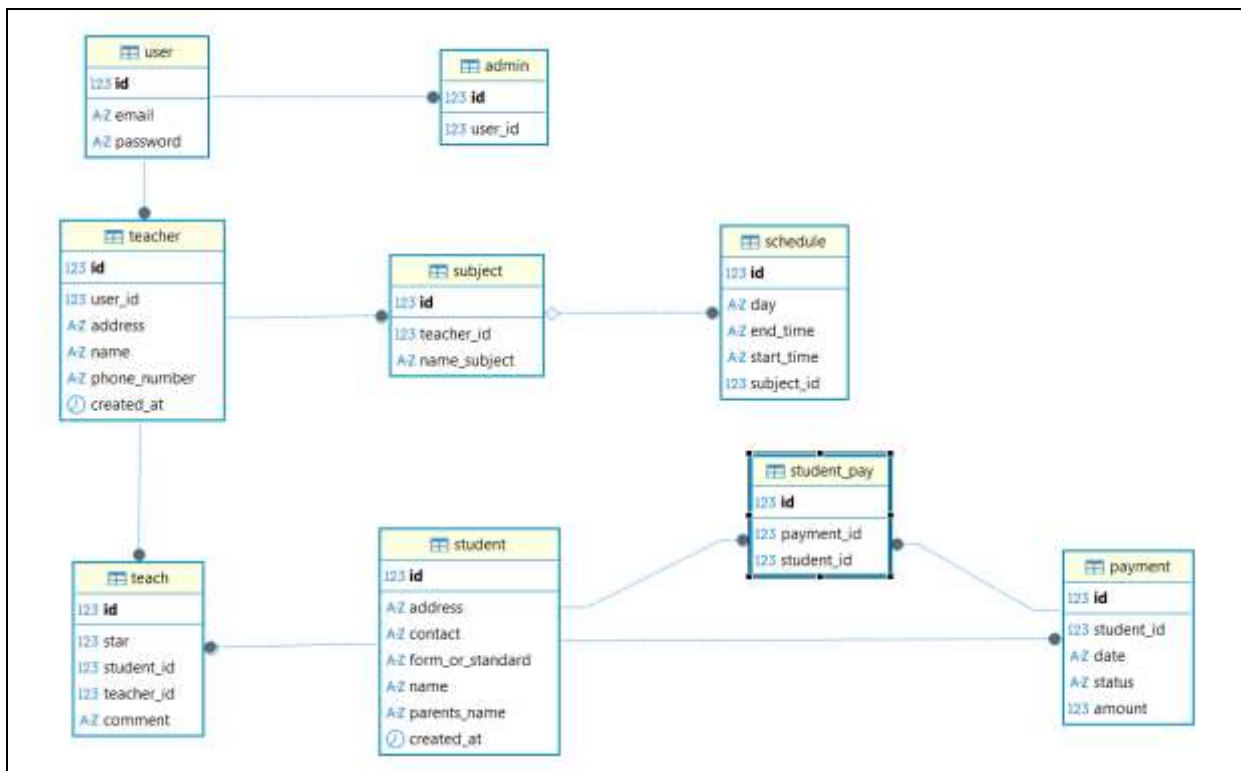


Figure 6. 21: Entity Relational Diagram

Figure 6.21 shows the Entity Relationship Diagram (ERD) of the Tuition Management System (TMS). The ERD illustrates the structure of the system database by presenting the entities and the relationships between them. In this system, the main entities include User, Admin, Teacher, Student, Subject, Schedule, Payment, Teach, and Student_Pay. Each entity represents a table in the database that stores specific information related to the system. For example, the Student entity stores student details, while the Teacher entity stores teacher information and the Subject entity represents the subjects offered in the tuition center.

The relationships between these entities allow the system to manage tuition activities efficiently. For instance, teachers are linked to subjects through the Teach entity, which shows which teacher teaches a particular subject. The Schedule entity stores information about class schedules, while the Payment and Student_Pay entities record student payment transactions. These relationships ensure that data is organized properly and allow the system to retrieve information easily when needed. Overall, the ERD helps provide a clear structure of how data is stored and connected within the Tuition Management System.

6.4 Flow of the System

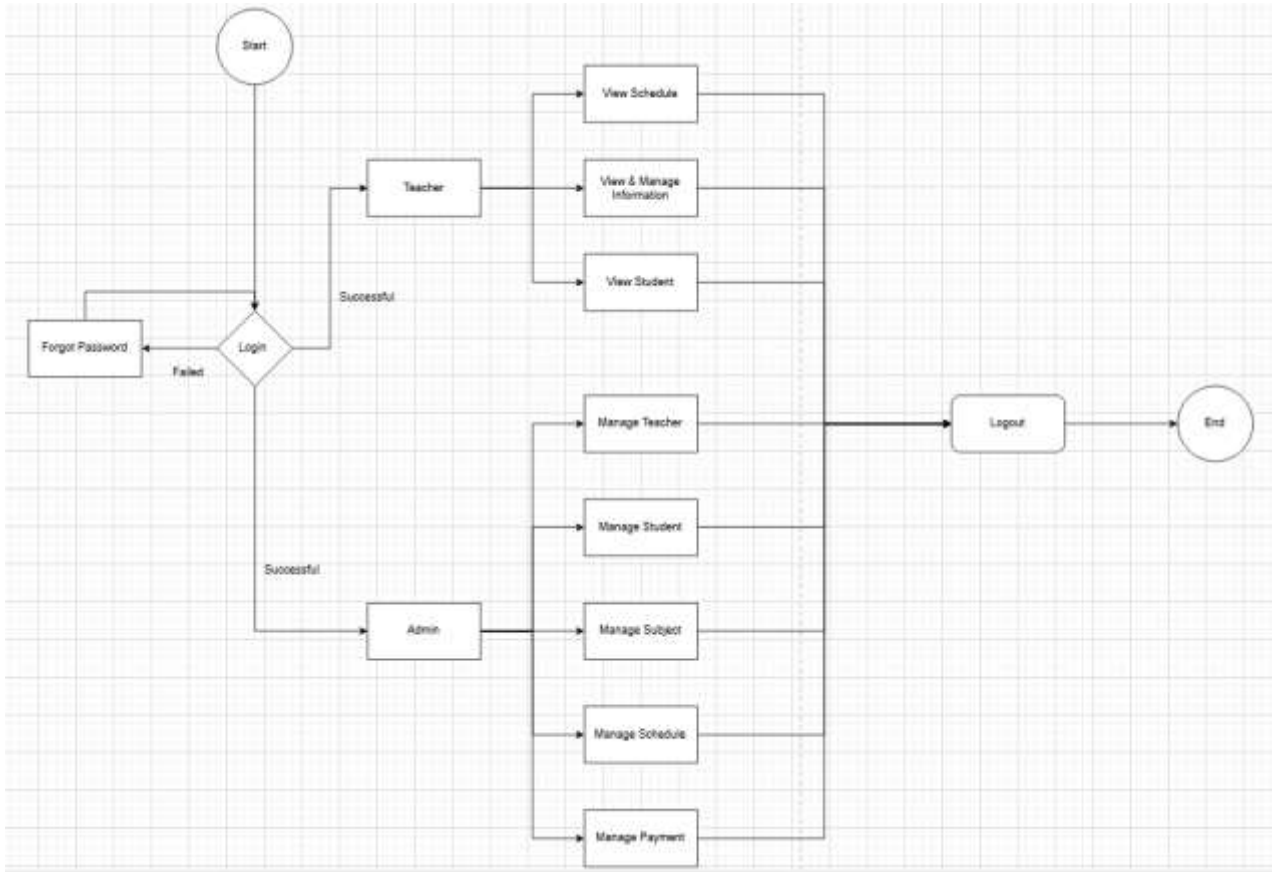


Figure 6. 22: Flow of the system

Figure 6.22 illustrates the overall flow of the Tuition Management System (TMS). The diagram shows the sequence of processes that occur when users interact with the system. The process begins with the user accessing the login page and entering valid login credentials. Once the system verifies the user information, the user is redirected to the system dashboard according to their role, such as administrator or teacher. This authentication process ensures that only authorized users can access the system features.

After successfully logging in, the administrator can perform various system management tasks such as managing student information, teacher records, class schedules, payment records, and generating reports. Teachers can access their assigned schedules, view reports, and update their personal profile information. All actions performed by the users are processed by the system and stored in the database to maintain accurate records. The system flow diagram helps illustrate how the system operates step by step and how users interact with the system to perform different functions.

6.5 Conclusion

In conclusion, the design phase of the Tuition Management System provided a structured blueprint for the development of the system. The interface design, database design, and system diagrams such as the Data Flow Diagram and Entity Relationship Diagram helped illustrate how the system components interact and manage data. A well-planned design ensured that the system structure was organized, efficient, and capable of supporting the required functionalities. This design stage served as an important foundation that guided the implementation phase of the system.

7 IMPLEMENTATION

7.1 Introduction

This chapter presents the implementation phase of the Tuition Management System (TMS). The implementation stage involves transforming the system design into a working application using the selected programming languages, tools, and technologies. It focuses on developing the system modules, integrating the database, and ensuring that all functional and non-functional requirements are properly executed. This chapter also discusses the execution platform, development tools, system interface, and significant features such as security elements. Through this phase, the proposed system is converted from theoretical design into a practical and operational web-based application ready for testing and deployment.

7.2 Execution Platform

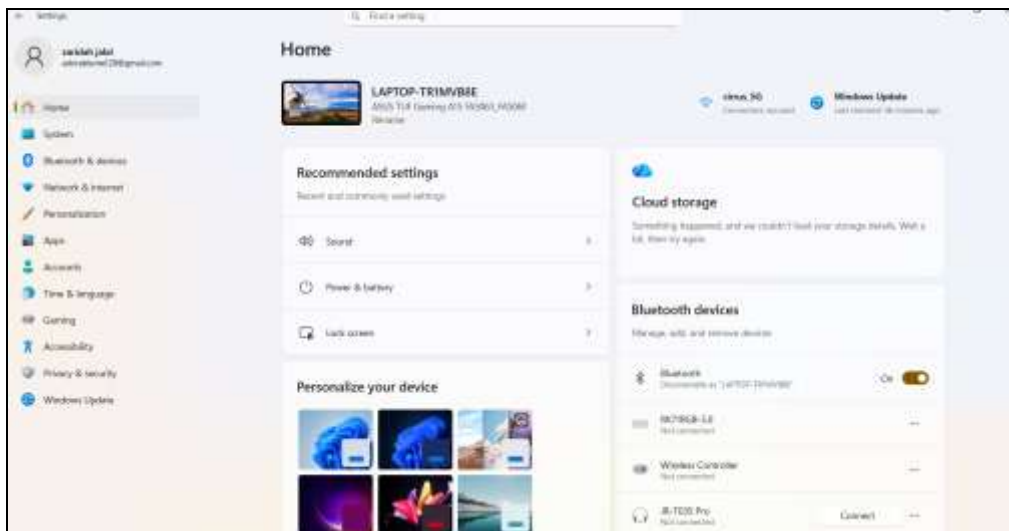


Figure 7. 1: Execution Platform

Figure 7.1 shows the execution platform used to run and test the Tuition Management System (TMS). The execution platform refers to the environment where the system is developed, executed, and evaluated during the implementation phase. This includes the computer system, operating system, and software tools required to support the application. The platform ensures that the system can run smoothly and that all components function correctly together. By using an appropriate execution platform, the developers can test the system performance and ensure that the application operates as expected.

7.3 Implementation Tools

7.3.1 Visual Studio Code

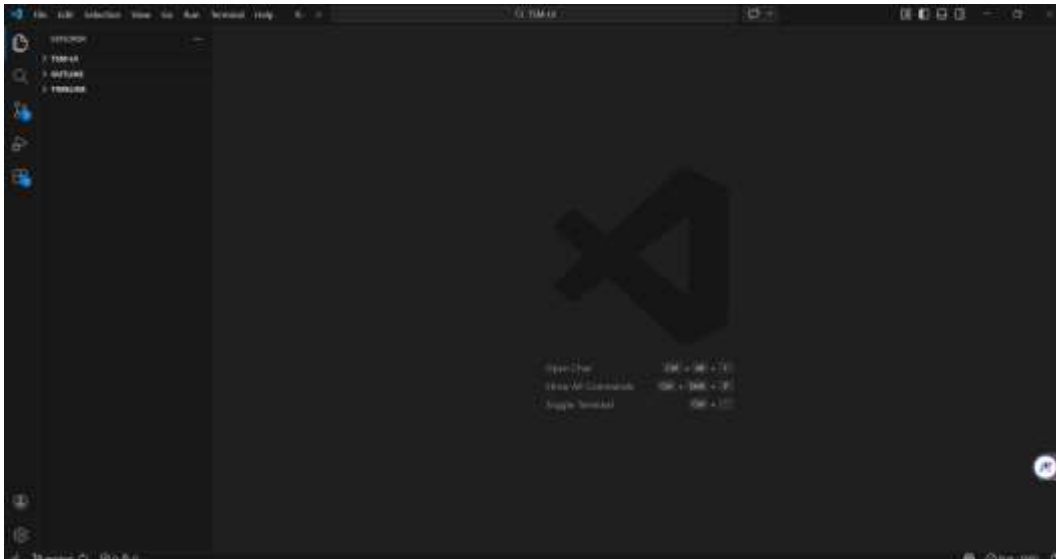


Figure 7. 2: Vs Code

Figure 7.2 illustrates the Visual Studio Code environment used during the development of the Tuition Management System. Visual Studio Code is a lightweight but powerful source code editor that supports various programming languages and extensions. It was used to write, edit, and manage the source code of the system during the implementation stage. The editor also provides useful features such as syntax highlighting, debugging tools, and code suggestions that help improve coding efficiency. By using Visual Studio Code, developers can organize project files and ensure the system is developed in a structured and efficient manner.

7.3.2 IntelliJ

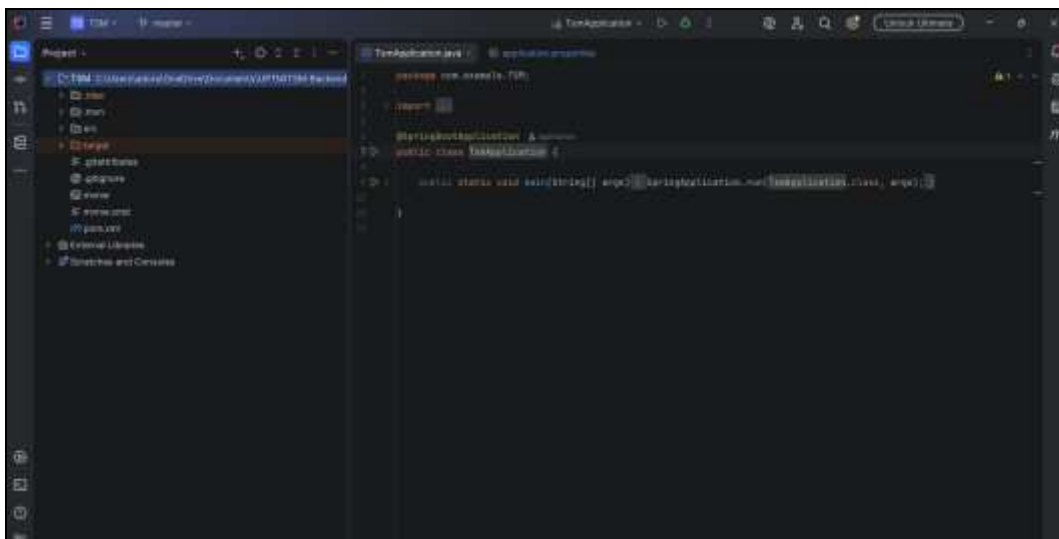


Figure 7. 3: IntelliJ

Figure 7.3 shows the IntelliJ Integrated Development Environment (IDE) used in the development process of the system. IntelliJ provides advanced development tools that support application development,

debugging, and project management. It allows developers to compile code, run the program, and identify errors during development. The environment also supports integration with databases and other development frameworks. By using IntelliJ, the development process becomes more efficient and helps ensure that the system functions properly during implementation.

7.3.3 MySQL Database

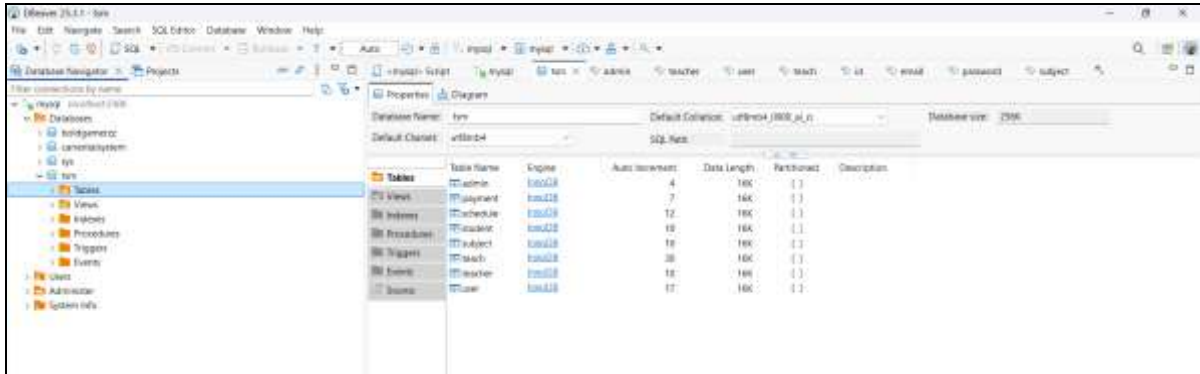


Figure 7. 4: MySQL Database

Figure 7.4 illustrates the MySQL database management system used to store and manage the data for the Tuition Management System. MySQL is a relational database management system that allows structured storage of system data in tables. In this project, MySQL is used to store important information such as student records, teacher details, schedules, subjects, and payment transactions. The database allows the system to retrieve, update, and manage data efficiently. Using MySQL ensures data consistency, reliability, and secure storage of system information.

7.4 System Interface

7.4.1 Log in Page

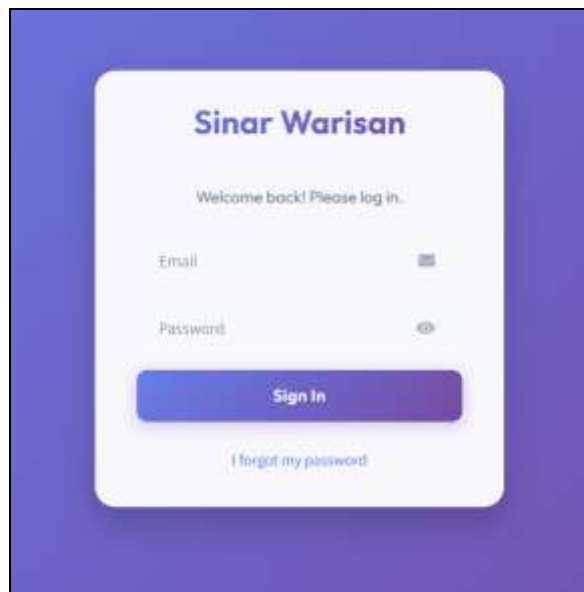


Figure 7. 5: Login Page

Figure 7.5 shows the implemented login page of the Tuition Management System. This page allows users to enter their username and password to access the system. The login function verifies the user credentials stored in the database before granting access to the system. This feature is important for ensuring that only authorized users such as administrators and teachers can use the system. If the credentials are correct, the system redirects the user to the dashboard. Otherwise, an error message will be displayed.

7.4.2 Admin – Dashboard

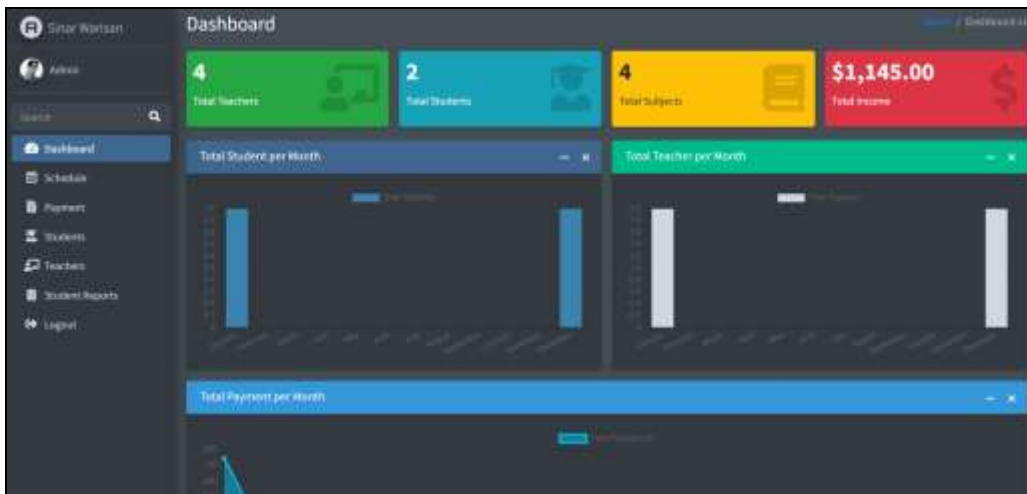


Figure 7. 6: Admin Dashboard

Figure 7.6 presents the dashboard interface of the Tuition Management System. The dashboard acts as the main page after a user successfully logs into the system. It provides an overview of the system and allows users to navigate easily to different modules such as student management, teacher management, schedule management, and payment records. The dashboard helps users quickly access important features without navigating through multiple pages. This interface improves the overall usability and efficiency of the system.

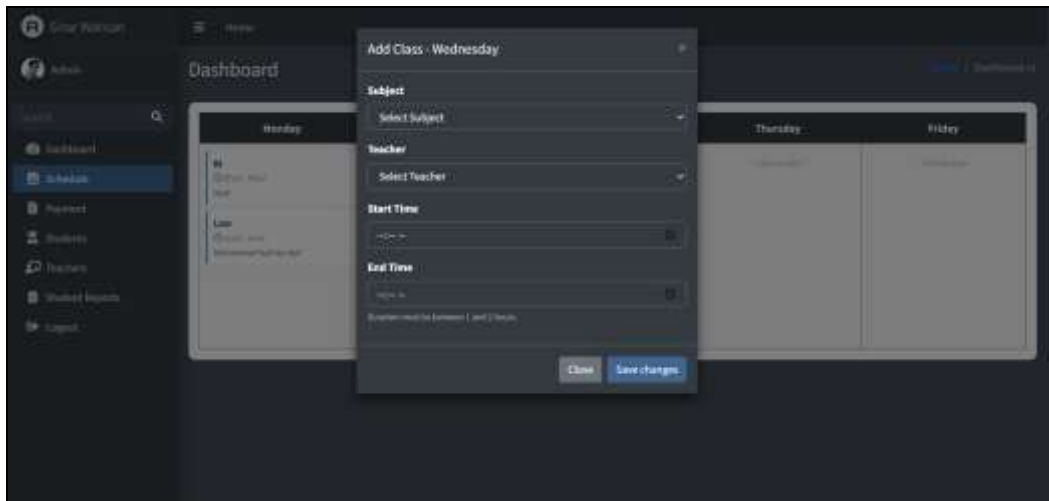


Figure 7. 7: Schedule

Figure 7.7 displays the schedule management interface implemented in the system. This page allows administrators to create and manage tuition class schedules. The administrator can enter details such as class date, time, subject, and assigned teacher. The system also helps prevent schedule conflicts by checking if the selected time slot is already occupied. This feature ensures that class sessions are properly organized. It also allows teachers to view their assigned teaching schedules.

7.4.4 Admin – Payment

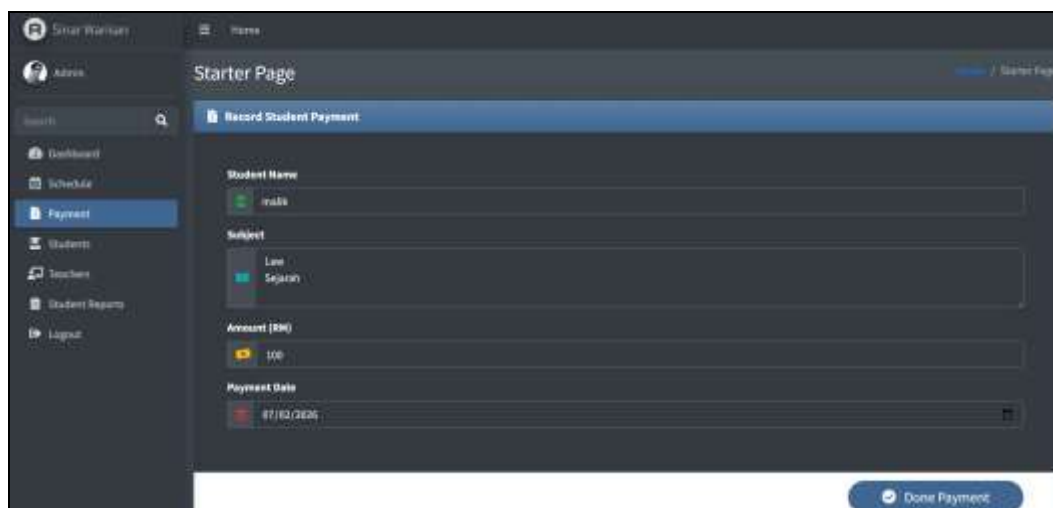


Figure 7. 8: Payment

Figure 7.8 illustrates the payment management interface used by administrators. This page allows the administrator to record tuition fee payments made by students. The system stores payment details such as student name, payment amount, and payment date. These records are saved in the database for future reference and monitoring. The payment management feature helps ensure accurate financial tracking within the tuition center. It also simplifies the process of managing student payments.

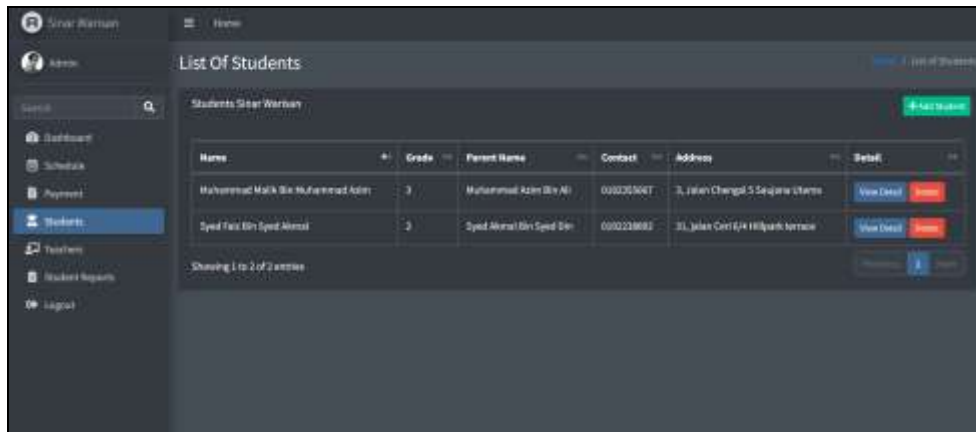


Figure 7. 9: Student

Figure 7.9 shows the student management interface implemented in the system. Through this page, the administrator can add, edit, or delete student records. The system stores important student information such as name, contact details, address, and enrolled subjects. This feature helps maintain an organized record of students within the tuition center. By using this interface, administrators can easily manage and update student data whenever necessary.

7.4.6 Admin – Teachers

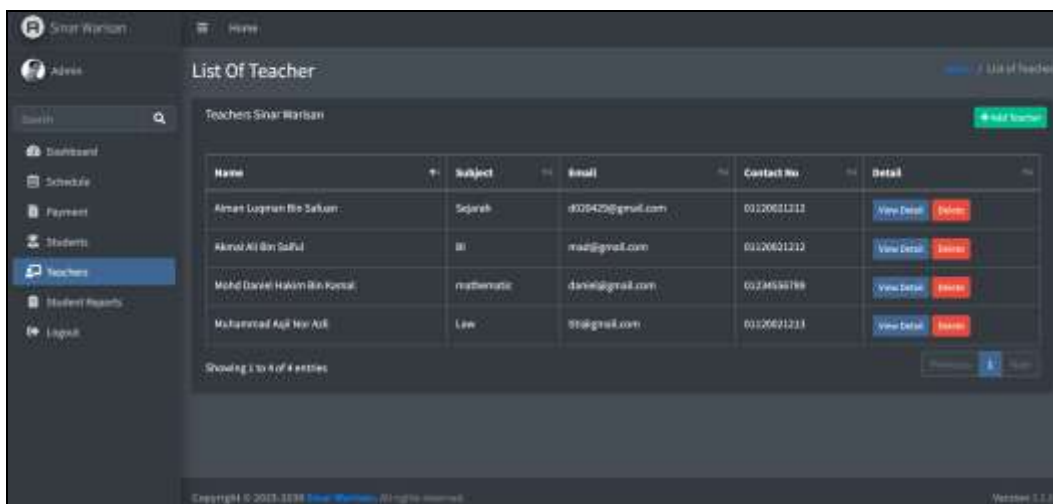


Figure 7. 10: Teachers

Figure 7.10 presents the teacher management interface used to manage teacher information. The administrator can add new teacher records, update existing information, or remove teacher data from the system. The information stored includes teacher name, email, contact details, and subject specialization. This feature helps keep track of all teachers working in the tuition center. Proper teacher management also helps in assigning subjects and schedules efficiently.

7.4.7 Admin – Student Report

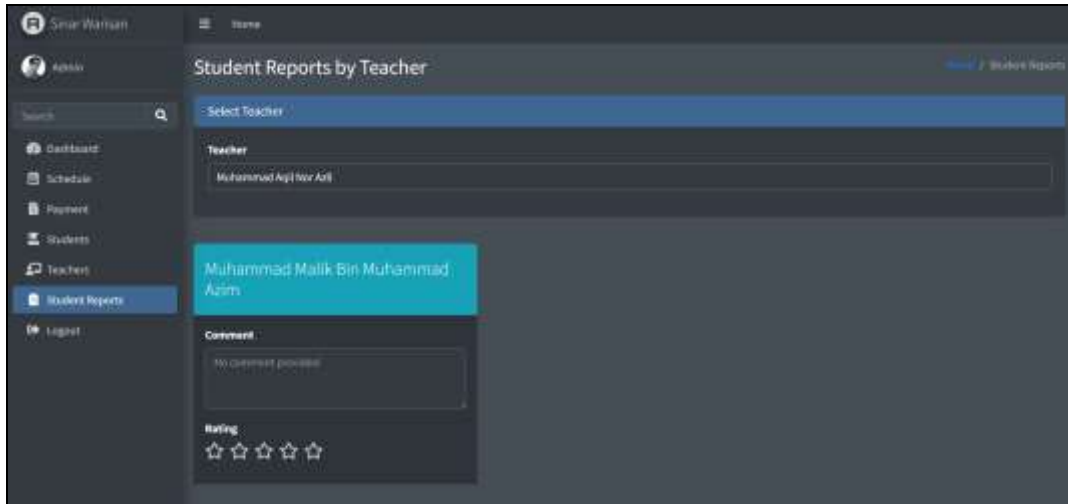


Figure 7. 11: Student Report

Figure 7.11 illustrates the student report interface available in the system. This page allows the administrator or teacher to view student performance records. The report may include student marks, comments, or performance evaluations provided by the teacher. These reports help monitor student progress and identify areas where improvement may be needed. The reporting feature also helps administrators maintain academic records in an organized format.

7.4.8 Teacher – Schedule

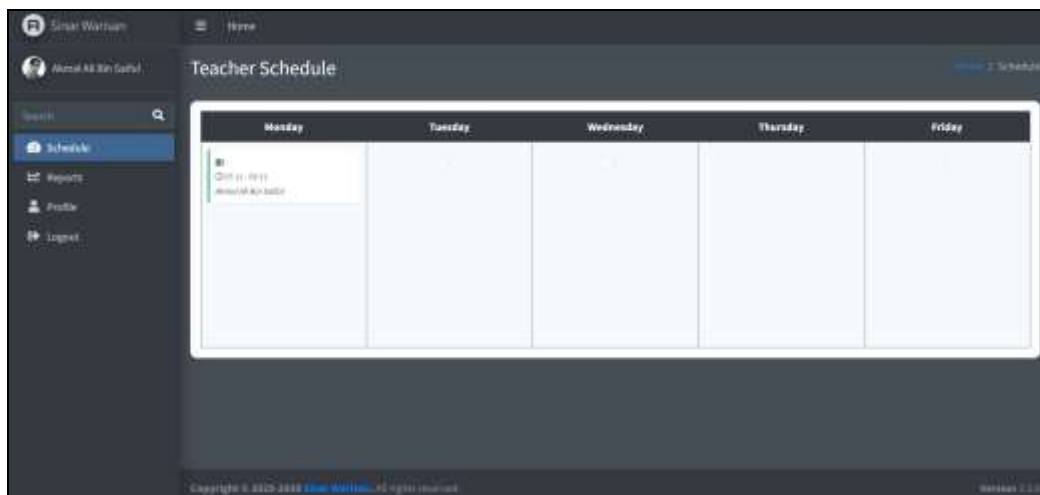


Figure 7. 12: Teacher Schedule

Figure 7.12 shows the teacher schedule interface that allows teachers to view their assigned teaching schedules. Teachers can see information such as class date, time, and subject they are responsible for teaching. This feature helps teachers plan their teaching activities more efficiently. The schedule information is generated based on the schedules created by the administrator. This ensures proper coordination between the administrator and teachers.

7.4.9 Admin – Payment Record

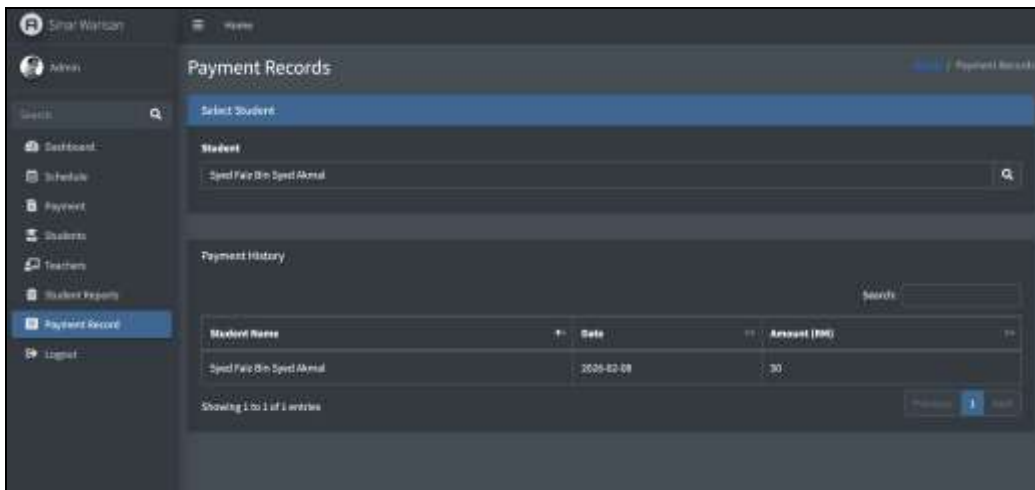


Figure 7. 13: Payment Record

Figure 7.13 presents the payment record interface where all recorded payment transactions are displayed. This page allows administrators to review and monitor tuition fee payments made by students. The system organizes payment data in a structured format for easy tracking. Administrators can quickly check whether students have paid their fees or if there are outstanding payments. This feature helps improve financial management within the tuition center.

7.4.10 Teacher – Reports

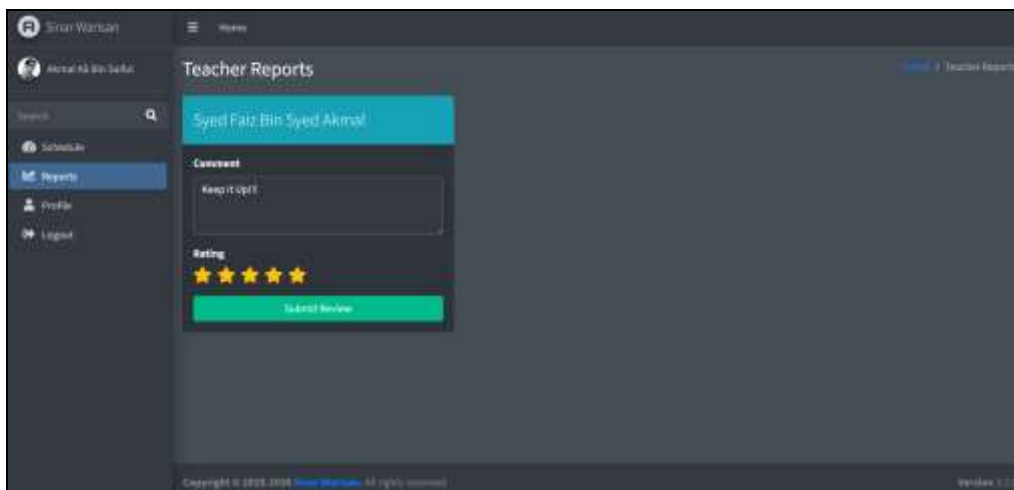


Figure 7. 14: Teacher Report

Figure 7.14 shows the teacher report interface used for recording or viewing student performance information. Teachers can enter comments, marks, or evaluation results for students based on their performance in class. This information is stored in the system database for future reference. The reporting function helps track student learning progress and supports better academic monitoring. It also allows administrators to review student performance records.

system grants access to the user and redirects them to the appropriate dashboard. If the login information is incorrect, the system will display an error message. This mechanism ensures that system access remains secure.

7.5.2 Forgot Password

```

187 <!-- forgot password.html -->
188
189 <html lang="en">
190
191 <body class="hold-transition login-page">
192
193 <div class="bg-shape shape-1"></div>
194 <div class="bg-shape shape-2"></div>
195
196 <div class="login-box">
197 <div class="card">
198 <div class="card-header text-center">
199 <h1>Sinar Marisan</h1>
200 </div>
201 <div class="card-body">
202 <p class="login-box-msg">You forgot your password? Here you can easily retrieve a new password.</p>
203
204 <form id="forgotpasswordform">
205 <div class="input-group mb-3">
206 <input type="email" class="form-control" placeholder="Email" required>
207 <div class="input-group-append">
208 <div class="input-group-text">
209 <span class="fas fa-envelope"></span>
210 </div>
211 </div>
212 </div>
213 <div class="row">
214 <div class="col-12">
215 <button type="submit" class="btn btn-primary btn-block">Request new password</button>
216 </div>
217 </div>
218 </div>
219 </div>
220 <div class="auth-links">
221 <p class="mb-1">
222 <a href="login.html">login</a>
223 </p>
224 </div>
225 </div>

```

Figure 7. 17: Forgot Password

Figure 7.17 illustrates the forgot password feature provided in the system. This feature allows users to recover access to their accounts if they forget their login password. The system guides the user through a process to reset or retrieve their password. This improves user convenience and prevents account lockouts. The feature also helps maintain system accessibility without requiring administrator intervention.

7.6 Conclusion

In conclusion, the implementation phase successfully transformed the system design into a functional Tuition Management System. Various development tools such as Visual Studio Code, IntelliJ, and MySQL were used to build and integrate the system components. The developed interfaces allow administrators and teachers to manage schedules, students, payments, and reports efficiently. Overall, the implementation stage demonstrated how the system design was translated into a working application that fulfils the system requirements.

8 TESTING

8.1 Introduction

This chapter discusses the testing phase of the Tuition Management System (TMS). Testing is a critical process to ensure that the system functions correctly, efficiently, and securely according to the specified requirements. The purpose of testing is to identify errors, verify system performance, and confirm that each module operates as expected. Various testing approaches, including unit testing, integration testing, system testing, and user acceptance testing, are carried out to evaluate different aspects of the system. This chapter explains the testing strategies applied, the test plans used, and the results obtained to validate the reliability and readiness of the system before full deployment.

8.2 Unit Testing

Unit testing refers to the process of testing individual components or modules of the system separately to ensure that each function works correctly. In the Tuition Management System (TMS), unit testing focuses on testing specific features such as the login function, student management, teacher management, schedule creation, and payment recording. Each module is tested independently with different inputs to verify that the expected output is produced. For example, the login module is tested using both valid and invalid credentials to check whether the system allows or denies access appropriately. Error handling is also tested to ensure that the system displays suitable error messages when incorrect data is entered. This testing helps identify programming errors early in the development process before integrating modules together. By performing unit testing, developers can ensure that every individual component of the system functions correctly and reliably.

Table 8. 1: Unit Testing

Unit/Function	Description	Input	Output	Error Handling	Status
Log in Page	Test login with valid credentials	Enter correct username and password	User successfully logs in and dashboard page appears	None	Pass
Log in Page with wrong credentials	Test login with invalid credentials	Enter wrong username or password	System displays "Login failed, Wrong email or password"	Error: Login failed, Wrong email or password	Pass
Admin - Student	Test adding a new student record	Enter student name, parent name and contact, address,	Student record saved in database	If required fields are empty, system failed to save student information in database	Pass

		subject and teacher			
Admin - Teachers	Test adding teacher information record	Enter teacher name, email, contact and subject	Teacher record saved in database	If required fields are empty, system failed to save teacher information in database	Pass
Admin - Schedule	Test creating a class schedule	Enter date, time and teacher	Schedule successfully created	If schedule time were clashing the system will display "The selected day and time are already booked by another subject"	Pass
Admin - Payment	Test recording student payment	Enter student name and date	Payment record successfully saved	None	Pass
Teacher - Profile	Test updating teacher profile	Edit profile information and click update	Updated profile displayed	None	Pass

8.3 Integration Testing

Integration testing is conducted after unit testing to verify that different modules of the system work correctly when combined together. In the Tuition Management System, integration testing focuses on how different components such as login, dashboard, schedule management, payment records, and student management interact with each other. This testing ensures that data flows correctly between modules and that the integration of components does not produce unexpected errors. For instance, after logging into the system, the user should be redirected to the dashboard and able to access other modules such as student or schedule management. Integration testing also verifies that related data is correctly connected, such as linking student information with payment records. If any module fails to communicate with another module properly, errors can occur during system operation. Therefore, integration testing ensures that the entire system works smoothly as a connected application rather than as isolated components.

Table 8. 2: Integration Testing

Unit/Function	Description	Input	Output	Error Handling	Status
Login Dashboard	Test system navigation after login	Enter valid login credentials	System redirects to dashboard	If login fails, dashboard is not accessible	Pass
Student Payment	Test linking student with payment record	Select student and enter payment details	Payment linked to correct student	System shows error if student record does not exist	Pass
Schedule Teacher	Test assigning teacher schedule	Assign teacher to class schedule	Teacher schedule displayed correctly	System error if the schedule time were clash	Pass
Student Performance	Test generating student performance	Select student and give a mark and comment based on their performance	Student mark and comment will be displayed	System shows no mark and comment if the teacher does not assign them	Pass
Payment Record	Test payment data integration	Record payment and save	Payment appears in payment record list	System displays "no records found" if payment does not save	Pass

8.4 System Testing

System testing evaluates the entire system as a complete and integrated application to ensure that it meets the specified requirements. In the Tuition Management System, system testing examines both functional and non-functional aspects of the application. Functional testing verifies whether system features such as login, managing student information, teacher management, schedule creation, and payment recording perform according to the defined requirements. Non-functional testing evaluates system qualities such as performance, security, usability, and reliability. For example, the system must respond quickly when loading the dashboard and must prevent unauthorized users from accessing system features. System testing also ensures that the interface is user-friendly and that users can easily navigate between different modules. This stage of testing helps confirm that the system operates correctly as a whole and meets both technical and user requirements.

Table 8. 3: Functional system testing

Unit/Function	Description	Input	Output	Error Handling	Status
Login Function	Test whether admin or teacher can log in to the system	Enter valid username and password	User successfully logs in and dashboard page is displayed	Error message shown for invalid login	Pass
Manage Student	Test admin ability to add or update student information	Enter student details and click save	Student data stored and displayed in student list	System displays “failed to save” if required fields are empty	Pass
Manage Teacher	Test admin ability to add teacher information	Enter teacher details and save	Teacher record successfully stored	System prevents saving incomplete data	Pass
Schedule Management	Test admin creating a schedule for teachers	Enter schedule date, time, and teacher	Schedule created and displayed correctly	System shows message if schedule has been clashed	Pass
Payment Management	Test admin recording student payment	Enter student name, payment amount, and date	Payment record successfully saved	System displays error if payment information invalid	Pass
Teacher Schedule	Test teacher viewing assigned schedule	Teacher logs in and selects schedule page	Teacher schedule displayed correctly	The schedule will be empty if admin does not field teacher schedule	Pass

8.4.2 Non-Functional Testing

Table 8. 4: Non-Functional system testing

Test type	Description	Input	Output	Error Handling	Status
Performance	Test system response time when loading dashboard	User logs in to system	Dashboard loads within a few seconds	System reloads or displays loading indicator if delay occurs	Pass
Security	Test system protection against unauthorized access	Enter invalid login credentials multiple times	System blocks access to unauthorized users	System displays error message and denies access	Pass
Usability	Test whether the system interface is easy to use	User navigates through menus and pages	User can easily access features and understand interface	Help messages or prompts displayed if user makes mistakes	Pass
Reliability	Test system stability during continuous use	Use system continuously for several operations	System remains stable without crashing	System recovers from minor errors without data loss	Pass

8.5 Acceptance Testing

Acceptance testing is the final stage of testing performed to ensure that the system meets user requirements and is ready for actual use. This testing is usually conducted from the perspective of the end users, such as administrators and teachers who will use the system in daily operations. In the Tuition Management System, acceptance testing verifies whether users can successfully perform their tasks such as logging into the system, managing student information, recording payments, viewing schedules, and generating reports. The goal of this testing is to confirm that the system behaves as expected in real usage scenarios. Users test the system functions and provide feedback on whether the system meets their needs. If any issues are discovered, improvements can be made before the system is officially deployed. Acceptance testing ensures that the system is practical, usable, and ready to be implemented in a real tuition management environment.

Table 8. 5: Acceptance Testing

User Role	Description	Input	Output	Error Handling	Status
Admin	Verify admin can access dashboard	Admin enters login credentials	Admin dashboard displayed	Access denied if credentials invalid	Pass
Admin	Verify admin can manage system data	Admin adds student, teacher, and schedule	Data stored correctly	System alerts if required fields empty	Pass
Admin	Verify admin can record student payment	Admin enters payment information	Payment successfully recorded	System displays error if payment details incorrect	Pass
Teacher	Verify teacher can view schedule	Teacher logs in and opens schedule	Schedule displayed correctly	The schedule will be empty if admin does not field teacher schedule	Pass
Teacher	Verify teacher can view profile and student performance	Teacher opens profile and student performance page	Information displayed successfully	System alerts if data cannot be loaded	Pass

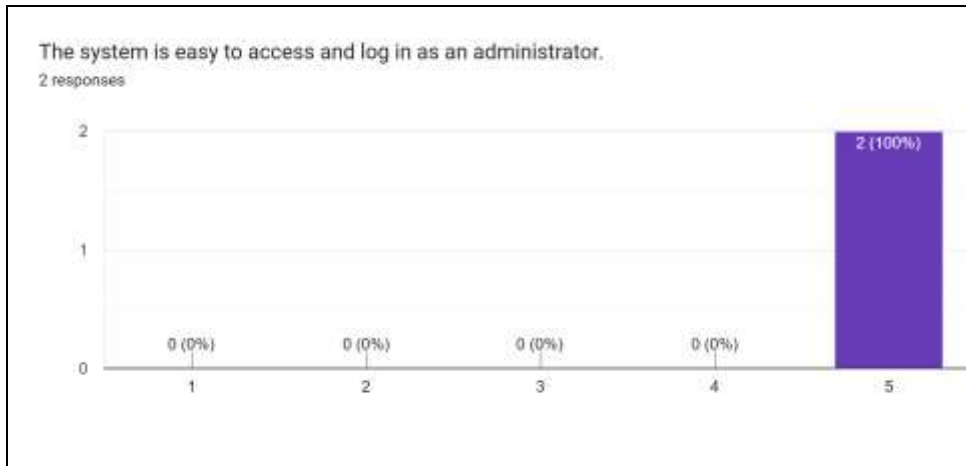


Figure 8. 1: Question 1

This figure presents the responses from admin users regarding Question 1 in the post-test questionnaire. Most respondents show a positive response, indicating that the system feature evaluated in this question works effectively. The result suggests that the admin users are satisfied with this aspect of the system. Overall, the system performs well based on the admin feedback.

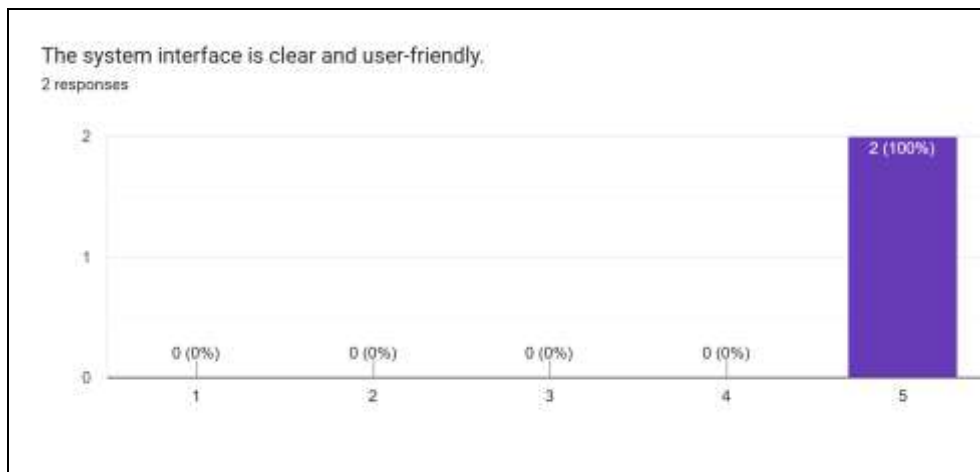


Figure 8. 2: Question 2

Figure 8.2 illustrates the admin responses for Question 2. The majority of respondents selected positive options, which indicates that the system function evaluated is easy to understand and use. This shows that the system interface is user-friendly for admin users. The result also reflects a good level of acceptance among the respondents.

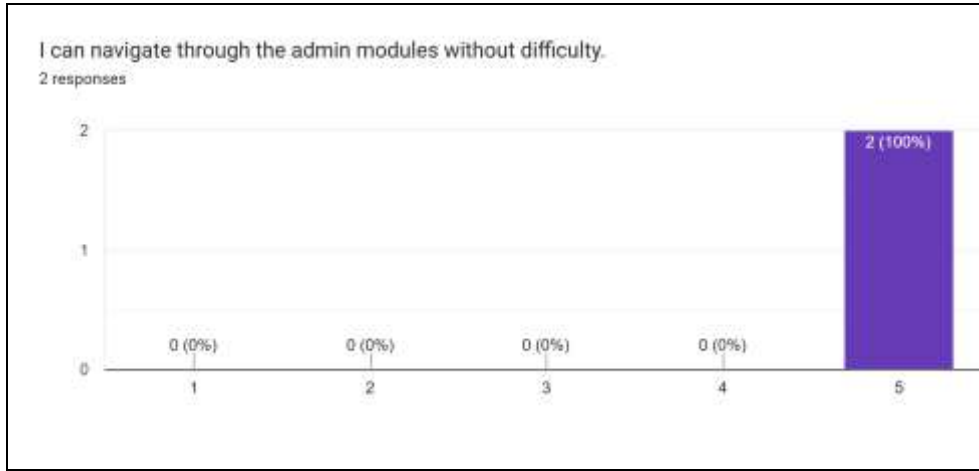


Figure 8. 3: Question 3

This figure shows the distribution of responses from admin users for Question 3. Most responses indicate agreement, suggesting that the system helps improve efficiency in performing administrative tasks. The results demonstrate that the system is useful for supporting admin operations. Therefore, the feature is considered effective by most respondents.

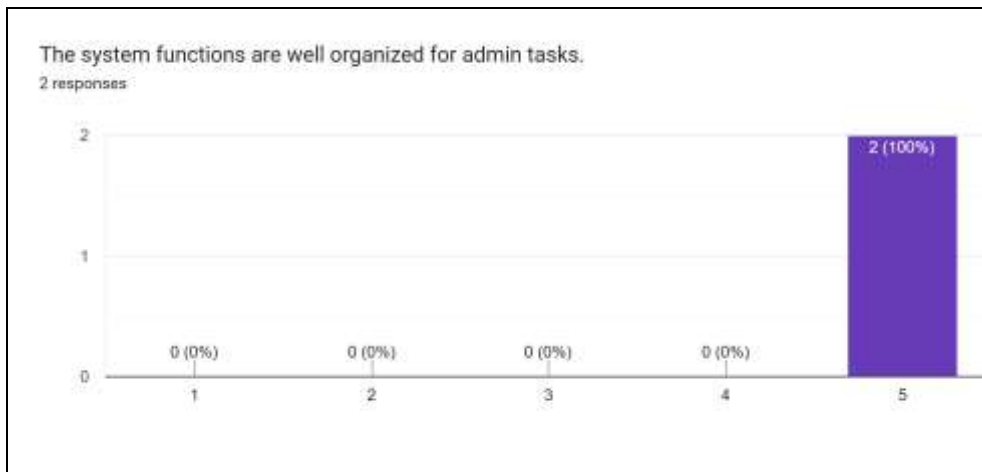


Figure 8. 4: Question 4

Figure 8.4 presents the feedback from admin users regarding Question 4. The responses indicate that most admins are satisfied with the functionality provided by the system. This suggests that the system meets the expectations of the users. The positive feedback also reflects good system usability.

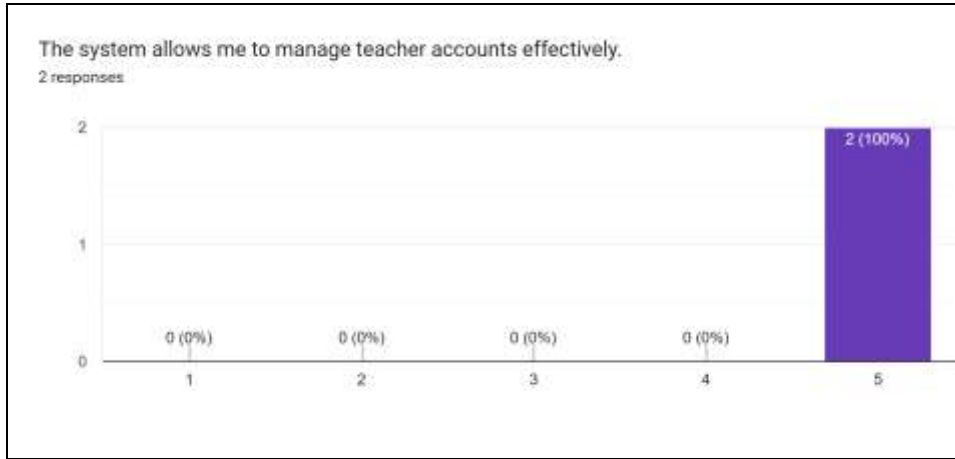


Figure 8. 5: Question 5

This figure shows the responses related to Question 5 from the admin perspective. Most respondents agree that the system helps simplify their tasks and improves workflow efficiency. The result demonstrates that the system contributes positively to the management process. Overall, the admin users show a favourable response.

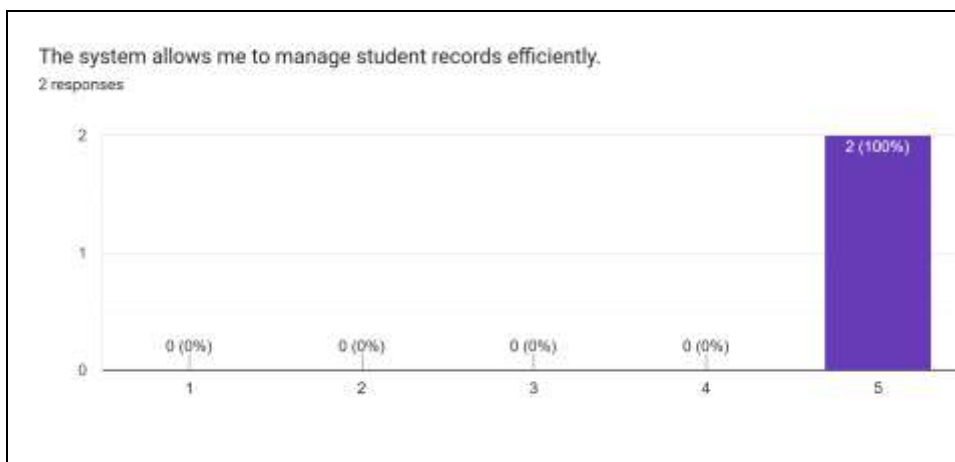


Figure 8. 6: Question 6

Figure 8.6 illustrates the responses from admin users regarding the system feature evaluated in Question 6. The majority of the respondent's express satisfaction with the system's performance. This indicates that the system is reliable, and functions as expected. The responses also show that the system is useful in supporting administrative activities.

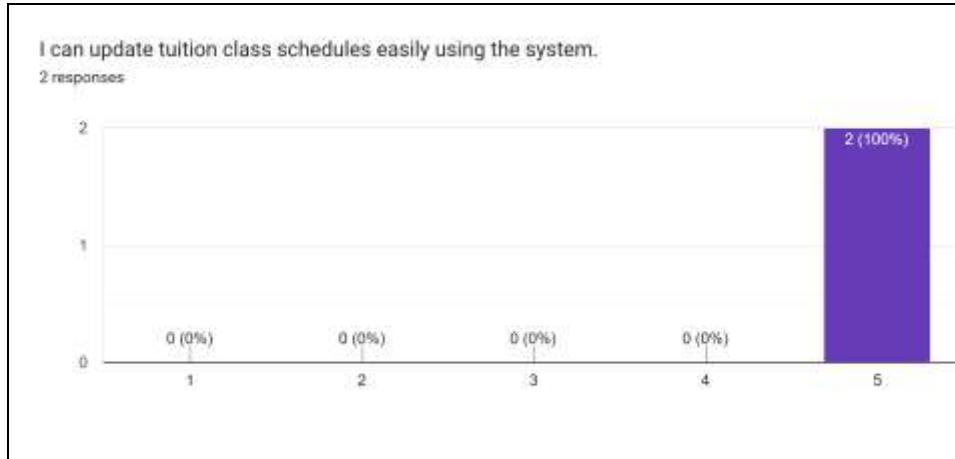


Figure 8. 7: Question 7

This figure displays the distribution of admin responses for Question 7. The results indicate that most respondents agree with the statement provided in the questionnaire. This suggests that the system feature being evaluated is well implemented. It also reflects positive user acceptance among admin users.

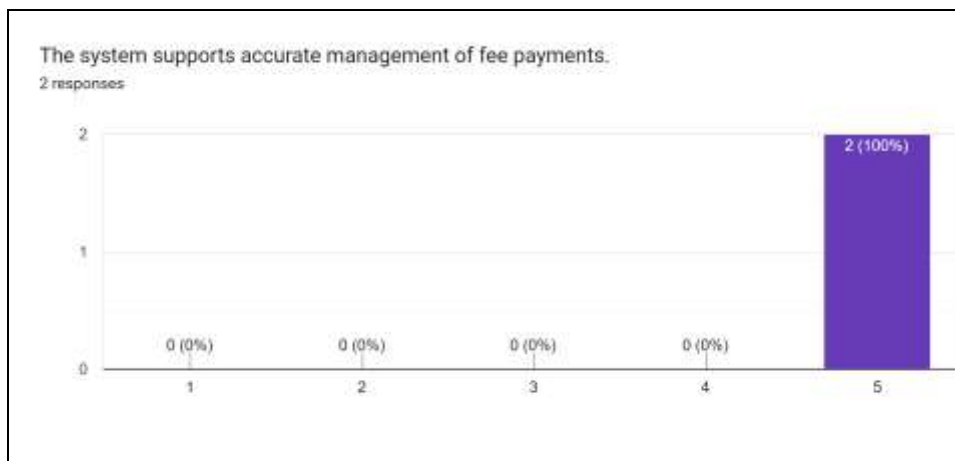


Figure 8. 8: Question 8

Figure 8.8 presents the feedback from admin respondents regarding Question 8. Most of the responses indicate agreement, showing that the system interface is clear and easy to navigate. This result highlights the effectiveness of the system design. The system is therefore considered convenient for admin users.

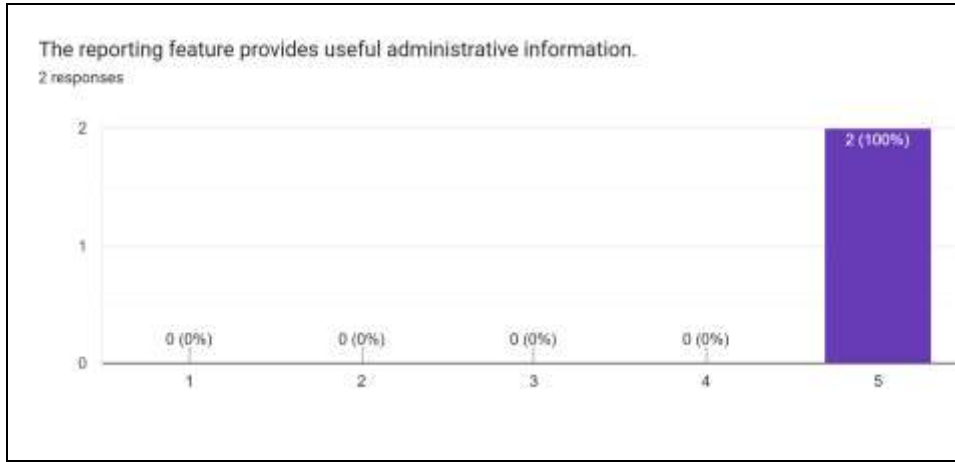


Figure 8. 9: Question 9

This figure shows the responses from admin users for Question 9. The majority of the respondents indicate a positive evaluation of the system feature tested. This suggests that the system helps improve productivity when performing administrative tasks. Overall, the responses demonstrate good system performance.

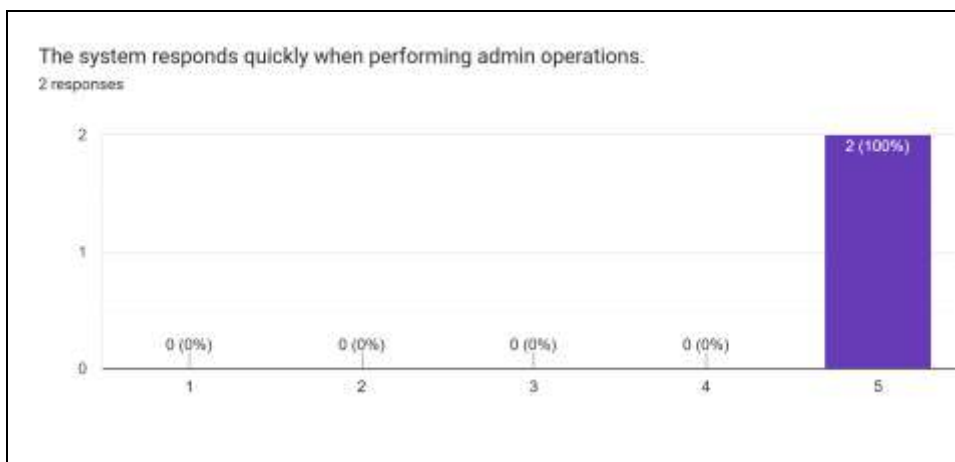


Figure 8. 10: Question 10

Figure 8.10 illustrates the admin feedback for Question 10. The responses show that most users agree with the effectiveness of the system. This indicates that the feature being evaluated meets user requirements. The system is therefore considered helpful for administrative management.

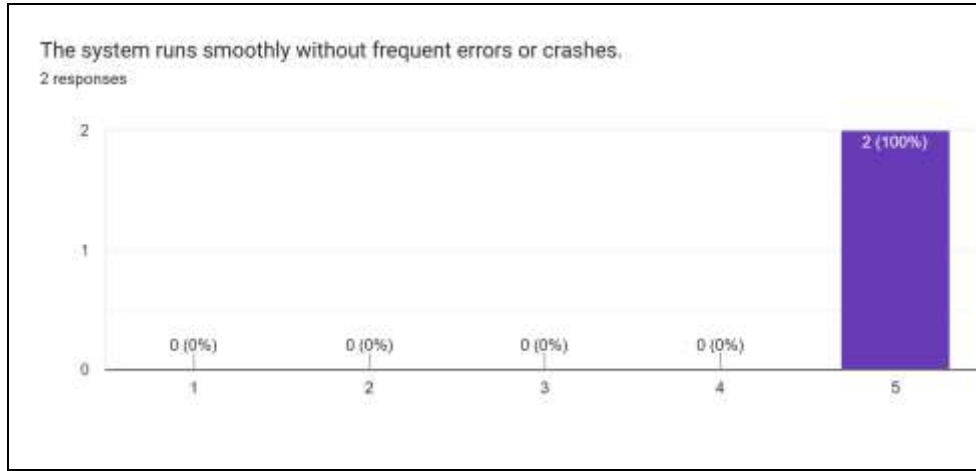


Figure 8. 11: Question 11

This figure presents the responses from admin users regarding Question 11. Most respondents provide positive feedback, indicating that the system functions properly without major difficulties. This suggests that the system is stable and reliable. The result reflects a high level of user satisfaction.

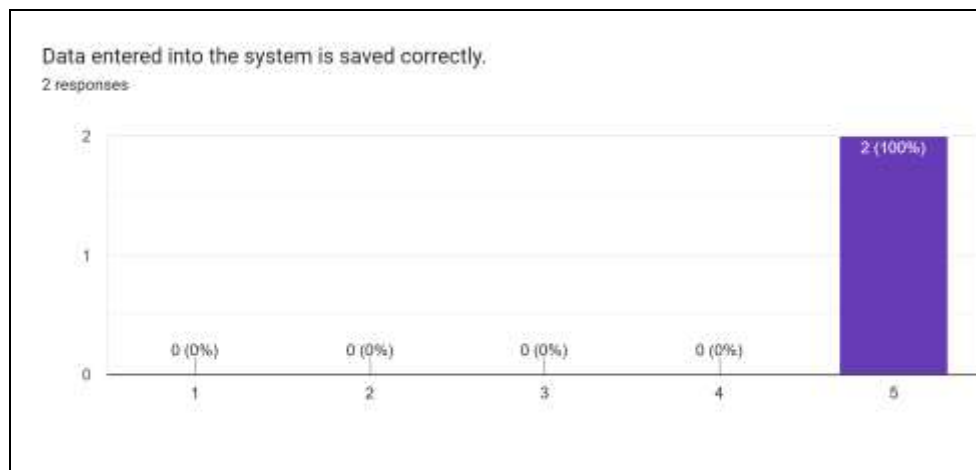


Figure 8. 12: Question 12

Figure 8.12 shows the admin responses for Question 12. The majority of respondents agree that the system is useful in completing their tasks. This demonstrates that the system improves efficiency and productivity. Overall, the system is considered beneficial for administrative work.

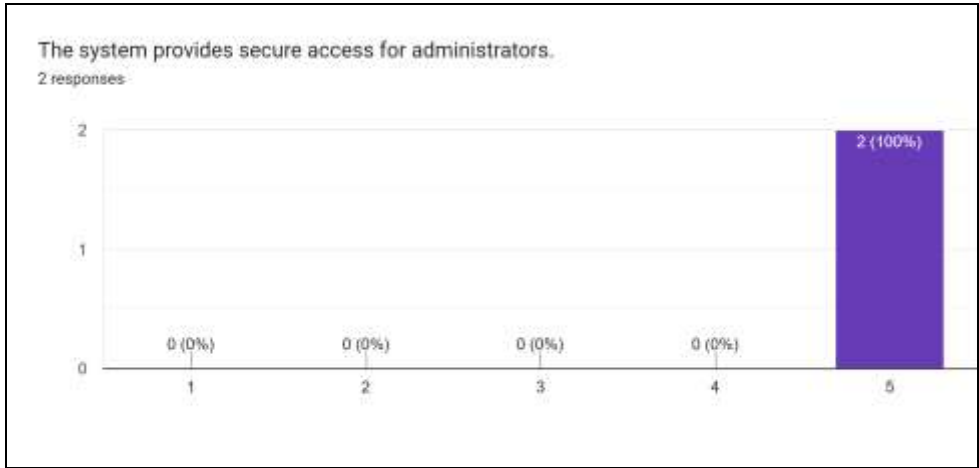


Figure 8. 13: Question 13

This figure illustrates the responses from admin users regarding Question 13. The results indicate that most respondents have a positive perception of the system functionality. This suggests that the system provides helpful features for admin operations. The feedback also indicates good usability.

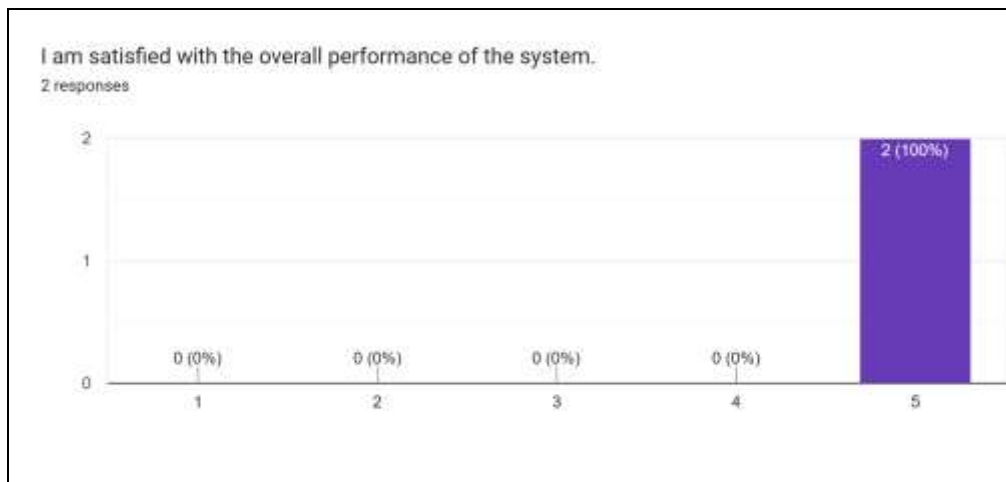


Figure 8. 14: Question 14

Figure 8.14 presents the responses collected from admin users for Question 14. The majority of respondents express satisfaction with the system's performance. This shows that the system meets user expectations. The results also indicate that the system operates efficiently.

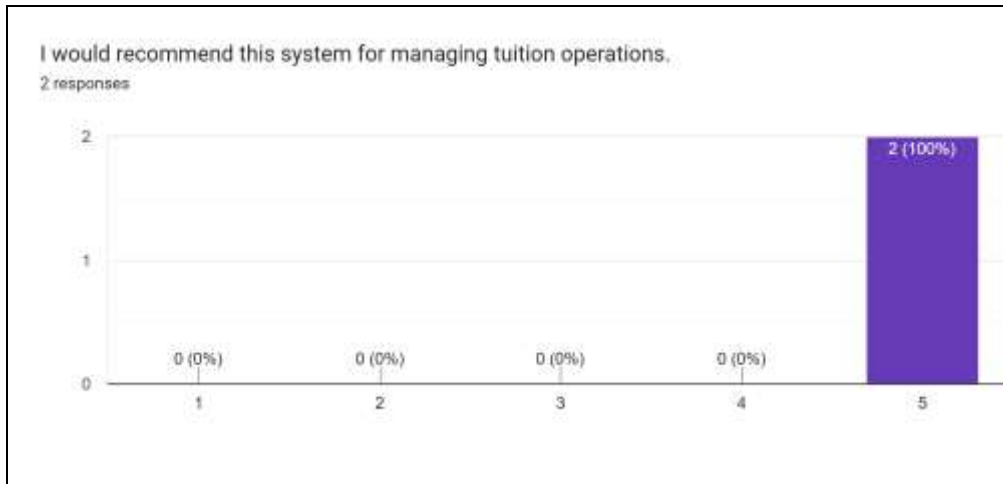


Figure 8. 15: Question 15

This figure shows the final question responses from admin users in the post-test questionnaire. Most respondents provide positive feedback regarding the overall system performance. This indicates that the system is effective and beneficial for administrative tasks. Overall, the system receives a good evaluation from the admin users.

Post-Test Analysis – Questionnaire (Teacher)

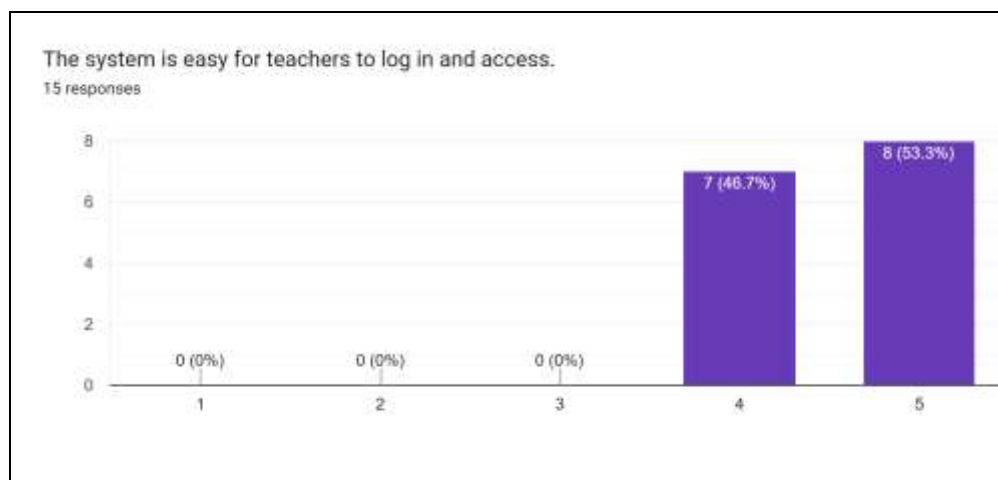


Figure 8. 16: Question 1

Figure 8.16 presents the responses from teacher users regarding Question 1. The results indicate that most respondents agree with the statement provided in the questionnaire. This suggests that the system is easy for teachers to understand and use. The responses show positive acceptance among teacher users.

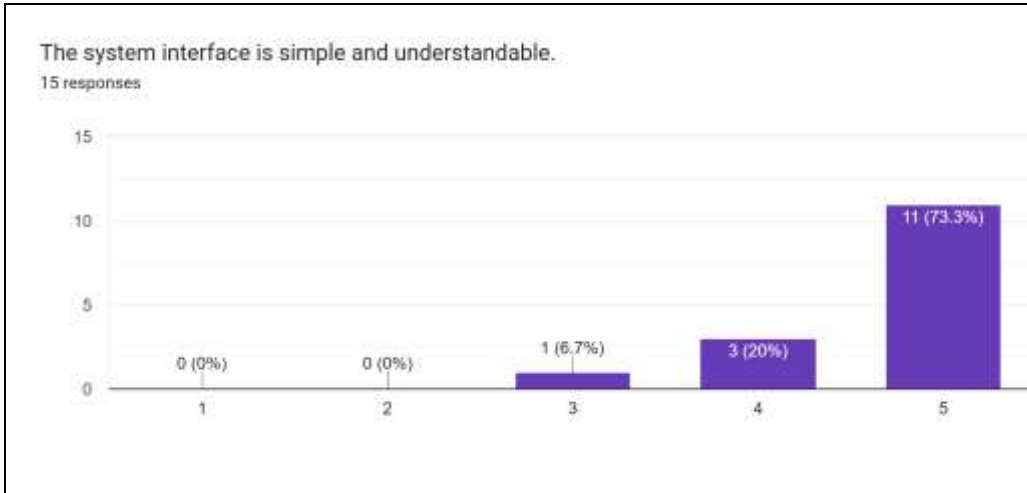


Figure 8. 17: Question 2

This figure shows the distribution of teacher responses for Question 2. The majority of respondents express agreement, indicating that the system supports their tasks effectively. This demonstrates that the system provides useful features for teachers. Overall, the feedback reflects good usability.

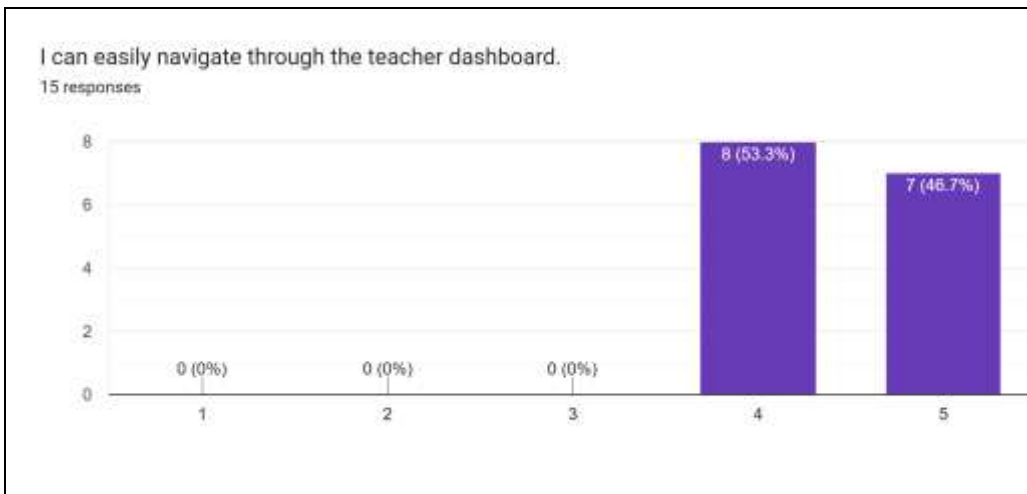


Figure 8. 18: Question 3

Figure 8.18 illustrates the responses from teachers regarding Question 3. Most respondents indicate satisfaction with the system feature being evaluated. This suggests that the system helps teachers perform their tasks more efficiently. The results reflect positive user experience.

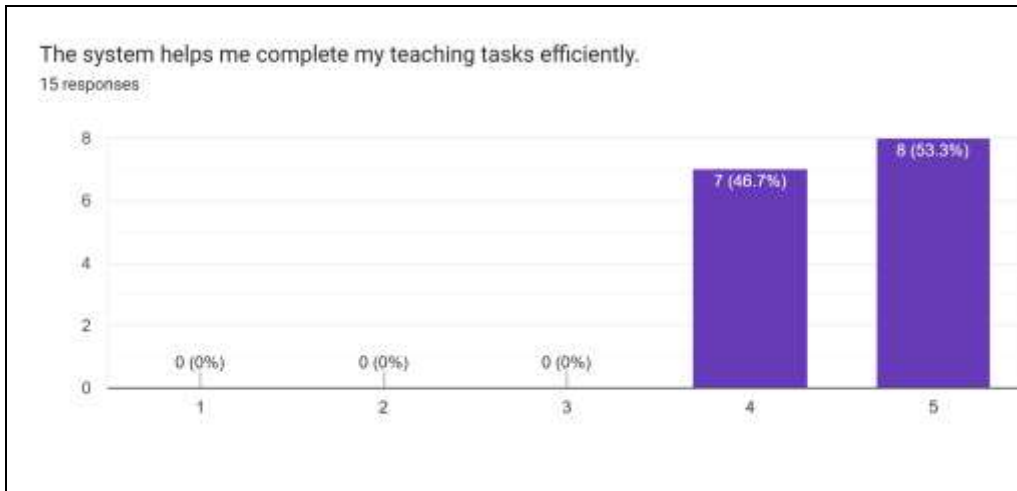


Figure 8. 19: Question 4

This figure presents the responses from teachers for Question 4. The results show that most respondents have a positive perception of the system. This indicates that the system functionality is appropriate for teaching-related activities. The feedback demonstrates overall user satisfaction.

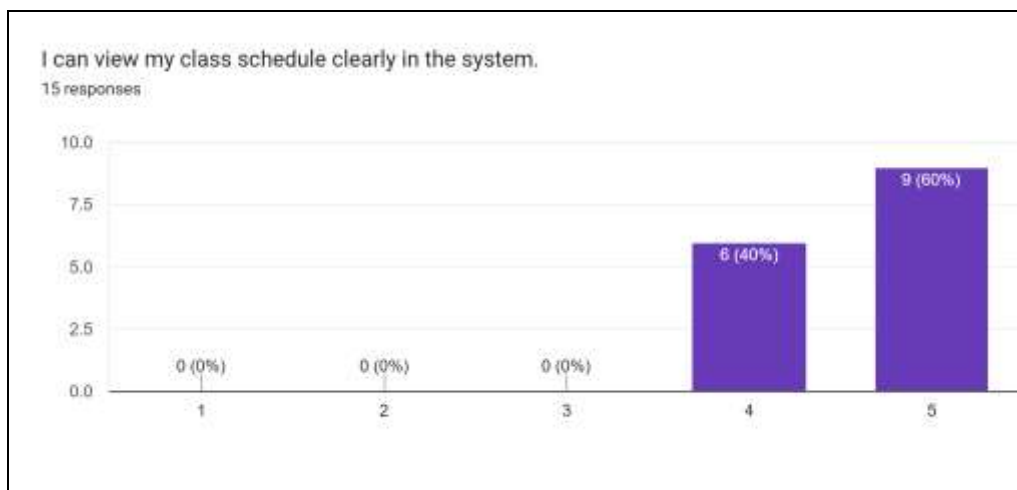


Figure 8. 20: Question 5

Figure 8.20 displays the teacher responses for Question 5. Most respondents agree that the system improves the efficiency of their work. This suggests that the system provides helpful tools for teachers. The results indicate that the system contributes positively to their tasks.

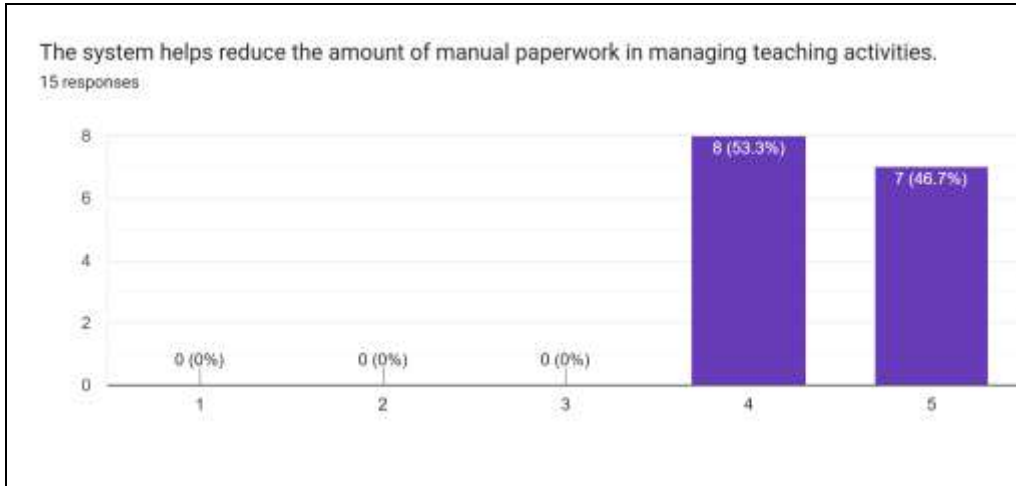


Figure 8. 21: Question 6

This figure shows the responses from teachers regarding Question 6. The majority of the respondents provide positive feedback about the system performance. This suggests that the system is reliable and easy to use. The results indicate good user acceptance.

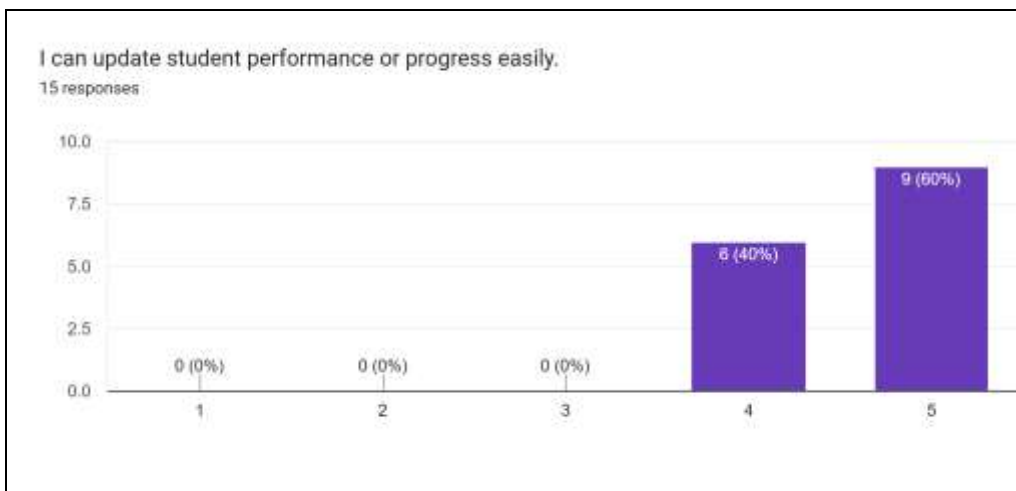


Figure 8. 22: Question 7

Figure 8.22 presents the responses collected from teachers for Question 7. Most respondents indicate agreement with the statement provided. This suggests that the system design is clear and understandable. The responses demonstrate that teachers are comfortable using the system.

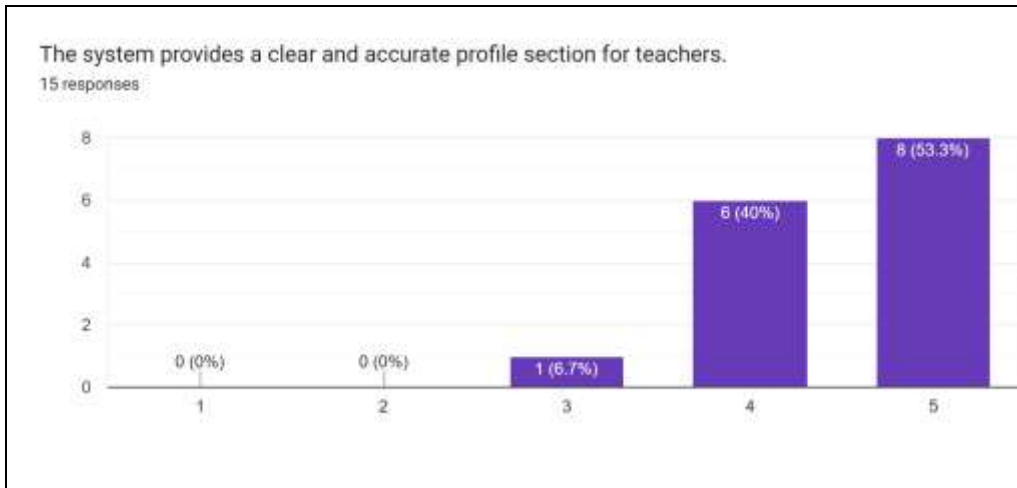


Figure 8. 23: Question 8

This figure illustrates the teacher feedback for Question 8. The majority of respondents express positive opinions about the system. This indicates that the system interface and features are suitable for teachers. Overall, the results show a high level of satisfaction.

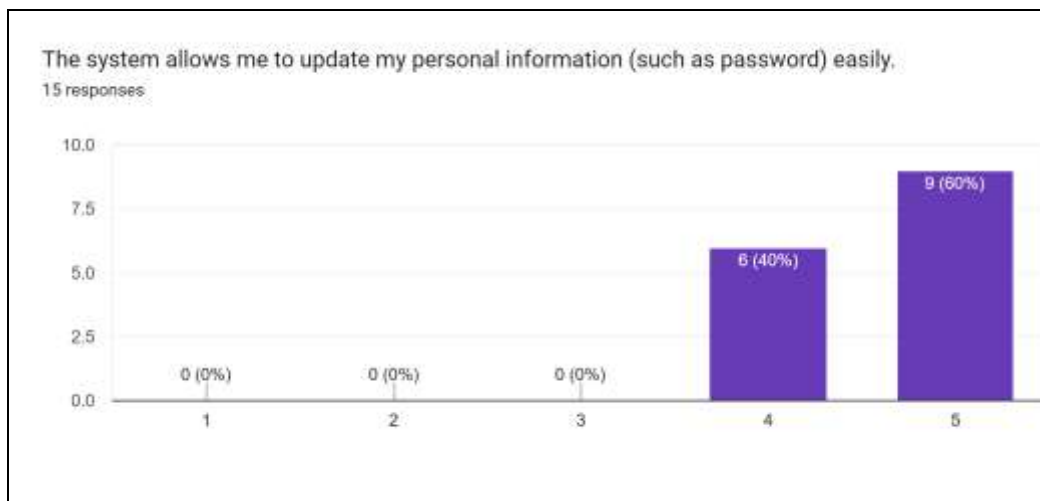


Figure 8. 24: Question 9

Figure 8.24 presents the responses from teachers regarding Question 9. Most respondents agree that the system helps improve their work efficiency. This suggests that the system is useful in supporting teaching-related tasks. The feedback reflects positive system performance.

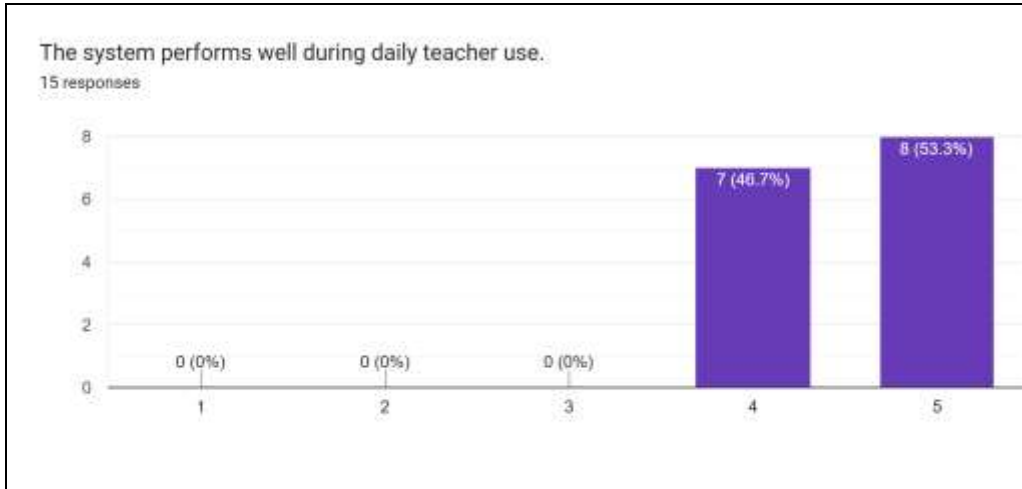


Figure 8. 25: Question 10

This figure shows the responses from teachers for Question 10. The majority of respondents provide positive evaluations of the system feature being tested. This indicates that the system meets user expectations. The system is therefore considered effective for teachers.

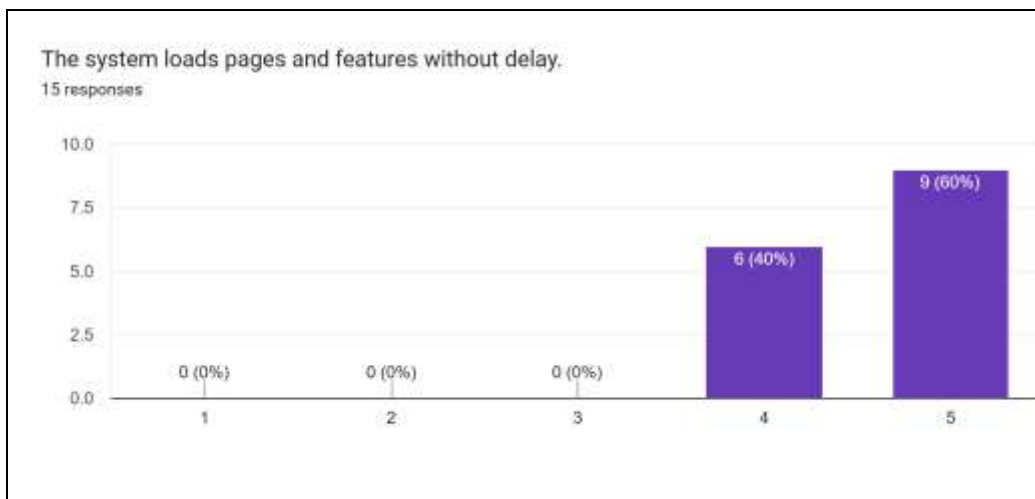


Figure 8. 26: Question 11

Figure 8.26 illustrates the teacher responses for Question 11. Most respondents agree that the system functions properly and supports their tasks. This suggests that the system is reliable and well designed. The responses show a good level of user satisfaction.

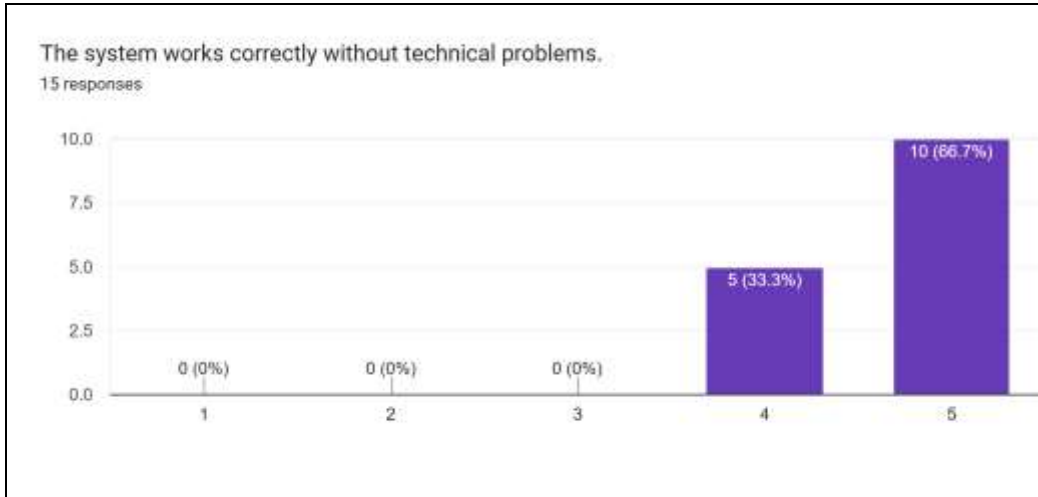


Figure 8. 27: Question 12

This figure presents the feedback from teachers regarding Question 12. The majority of respondents express agreement with the statement provided. This indicates that the system is useful and beneficial for teachers. The results reflect positive acceptance of the system.

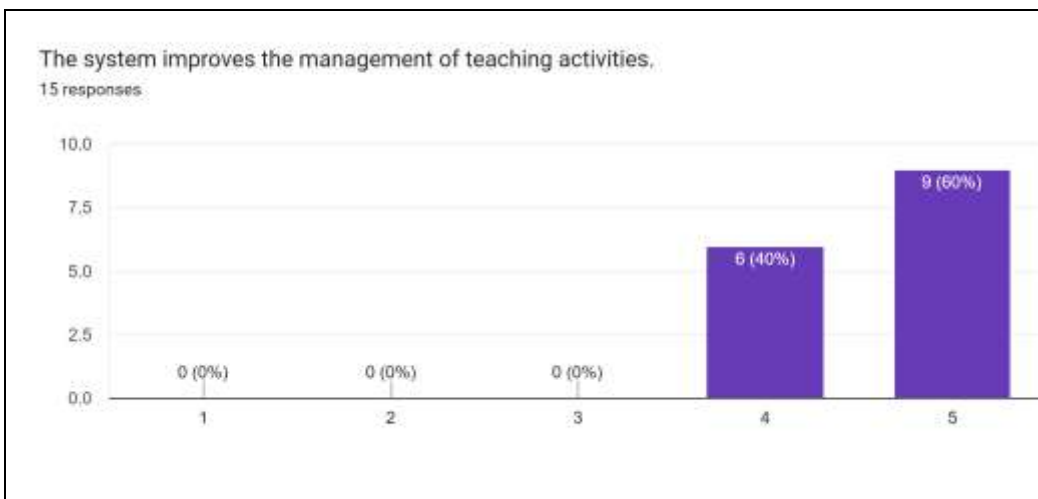


Figure 8. 28: Question 13

Figure 8.28 shows the responses from teachers for Question 13. Most respondents give positive feedback about the system functionality. This suggests that the system meets the needs of teacher users. Overall, the system receives favourable responses.

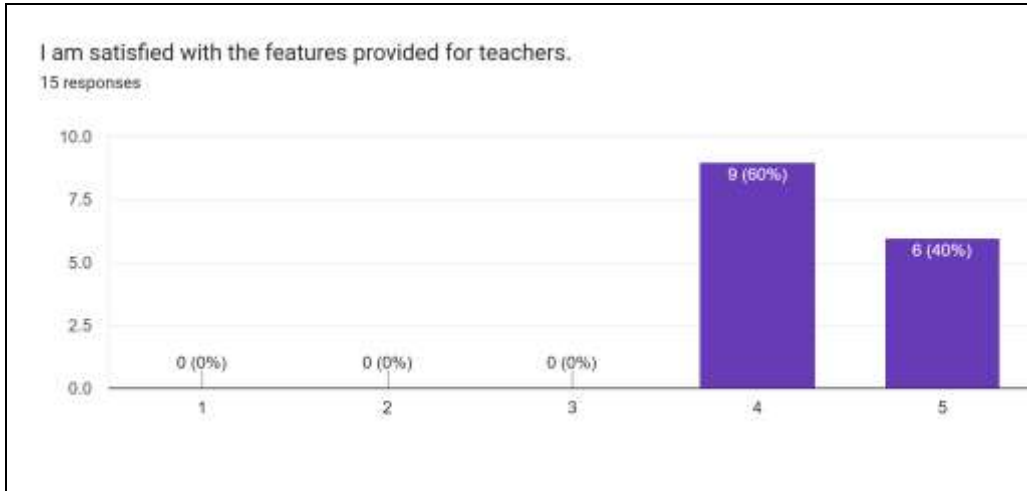


Figure 8.29: Question 14

This figure illustrates the teacher responses regarding Question 14. The results indicate that most respondents are satisfied with the system performance. This suggests that the system operates efficiently and effectively. The responses demonstrate positive user experience.

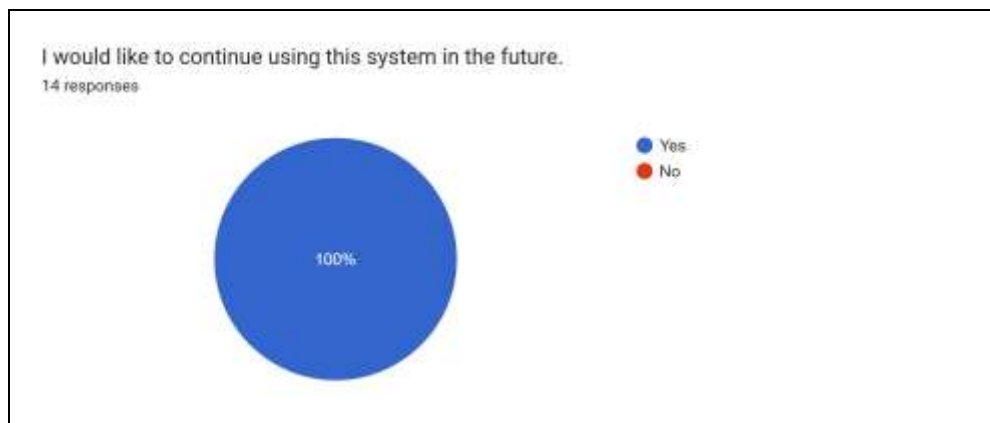


Figure 8.30: Question 15

Figure 8.30 presents the final responses from teachers in the post-test questionnaire. Most respondents provide positive feedback about the overall system. This indicates that the system is helpful and suitable for teaching purposes. Overall, the system receives a good evaluation from teacher users.

8.6 Conclusion

In conclusion, the testing phase confirmed that the Tutorial Management System operates correctly and meets the defined requirements. Various testing methods including unit testing, integration testing, system testing, and acceptance testing were conducted to evaluate system performance and reliability. The testing results indicated that the system functions properly with minimal errors. This phase ensured that the system is stable, secure, and ready for deployment.

9 PROJECT MANAGEMENT

9.1 Introduction

This chapter presents the project management aspects involved in the development of the Tuition Management System (TMS). Effective project management ensures that the project is completed within the planned timeline, scope, and resources. This chapter outlines the project schedule, work breakdown structure (WBS), Gantt chart, and risk management strategies applied throughout the project lifecycle. It also discusses how challenges were managed and how adjustments were made when necessary. Proper planning and monitoring contributed to the successful completion of the project and ensured that the objectives were achieved systematically.

9.2 Project Schedule

The project schedule outlines the timeline and major activities involved in developing the Tuition Management System. The project was divided into several phases following the SDLC with Agile approach, including requirement analysis, system design, development, testing, and documentation.

Each phase was assigned a specific duration to ensure systematic progress and timely completion. Regular monitoring was conducted to ensure that tasks were completed according to schedule. Minor adjustments were made during development to accommodate additional improvements and user feedback.

9.2.1 Work Breakdown Structure

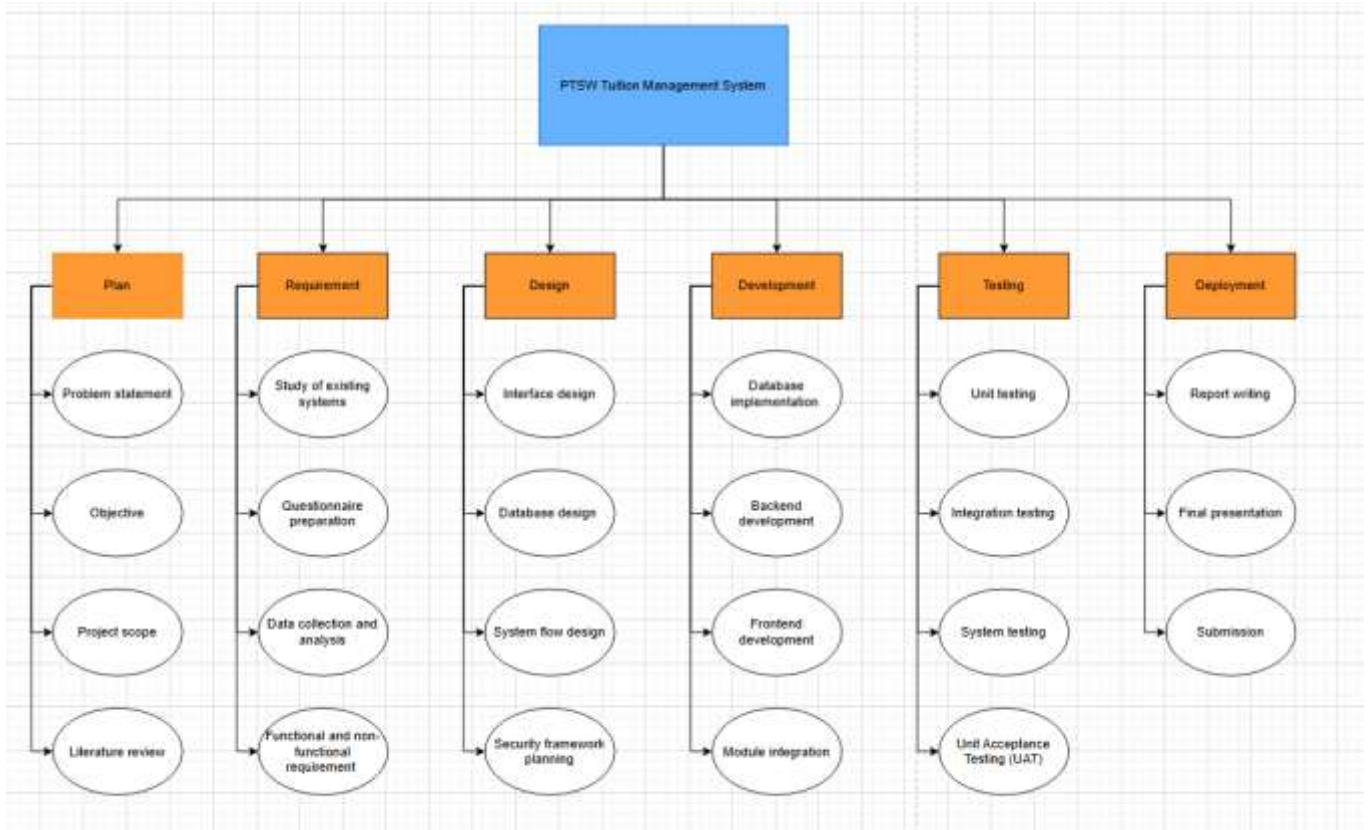


Figure 9. 1: Work Breakdown Structure

Figure 9.1 illustrates the Work Breakdown Structure (WBS) for the development of the Tuition Management System (TMS). The WBS is a project management tool used to divide the entire project into smaller and more manageable tasks. In this project, the development process is organized into several major phases such as project planning, requirement analysis, system design, system development, testing, and documentation. Each phase is further broken down into specific activities that need to be completed during the project lifecycle. This structured breakdown helps the development team clearly understand the scope of work and responsibilities for each stage. By organizing the tasks systematically, the WBS helps ensure that all project activities are properly planned and completed. It also improves project monitoring and allows easier tracking of progress throughout the system development process.

9.2.2 Gantt Chart

A Gantt chart was used to plan and monitor the development progress of the Tuition Management System (TMS) over a 28-week period. The project timeline was carefully structured into major phases, including planning, analysis, design, development, testing, and documentation, to ensure systematic and organized progress throughout the project lifecycle. Each activity was assigned a specific start and end week, allowing clear visibility of task duration and dependencies between phases. The Gantt chart also helped track overlapping tasks, such as documentation being prepared alongside development, which improved time efficiency. Regular monitoring of the schedule enabled early identification of potential delays and allowed timely adjustments to maintain progress according to the planned timeline. By using the Gantt chart as a project control tool, the development process remained structured, focused, and aligned with the overall project objectives, ensuring completion within the allocated 28 weeks.



Figure 9. 2: Gantt Chart

Figure 9.2 presents the Gantt chart used to plan and monitor the development timeline of the Tuition Management System. The Gantt chart visually represents the project schedule by showing the start and end time for each activity involved in the project. In this project, the development process was planned over a 28-week period and divided into several phases such as planning, analysis, design, development, testing, and documentation. Each activity is assigned a specific timeframe to ensure that tasks are completed in an organized and timely manner. The Gantt chart also helps identify overlapping tasks and dependencies between different phases of the project. By using this chart, the project progress can be monitored regularly to ensure that the project remains on schedule. Overall, the Gantt chart helps improve project planning, coordination, and time management during system development.

9.3 Risk Management

According to Pressman and Maxim (2020), risk management is an essential process in software development that involves identifying, analysing, and mitigating potential risks that may affect project success. During the development of the Tuition Management System (TMS), several possible risks were identified, including technical risk, scheduling risk, and data loss risk. These risks could potentially impact system functionality, project timeline, and overall quality if not properly managed.

Table 9. 1: Risk Management

Risk Identification	Description	Risk Analysis	Mitigation Plan
Technical Risk	May occur due to compatibility issues between frontend and backend components or database connection errors.	Moderate	Conduct early prototype testing and continuous debugging during development.
Scheduling Risk	Risk of delay due to underestimating development time or overlapping academic commitments.	High	Create a 28-week Gantt chart schedule and monitor weekly progress with supervisor consultation.
Data Loss Risk	Possibility of losing project files or database records due to system crash or accidental deletion.	High	Perform regular backups using cloud storage and version control (Git).
Requirement Change Risk	Changes in user requirements after development has started.	Moderate	Apply Agile approach to allow iterative updates and flexible improvements.

Despite careful planning, minor technical issues occurred during the development phase, particularly involving database configuration and local server setup. These issues were resolved through debugging and reconfiguration without significantly affecting the overall project timeline. No major scheduling delays were encountered due to consistent weekly monitoring and task prioritization. The implementation of regular data backups ensured that no critical data was permanently lost during development. Overall, proactive risk management contributed to the successful completion of the Tuition Management System within the allocated 28-week period.

9.4 Conclusion

In conclusion, effective project management contributed significantly to the successful development of the Tuition Management System (TMS). Through proper planning, structured task breakdown, scheduling with a Gantt chart, and proactive risk management, the project was completed systematically and within the allocated timeframe.

The use of Agile methodology supported flexibility and continuous improvement throughout the development process. Overall, strong project management practices ensured that the system met its objectives while maintaining quality, efficiency, and organization.

10 CONCLUSION

10.1 Introduction

This chapter presents the overall summary of the Tuition Management System (TMS) project and evaluates the outcomes achieved throughout the development process. It highlights how the project objectives were accomplished and discusses the achievements obtained from the system implementation. In addition, this chapter identifies the constraints and limitations encountered during the development phase. Finally, several recommendations and suggestions for future improvements are proposed to enhance the functionality and performance of the system in future development.

10.2 Achievement

The development of the Tuition Management System (TMS) has successfully achieved the main objectives outlined at the beginning of this project. The system was designed to improve the management of tuition-related activities such as student information management, class scheduling and payment tracking. Through the implementation of the system, administrative tasks can be handled more efficiently to traditional manual methods.

The system also provides a user-friendly interface that allow administrators and users to easily manage data and access important information. Overall, the project has demonstrated how information technology can improve operational efficiency and data organization within tuition management environments.

10.2.1 First Point of the Objective

The first objective of this project was to design and develop a web-based Tuition Management System that can efficiently manage student and teacher information. This objective was successfully achieved by implementing a centralized system that allows administrators to store, update, and retrieve student and teacher data in an organized database. The system helps reduce manual paperwork and minimizes errors in data management.

10.2.2 Second Point of the Objective

The second objective was to develop a system that can manage tuition class scheduling and information records. The implemented system allows administrators to create class schedules and manage information efficiently. This feature helps ensure better organization of tuition sessions and enables easier tracking of student participation in each class.

10.2.3 Third Point of the Objective

The third objective of the project was to develop a system that can manage payment records and generate reports. The system successfully allows administrators to record tuition payments and monitor payment status. In addition, the reporting feature enables the generation of simple reports that provide

useful insights into tuition management activities. This helps improve financial tracking and overall administrative efficiency.

10.3 Constraint and Limitation

Although the Tuition Management System was successfully developed, several constraints and limitations were encountered during the project development process. One of the main limitations was the limited development time, which restricted the ability to implement more advanced features such as automated notifications and online payment integration.

Another limitation was the reliance on a local development environment, which means the system was not fully deployed on a live server for real-world use. Additionally, the system was tested with a limited number of users, which may not fully represent the behaviour of larger-scale users. These limitations provide opportunities for further improvement in future development stages.

10.4 Future Work and Recommendation

For future development, several improvements can be made to enhance the functionality and usability of the Tuition Management System. One possible improvement is the integration of an online payment gateway to allow students or parents to make tuition payments directly through the system.

Another recommendation is the implementation of automated notifications through email or mobile applications to inform students about upcoming classes, schedule changes, or payment reminders. In addition, the system could be further improved by developing a mobile application version to allow users to access the system more conveniently using smartphones.

Enhancing system security features and improving the user interface design are also recommended to provide a better user experience and ensure data protection.

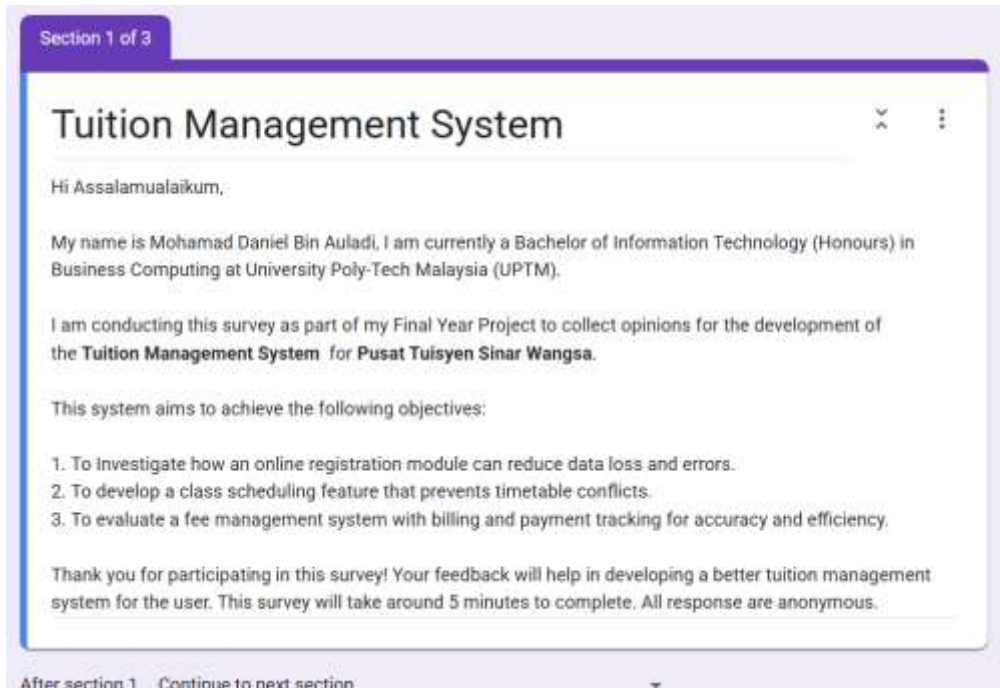
10.5 Conclusion

In conclusion, the Tuition Management System (TMS) was successfully designed and developed to improve the management of tuition-related activities. The system provides an organized platform for managing student information, scheduling classes, and recording payment transactions. The implementation of this system demonstrates how digital solutions can improve efficiency, accuracy, and accessibility compared to traditional manual methods.

Despite some limitations during development, the system achieved its intended objectives and provided valuable experience in system analysis, design, and implementation. With further enhancements and improvements, the Tuition Management System has the potential to become a more comprehensive solution for managing tuition center operations in the future.

Appendix A – Requirements Specification Document

Requirement Analysis – Questionnaire



Post-Test Analysis – Questionnaire

Admin

The system is easy to access and log in as an administrator. *	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system interface is clear and user-friendly. *	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
I can navigate through the admin modules without difficulty. *	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system functions are well organized for admin tasks. *	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

The system allows me to manage teacher accounts effectively. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system allows me to manage student records efficiently. *

1 2 3 4 5

Strongly Disagree Strongly Agree

I can update tuition class schedules easily using the system. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system supports accurate management of fee payments. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The reporting feature provides useful administrative information. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system responds quickly when performing admin operations. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system runs smoothly without frequent errors or crashes. *

1 2 3 4 5

Strongly Disagree Strongly Agree

Data entered into the system is saved correctly. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system provides secure access for administrators. *

1 2 3 4 5

Strongly Disagree Strongly Agree

I am satisfied with the overall performance of the system. *

1 2 3 4 5

Strongly Disagree Strongly Agree

I would recommend this system for managing tuition operations.

1 2 3 4 5

Strongly Disagree Strongly Agree

Teacher

The system is easy for teachers to log in and access. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system interface is simple and understandable. *

1 2 3 4 5

Strongly Disagree Strongly Agree

I can easily navigate through the teacher dashboard. *

1 2 3 4 5

Strongly Disagree Strongly Agree

The system helps me complete my teaching tasks efficiently. *

1 2 3 4 5

Strongly Disagree Strongly Agree

I can view my class schedule clearly in the system. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system helps reduce the amount of manual paperwork in managing teaching activities. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
I can update student performance or progress easily. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system provides a clear and accurate profile section for teachers. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system allows me to update my personal information (such as password) easily. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

The system performs well during daily teacher use. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system loads pages and features without delay. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree
The system works correctly without technical problems. *						
	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

The system improves the management of teaching activities.

1 2 3 4 5

Strongly Disagree Strongly Agree

...

I am satisfied with the features provided for teachers.

1 2 3 4 5

Strongly Disagree Strongly Agree

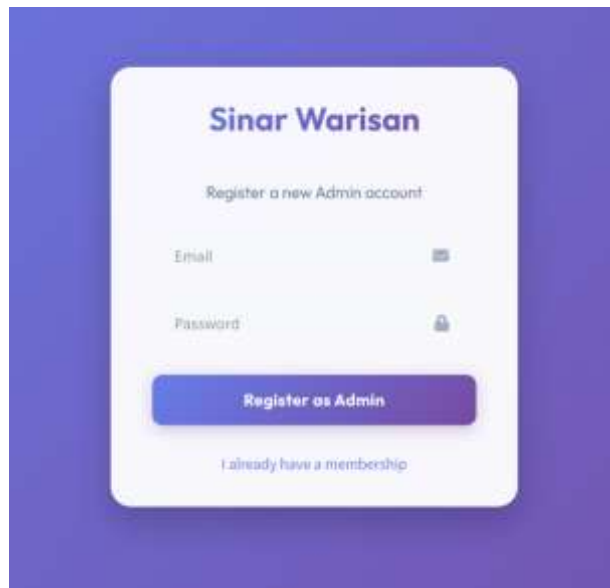
I would like to continue using this system in the future.

Yes

No

Appendix B – User manual

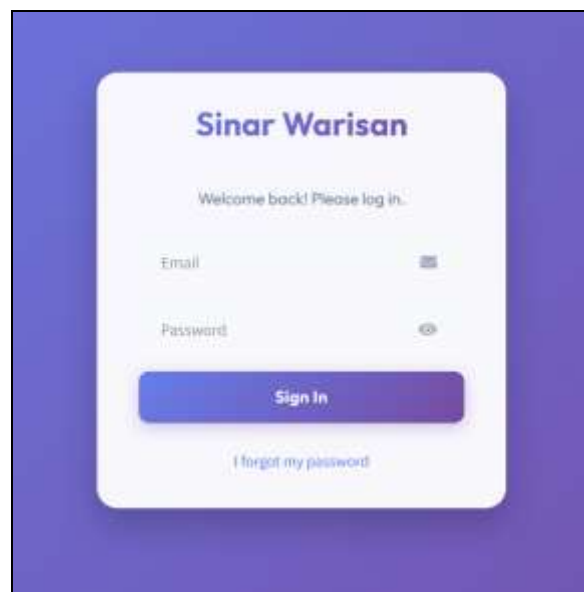
Register Admin



The screenshot shows a registration form for an Admin account on the Sinar Warisan platform. The form is titled "Sinar Warisan" and includes the instruction "Register a new Admin account". It features two input fields: "Email" and "Password", each with a corresponding icon (an envelope for email and a lock for password). Below the fields is a prominent blue button labeled "Register as Admin". At the bottom of the form, there is a link that says "I already have a membership".

- We will provide a separate URL link only to clients so they can register an account as admin.
- Click Register as Admin and the account will be created.

Log in Admin



The screenshot shows a login form for an Admin account on the Sinar Warisan platform. The form is titled "Sinar Warisan" and includes the instruction "Welcome back! Please log in.". It features two input fields: "Email" and "Password", each with a corresponding icon (an envelope for email and an eye for password). Below the fields is a prominent blue button labeled "Sign In". At the bottom of the form, there is a link that says "I forgot my password".

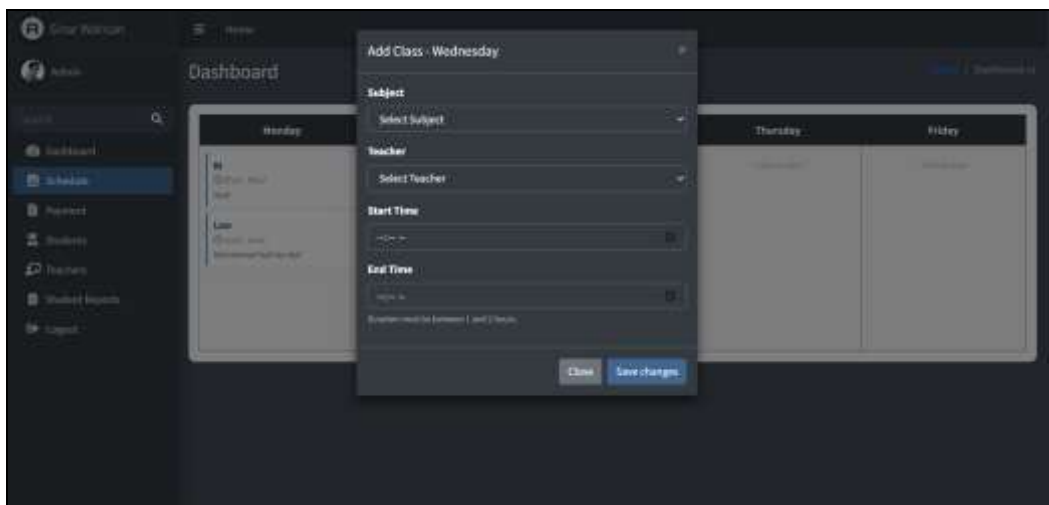
- You must enter the email and password that you registered as admin earlier.
- And then click sign in.

Admin Dashboard

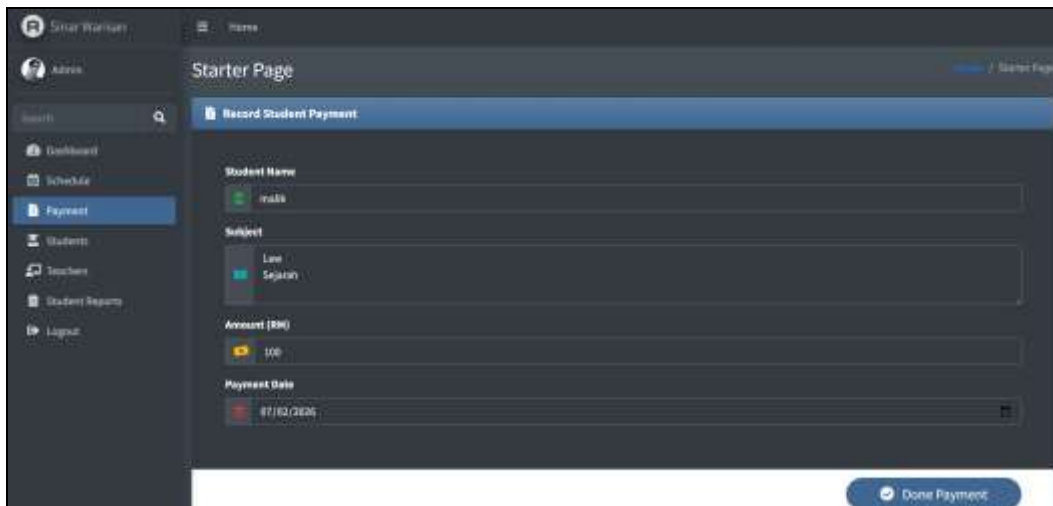


- This is the admin dashboard which contains information such as total teachers, total students, total subjects and also total income.

Admin Schedule

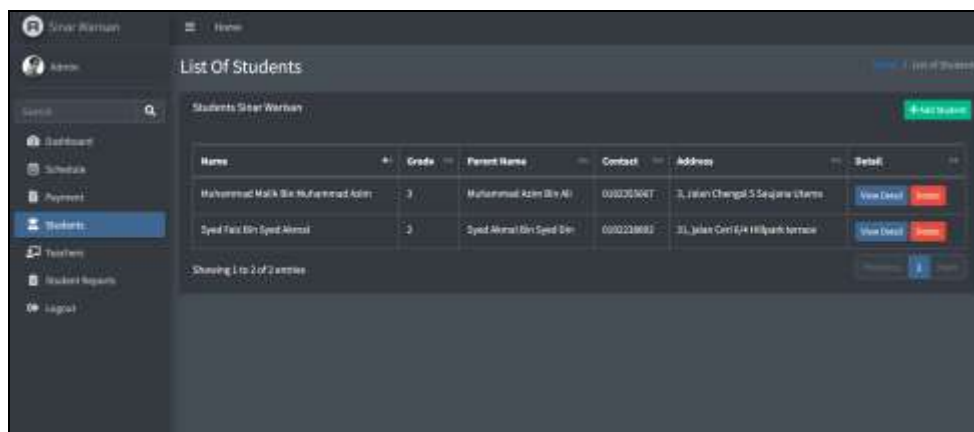


- For schedule, click on that day and fill in the class start and end times.
- You can edit and change the time, and you also can delete if the schedule is no longer used.

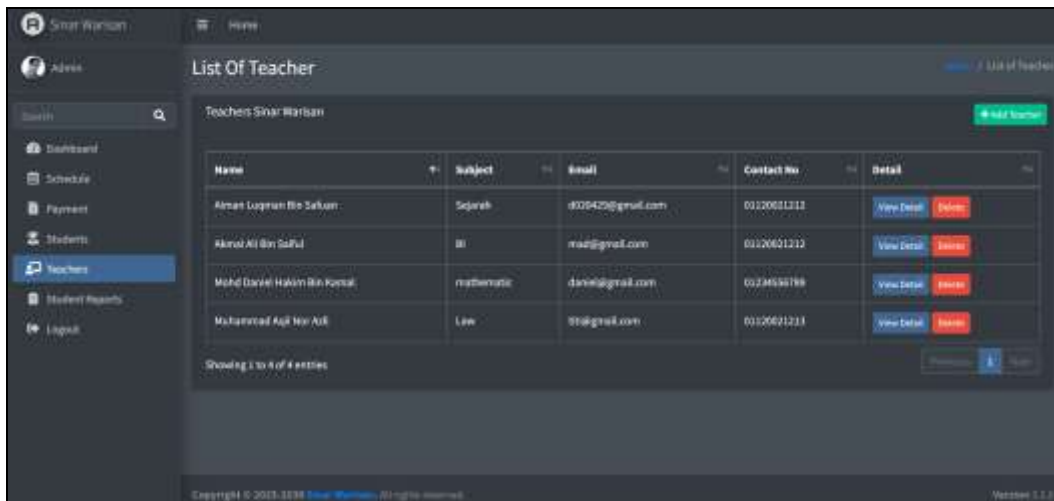


- You must fill in the student's name, and the system will directly issue the total amount that must be paid.
- Click done payment and the system will generate invoice.

Admin Student

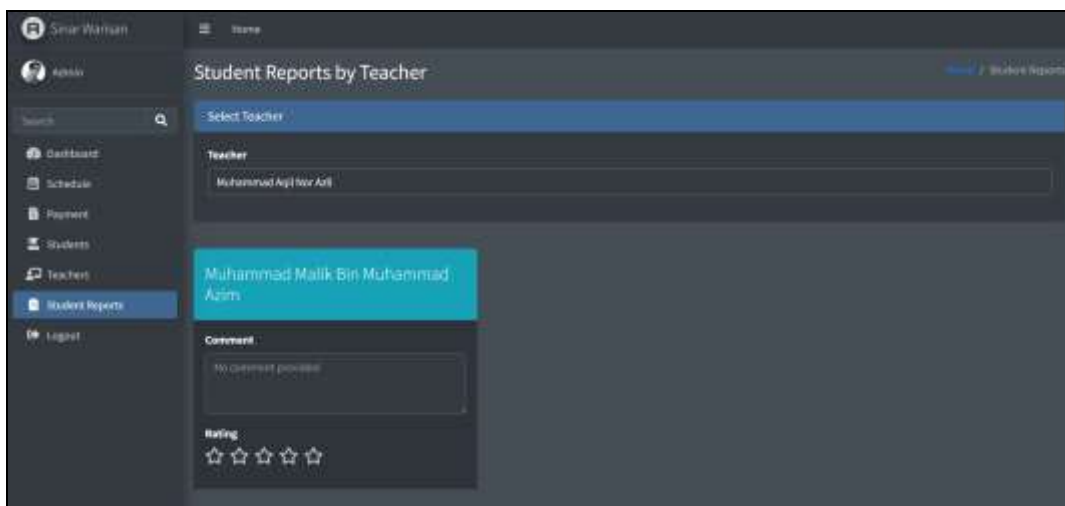


- You can view student details and edit student information.
- If you want to add student, you must click the add student button and fill in the details and click save.

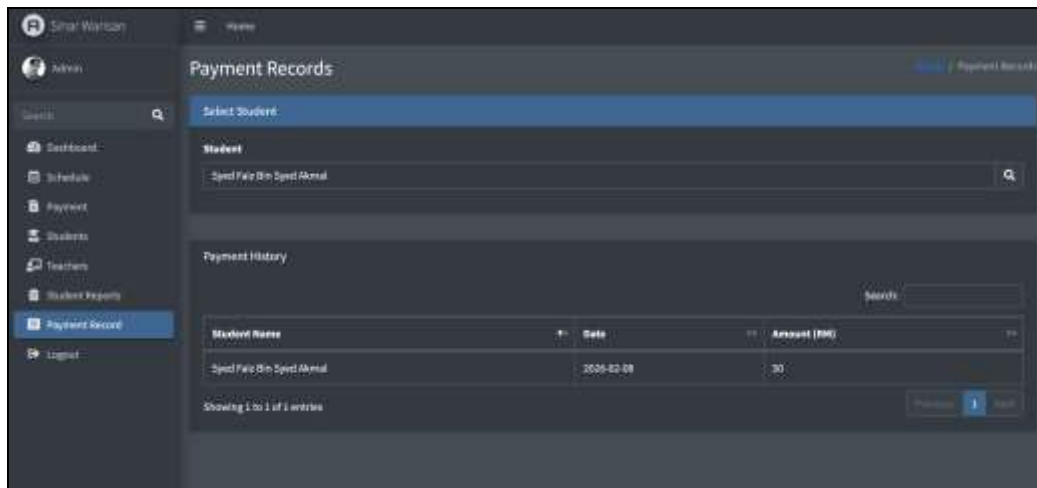


- You can view teacher details and edit teacher information.
- If you want to add teacher, you must click the add teacher button and fill in the details and click save.

Admin Student Report

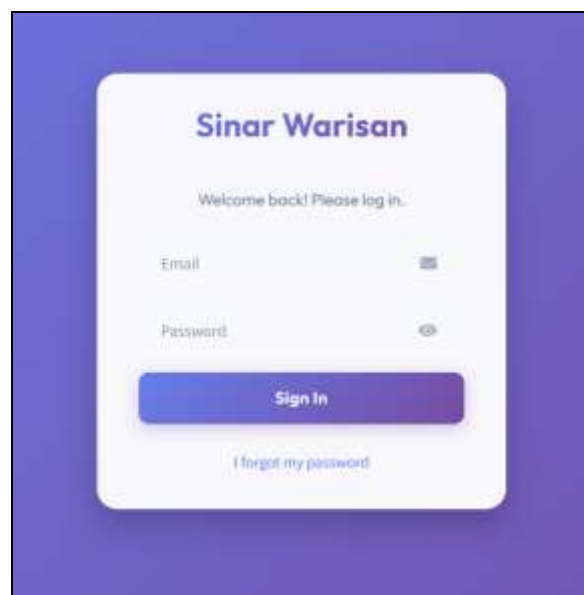


- This page allows the administrator to view student performance records.
- Admins have to search teacher and then the list of the student performance will be shown.



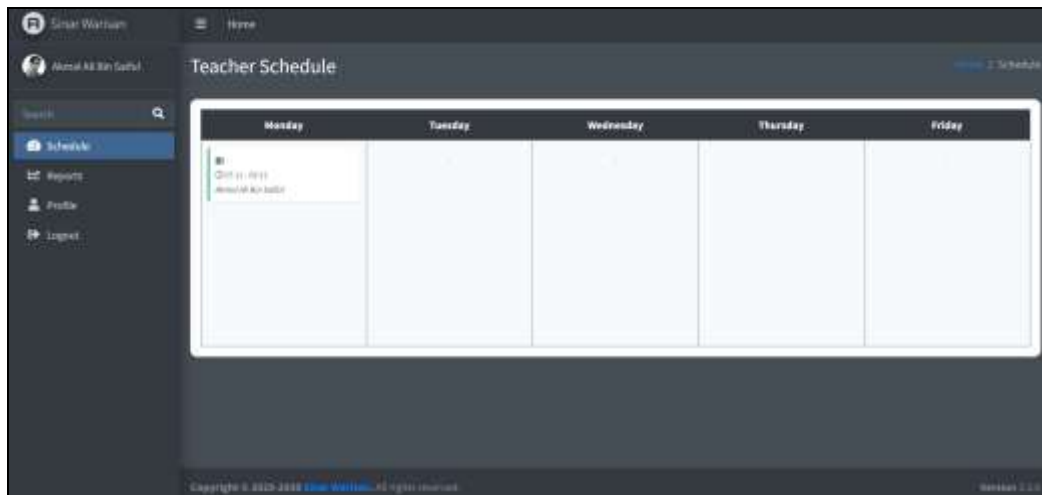
- Admin can view the records of the students who have made payment transactions.
- You have to search the students name, and the payment history will be appeared.

Login Teacher



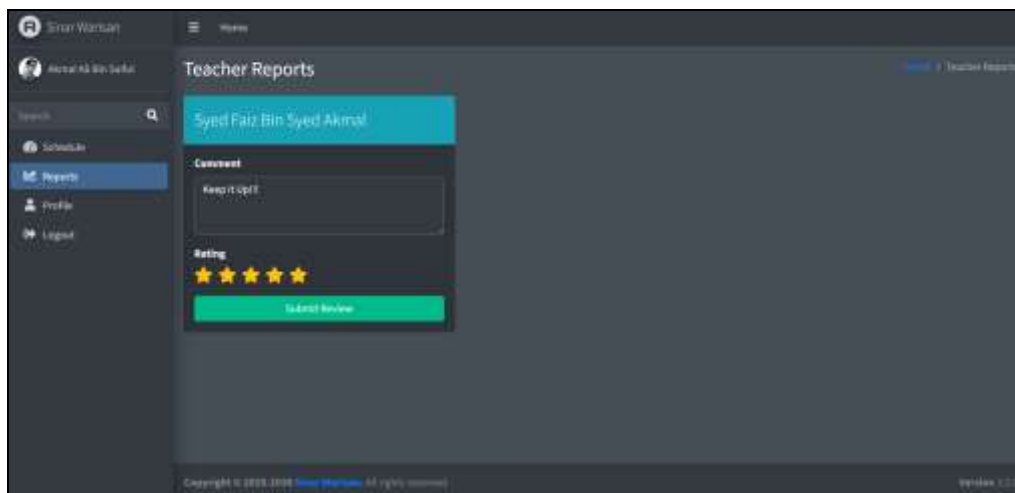
- Teacher needs to enter that admin have registered.
- The password to login is default to '123'

Teacher Schedule



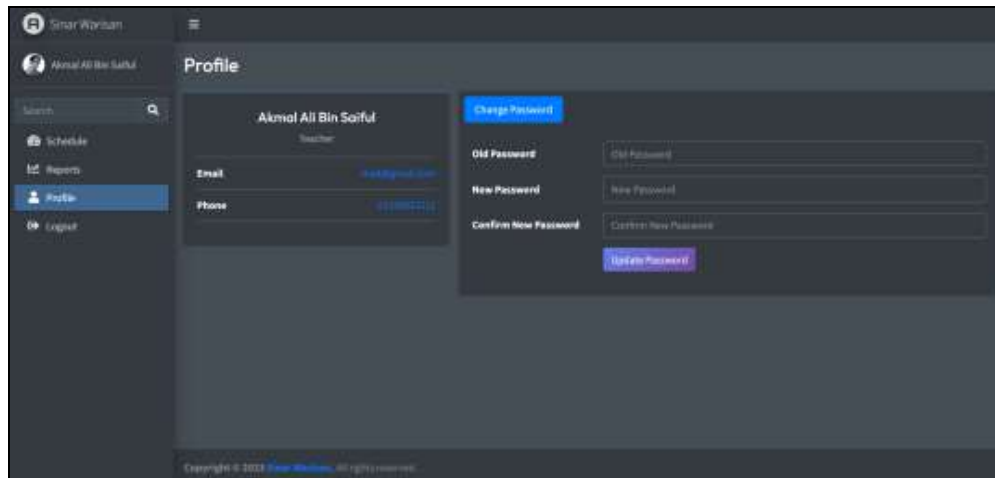
- In this section, teachers can see what days and times they teach.

Teacher Reports



- The teacher can provide comments and also marks based on the student's performance

Teacher Profile



- Teacher can change the password that given earlier.
- They must put old password and then create new password.
- After that click 'Update Password'



15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- 120 Not Cited or Quoted 14%**
Matches with neither in-text citation nor quotation marks
- 4 Missing Quotations 0%**
Matches that are still very similar to source material
- 1 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 5%** Internet sources
- 2%** Publications
- 14%** Submitted works (Student Papers)



*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.



Appendix D – Log Book

FYP 1

CT203/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN BUSINESS COMPUTING



FACULTY OF COMPUTING & MULTIMEDIA (FCOM)

BUSINESS COMPUTING PROJECT 1
(FYP4094)

LOG BOOK

STUDENT'S NAME : MOHAMAD DANIEL BIN AULADI

ID NO. : AM2311015219

SUPERVISOR : MISS NUR AFINI NATRAH BT MOHD ASHRIL

PROJECT TITLE : TUITION MANAGEMENT SYSTEM

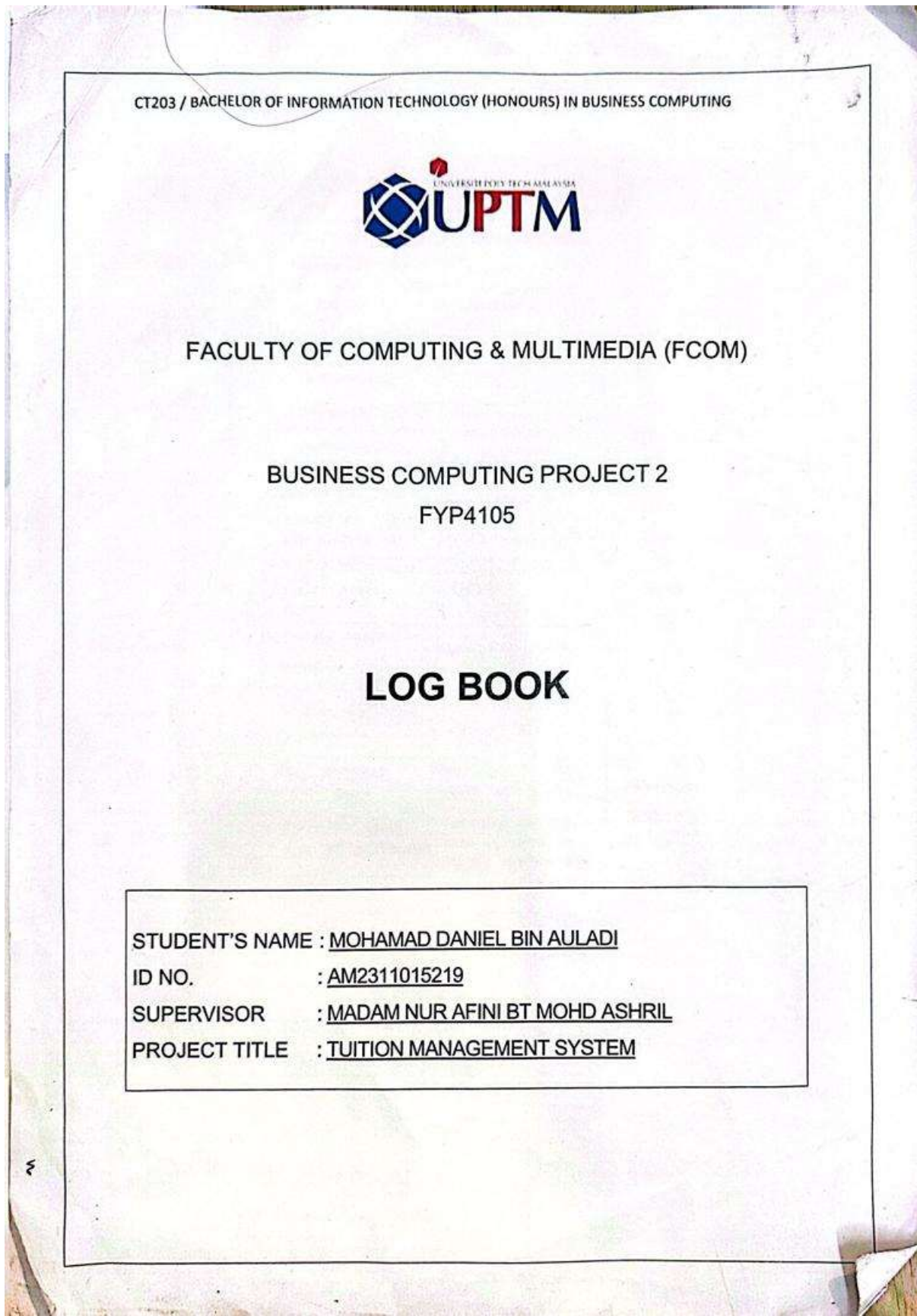
CT203/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN BUSINESS COMPUTING

Date/ Week		Agenda	Next Agenda	Signature (Supervisor)
5/8/2025	1	First class with Madam Noraliza and introduction about the course	Find supervisor and title for fyp	<i>afw</i> 5/8/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
12/8/2025	2	Find supervisor and title for fyp	Finalize title with supervisor	<i>afw</i> 12/8/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
19/8/2025	3	Finalize title with supervisor	Prepare project proposal	<i>afw</i> 26/8/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
26/8/2025	4	Consultation for project proposal with supervisor	Submit project proposal	<i>afw</i> 26/8/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
2/9/2025	5	Preparation for chapter 1 for FYP 1 report	Complete chapter 1 report writing	<i>afw</i> 2/9/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
9/9/2025	6	Preparation draft for chapter 2	Complete chapter 2 report writing	<i>afw</i> 9/9/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
23/9/2025	7	Consultation for chapter 1 and chapter 2 with supervisor	Prepare a draft for chapter 3	<i>afw</i> 23/9/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
30/9/2025	8	Complete chapter 3 for report writing	Prepare a draft for chapter 4	<i>afw</i> 30/9/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
7/10/2025	9	Distribute questionnaire to gain user requirements	Consultation for questionnaire with supervisor	<i>afw</i> 7/10/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
14/10/2025	10	Complete chapter 4 for report writing	Prepare a draft for chapter 5 and analyze questionnaire	<i>afw</i> 14/10/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
21/10/2025	11	Finishes chapter 5 for report writing	Create mockup design using Pencil for FYP 1	<i>afw</i> 21/10/2025 NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia

CT203/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN BUSINESS COMPUTING

29/10/2025	12	Complete mockup design for FYP 1	Consultation for report writing and mockup design with supervisor	<i>af.w 29/10/2025</i> NUR AFINI NATIAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
3/11/2025	13	Create a slide presentation for FYP 1	Last consultation with supervisor for presentation FYP 1	<i>af.w 3/11/2025</i> NUR AFINI NATIAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia
10/11/2025	14	Presentation FYP 1	Report submission	<i>af.w 10/11/2025</i> NUR AFINI NATIAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia

FYP 2



CT203 / BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN BUSINESS COMPUTING

Week	Agenda	Next Agenda	Signature (Supervisor / Coordinator)
22.12.2025	1 Project briefing & planning. Introduction for FYP 2	Begin UI/UX design & database planning	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 13/1/2026
29.12.2025	2 Prepare wireframe of system pages for admin and teachers	Continuing interface design and Setup database connection	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 13/1/2026
9.01.2026	3 Consultation with supervisor about the implementation	Prepare a draft report for chapter 6	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 13/1/2026
13.01.2026	4 Continue the system development and consult with supervisor	Finish writing chapter 6 and prepare draft for chapter 7	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 13/1/2026
20.01.2026	5 Continue the system development & consult with supervisor	check research objective & post-KJ questionnaire questions	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 20/1/2026
27.01.2026	6 -Updated post-develop question -poster draft -Report	-continue develop system	NUR AFINI NATRAH BINTI MOHD ASHRIL Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia 27/1/2026
3.02.2026	7 complete system demo	complete system demo	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 3/2/2026
10.02.2026	8 complete system demo	video complete system dem	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 10/2/2026
17.02.2026	9 video complete system demo	video system demo & post-test questionnaire	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 17/2/2026
26.02.2026	10 video system demo & post-test questionnaire	-complete the report -finalised demo video -submit acknowledgment form.	NUR AFINI NATRAH BINTI MOHD ASHRIL Lecturer Faculty of Computing and Multimedia Universiti Poly-Tech Malaysia 26/2/2026
2.03.2026	11 -complete the report -show post-test analysis	-send report to supervisor -submit acknowledgment form.	NUR AFINI NATRAH BINTI MOHD ASHRIL Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia 2/3/2026
11.03.2026	12 -send report to su -submit acknowledgment form.	-mock presentation	NUR AFINI NATRAH BINTI MOHD ASHRIL Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia 11/3/2026
16.03.2026	13 -mock presentation	FYP 2 Presentation with Examiner	NUR AFINI NATRAH BINTI MOHD ASHRIL Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia 16/3/2026
30.03.2026	14 Presentation FYP 2	Report Submission	NUR AFINI NATRAH BINTI MOHD ASHRIL Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia 30/3/2026

References

1. Arias, J., Macatulad, N. and Malaluan, J. (2021). *Digital Transformation in Educational Administration: Challenges and Solutions*. Journal of Educational Technology, 18(4), pp.122–130.
Available at: <https://doi.org/10.1016/j.jedutech.2021.04.007>
2. Nair, R. and Saini, V. (2020). *Improving Record Management Systems in Private Institutions*. International Journal of Computer Applications, 176(36), pp.10–15.
Available at: <https://www.ijcaonline.org/archives/volume176/number36/nair2020.pdf>
3. Mohanty, P. and Dash, S. (2022). *Data Handling and Security in School Information Systems*. International Journal of Information Systems and Technology, 14(2), pp.55–67.
Available at: <https://ijst.org/articles/2022/02/mohanty-dash-data-systems.pdf>
4. Sahu, R., Singh, K. and Jena, A. (2020). *Enhancing Educational Efficiency through Centralized Student Information Systems*. Education and Information Technologies, 25(6), pp.5299–5312.
Available at: <https://doi.org/10.1007/s10639-020-10189-9>
5. Gupta, A. and Sharma, D. (2022). *Role-Based Authentication in Academic Systems*. Journal of Computer Science and Engineering, 20(3), pp.45–51.
Available at: <https://jcse.org/2022/03/role-based-authentication.pdf>
6. Rahman, N., Ali, M. and Yusof, M. (2019). *Timetable Automation for Educational Institutions*. International Journal of Advanced Computer Science, 10(5), pp.124–132.
Available at: <https://doi.org/10.14569/IJACSA.2019.0100518>
7. Chavda, H. and Solanki, S. (2021). *Smart Scheduling Systems for Education Sector*. Journal of Information Systems, 19(2), pp.63–72.
Available at: <https://doi.org/10.1007/s10257-021-00519-6>
8. Kaur, G. and Kaur, J. (2021). *Algorithmic Scheduling for Academic Institutions*. International Journal of Innovative Computing, 7(1), pp.22–29.
Available at: <https://ijic.org/articles/academic-scheduling-2021.pdf>
9. Alam, M. and Ahmed, R. (2020). *Automation in Financial Record Management in Education*. Journal of Accounting and Information Systems, 16(4), pp.89–96.
Available at: <https://doi.org/10.1108/JAIS-2020-0164>
10. Yusof, Z., Rahim, S. and Khalid, H. (2021). *Challenges in Manual Financial Record-Keeping in Private Tuition Centres*. Asian Journal of Business and Management, 9(1), pp.77–85.
Available at: <https://ajbm.org/articles/manual-record-keeping-education.pdf>
11. Asim, A., Khan, N. and Ahmad, M. (2021). *Automated Billing in Educational Systems: A Case Study*. International Journal of Computer and Information Research, 19(3), pp.42–50.
Available at: <https://ijcir.org/articles/2021-03/automated-billing.pdf>

12. Patil, S. and Dhamale, P. (2022). *Enhancing Transparency in Educational Payments through Automation*. International Journal of Computer Trends and Technology, 70(8), pp.12–18.
Available at: <https://doi.org/10.14445/22312803/IJCTT-V70I8P203>
13. Adebayo, B. and Ncube, M. (2021). *Automation and Transparency in Financial Processes: A Case Study of FACTS TMS*. Journal of Financial Technology, 9(4), pp.115–121.
Available at: <https://journals.fintech.org/facts-tms.pdf>
14. Pressman, R.S. and Maxim, B.R. (2020). *Software Engineering: A Practitioner's Approach*. 9th ed. New York: McGraw-Hill.
Available at: <https://www.mheducation.com/highered/product/software-engineering-practitioners-approach-pressman-maxim/M9780078022128.html>
15. Sommerville, I. (2020). *Software Engineering*. 10th ed. Harlow: Pearson Education.
Available at: <https://www.pearson.com/en-gb/subject-catalog/p/software-engineering/P200000003578/9780137035151>
16. Dennis, A., Wixom, B. and Tegarden, D. (2021). *Systems Analysis and Design: An Object-Oriented Approach with UML*. 6th ed. New York: John Wiley & Sons.
Available at: <https://www.wiley.com/en-us/Systems+Analysis+and+Design%2C+6th+Edition-p-9781119559917>
17. Beck, K., Beedle, M., Bennekum, A. van et al. (2001). *Manifesto for Agile Software Development*.
Available at: <https://agilemanifesto.org/>
18. Highsmith, J. (2020). *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. 2nd ed. Addison-Wesley.
Available at: <https://www.pearson.com/en-us/subject-catalog/p/adaptive-software-development/P100002302318>
19. FACTS Management. (n.d.). *FACTS Tuition Management System*.
Available at: <https://factsmgt.com/>
20. Blackbaud. (n.d.). *Blackbaud Tuition Management*.
Available at: <https://www.blackbaud.com/products/tuition-management>
21. openSIS. (n.d.). *Open Source Student Information System*.
Available at: <https://opensis.com/>
22. Arctiq. (n.d.). *Technology Solutions for Education*.
Available at: <https://arctiq.ca/>
23. OS4ED. (2023). *openSIS School Management System*.
Available at: <https://www.os4ed.com/>
24. Marketing91. (n.d.). *Agile Methodology Explained*.
Available at: <https://www.marketing91.com/agile-methodology/>

GitHub

Repository Link: <https://github.com/mdddaniel/tsm-backend> (Backend)
<https://github.com/mdddaniel/tsm-frontend> (Frontend)

Youtube

Demo Video Link: <https://youtu.be/Uut3hRQIAh8?feature=shared>