

Development of Pusat Tuisyen Pilihan Tepat Management System

Kwa Yee Zhen¹, Noor Zuraidin Mohd Safar^{1*}

¹Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: <https://doi.org/10.30880/aitcs.2022.03.02.032>

Received 14 June 2022; Accepted 26 September 2022; Available online 30 November 2022

Abstract: As a famous Tuition Centre, Pusat Tuisyen Pilihan Tepat had encountered two main problems, which are the inefficient manual management system and lack of platform for viewing the latest notice and information. Thus, this study is conducted with the objective to design, develop, and test an efficient web-based Tuition Centre Management System with the purpose of improving the management process of Pusat Tuisyen Pilihan Tepat. The Agile Model is used as the methodology to conduct the study and PHP Programming Language is used to develop the proposed system. The functional requirements, non-functional requirements, and UML diagrams of the proposed system will be identified. As a result, a web-based Tuition Centre management system named Pusat Tuisyen Pilihan Tepat Management System will be developed successfully so that the problems encountered can be resolved. With the developed proposed system, the management process of the Tuition Centre can be improved due to its efficiency.

Keywords: Tuition Centre Management System, Web-based System, PHP Programming Language

1. Introduction

The Tuition Centre, which is a private school or educational institution, plays a critical role among students especially candidates by guiding them to improve their academic performance. Nowadays, due to the efficiency of the advanced technology in reducing the time required to manage and monitor the daily operations of educational institution [1], a management system is necessary to implement in a Tuition Centre. However, most of the Tuition Centre in Malaysia generally used manual record management system to store and manage the information of the students and tutors [2]. The Pusat Tuisyen Pilihan Tepat, which is a famous Tuition Centre at Kuala Lumpur also implement manual record management system as the method to manage the information of students and tutors. All personal information of students and tutors will be recorded in papers and keep in folders.

Based on the situation above, the first and the most lethal problem is caused by the manual system of Pusat Tuisyen Pilihan Tepat itself. Although recording information by using paper is the most traditional method, it has various drawbacks, including being difficult to track and manage [3].

*Corresponding author: zuraidin@uthm.edu.my

2022 UTHM Publisher. All rights reserved.

publisher.uthm.edu.my/periodicals/index.php/aitcs

Therefore, the manual system implemented by Pusat Tuisyen Pilihan Tepat is not secure, time consuming, and inefficient as it will be possible that the loss of important information of students and tutors might occur due to the missing of papers or folders. Next, the second problem is the lack of a platform to view the latest notice or information of the Pusat Tuisyen Pilihan Tepat due to the impact of COVID-19. Pusat Tuisyen Pilihan Tepat had decided to conduct online lessons to avoid the spreading of COVID-19. Due to this situation, both students and tutors have no platform to view the latest notice or information such as the class information or important announcement. This may cause the students and tutors unable to follow up to the latest information of Pusat Tuisyen Pilihan Tepat. To overcome the problems encountered by Pusat Tuisyen Pilihan Tepat, a web-based Tuition Centre management system named Pusat Tuisyen Pilihan Tepat Management System has been developed.

The objective of this study is to design a web-based Tuition Centre management system that can enhance the management process of Pusat Tuisyen Pilihan Tepat, to develop an efficient Tuition Centre management system that acts as the platform for tutors and students to follow up with the latest notice or information and conduct the test on the developed system. The proposed system will be developed based on the concept of school management system as the Tuition Centre is a type of school. The administrator (who is the owner), tutors, and students of Pusat Tuisyen Pilihan Tepat will be the targeted users of the proposed system. The modules that included in scope of the proposed system will be the user login, user management, class management, notice management, and notes management modules. The result of the proposed system should be a proper Tuition Centre management system which able to provide its functionalities and services to the users without any bugs or defects.

Besides this chapter, this paper should consist other chapters as well. Chapter 2 of this paper should include the related work of the proposed system which consists of the related terms, concepts, and the comparison between the existing systems and the proposed system. Whereas in Chapter 3, the methodology used to conduct this study will be discussed based on different phases. Furthermore, the results and discussion of this paper should be stated in Chapter 4. Lastly, the conclusion of this paper will be included in Chapter 5.

2. Related Work

In this chapter, the related work of the proposed system will be discussed. The related terms and concepts of the proposed system, which are school management system and Tuition Centre management system will be included in this chapter. Next, the comparison of the existing systems with the proposed system will be shown.

2.1 School management system

The proposed system will be developed based on the concept of school management system. A school management system, also as known as student information system or school management software is an administration tool with the purpose of automating the management task of educational institutions [4] such as universities, Tuition Centre, and high schools. Educational institutions are recommended to implement the school management system as it provides efficient feature which can ease the task of student information management, class management, and streamlined record keeping.

2.2 Tuition Centre management system

Basically, the concept of the Tuition Centre management system is developed based on the concept of the school management system. A Tuition Centre management system is a web-based management system that can assist in the administration process of a Tuition Centre [4].

A Tuition Centre management system will provide the functionalities that can effectively manage the information of users such as administrator, student, and tutor, as well as the task of managing the day-to-day operations of a Tuition Centre. The staff or the owner of the Tuition Centre will take the

role as the administrator who take full responsibility on managing the detail and information of student and tutor. [5]. The use of the Tuition Centre management system can replace the manual management system which is using the paper by providing the functionality to store, manage, and manipulate the data of the users.

2.3 Comparison between existing systems and the proposed system

Table 1: Comparison between existing systems and proposed system

Features	Glexis Tuition Centre Management System	A ONE EDUCATION Management System	SimTrain School Management System	Proposed System
Web-based	X	✓	✓	✓
Administrator able to add new student and tutor	✓	✓	✓	✓
Administrator able to manage all user's profile	✓	✓	✓	✓
Administrator able to login and logout	X	✓	✓	✓
Tutor able to manage profile	X	✓	X	✓
Tutor able to login and logout	X	✓	X	✓
Students able to manage profile	X	X	X	✓
Students able to login and logout	X	✓	X	✓
Administrator able to CRUD notice	X	X	✓	✓
Administrator able to CRUD class	X	X	✓	✓
Tutor able to manage notes	X	X	X	✓
Student able to download notes	X	X	X	✓
Fee payment management	✓	✓	✓	X
Parents able to view children information and performance	X	✓	✓	X
Parents able to login and logout	X	✓	✓	X

3. Methodology/Framework

The Agile Model is selected as the methodology to conduct this study and the development of the proposed system. Agile Model defines the latest software development methodology which is based on the combination of iterative and incremental development. Collaboration between the self-organizing and cross-functional teams will result in the requirements and the solutions in Agile Model [6]. There are five main phases involve in the Agile Model, which are planning phase, analysis phase, design

phase, implementation phase, and testing phase. These completed main phases are defined as an iteration and each iteration will be followed up by a new iteration if there are any further updates.

3.1 Planning phase

Planning Phase is the very first main phase in Agile Model. In this phase, the problem statements of the study are identified based on the problem faced by Pusat Tuisyen Pilihan Tepat. The objectives, scope, expected result, and significance of the proposed system are then established. Discussion with the owner of Pusat Tuisyen Pilihan Tepat will be conducted, and the important information are gathered to determine the requirements of the proposed system. Further research is conducted to acquire related information. To collect the information of the similar existing systems, interview with several Tuition Centre's owners and literature review will be conducted. A Gantt Chart is generated with the purpose to provide better visibility on the project development timeline. The Gantt Chart will be attached to Appendix A.

3.2 Analysis phase

The purpose of the analysis phase is to determine the requirements of the proposed system, which are the functional requirements and non-functional requirements. Analysis on the manual management system implemented by Pusat Tuisyen Pilihan Tepat is conducted to determine the functional requirements that can improve the management process. During this phase, the type of user, modules, technology, and functionalities of each existing Tuition Centre management systems are analyzed. The functional requirements and non-functional requirements which suitable for the proposed system are determined based on the analysis of the existing systems.

The functional requirements of the proposed system will be shown in the Table 2.

Table 2: Functional requirements of the proposed system

Modules	Functionalities
User Login	<ol style="list-style-type: none"> 1. All users can login to their account with correct email and password and logout from the system. 2. Error message will display if the email and password is incorrect.
User Management	<ol style="list-style-type: none"> 1. Administrator is able to add new users or delete users. 2. All users are display in the user list. 3. All users are able to manage their own profile and upload profile picture. Only administrator can manage all user's profile.
Class Management	<ol style="list-style-type: none"> 1. Administrator can add class or delete class. Administrator can also assign or remove tutor from the class. 2. Administrator and tutor can edit the class and add or remove students from the class.
Notes Management	<ol style="list-style-type: none"> 1. Administrator and tutor can manage the note folder. 2. Administrator and tutor can manage the note file that included in the note folder. 3. Administrator and tutor can upload the note file in PPTX or PDF format. 4. Students can download the note file uploaded by administrator or tutor.
Notice Management	<ol style="list-style-type: none"> 1. Administrator can manage the notice. 2. Tutor and student can view the full article of the latest notice.

The non-functional requirements of the proposed system will be shown in the Table 3.

Table 3: Non-functional requirements of the proposed system

Requirements	Explanation
Performance	1. Response time for the system is less than 5 seconds
Security	1. The system will only allow access to the account when the user input correct email and password. 2. User unable to access the account if the email and password are incorrect.
Availability	1. The system is available if there is internet connection. 2. The system is available for 24 hours a day, and seven days a week.
Usability	1. This system is user-friendly and easy to learn for all users. 2. This system should use icons and buttons that easy to understand by all users in user interfaces.
Localization	1. The language used in this system is English as this system is published in Malaysia.

3.3 Design phase

This is the third main phase of the Agile Model which outlines the specifications, features, and operations that required to meet the functional requirements of the proposed system. The methods to deliver the required functionality will be focus on this phase. The system architecture diagram which provides an overview of the proposed system’s physical deployment will be illustrated. Unified Modeling Language (UML) diagrams which are use case diagram, sequence diagram, class diagram, and activity diagram will be created in this phase. UML diagrams are created in an attempt to unify and simplify the plethora of methods of Object-Oriented development that had sprung up [7]. To design the user interface, wireframe will be created to visualize the page layout and the available functionalities of the interface.

The system architecture diagram of the proposed system will be shown in Figure 1.

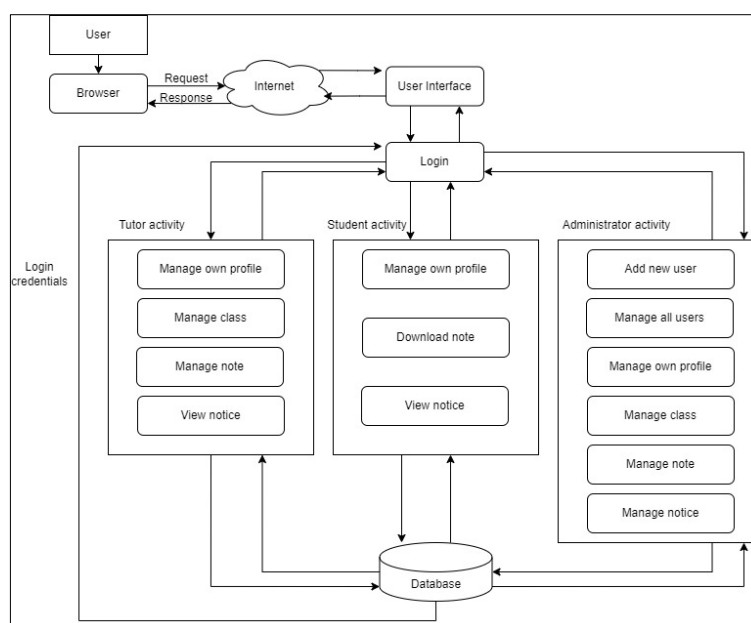


Figure 1: System Architecture diagram of the proposed system

The UML diagrams including the use case diagram, activity diagrams, and class diagram of the proposed system will be shown in Figure 2 to Figure 6.

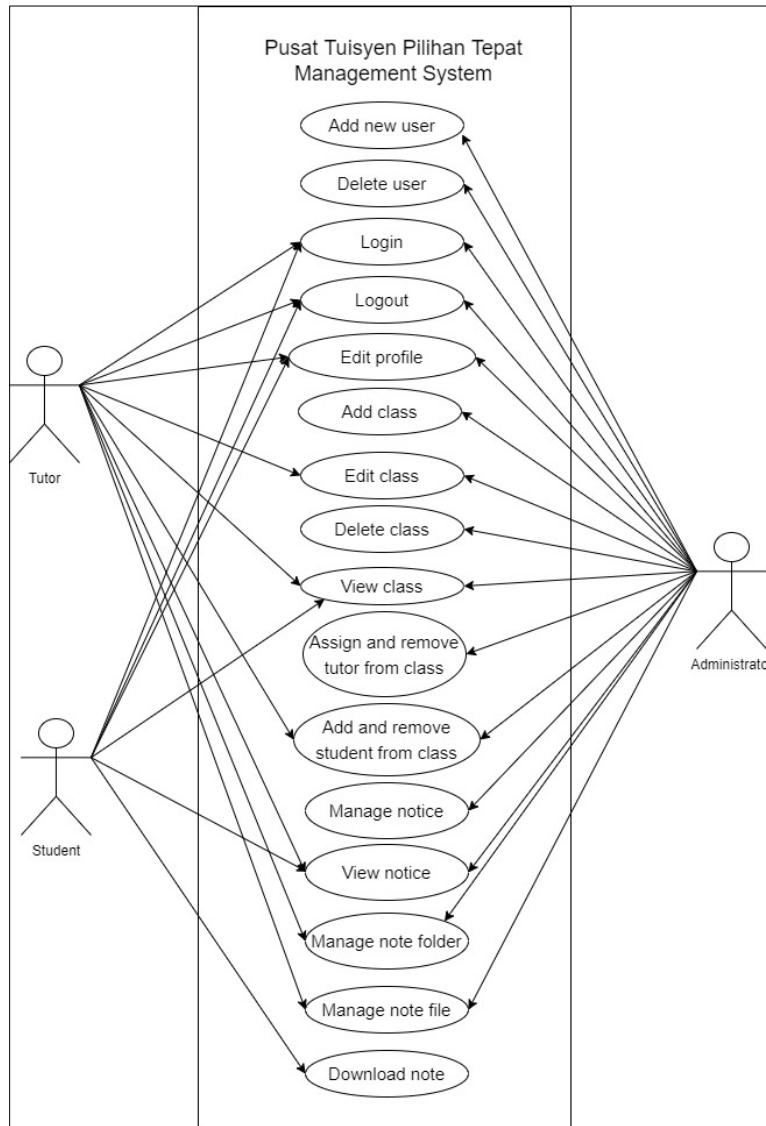


Figure 2: Use case diagram of the proposed system

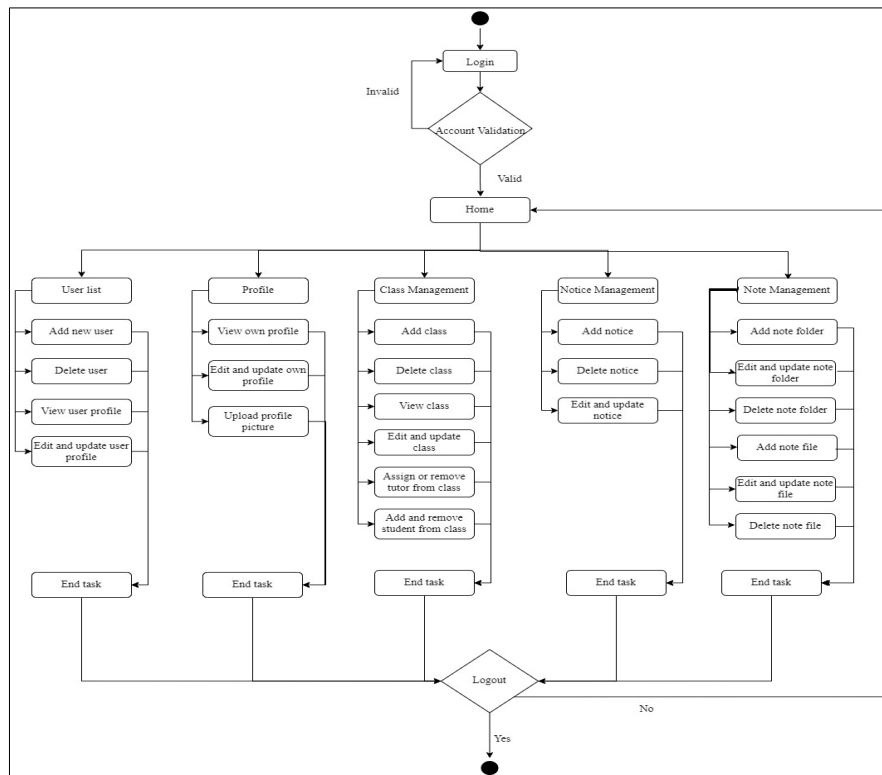


Figure 3: Activity diagram of administrator

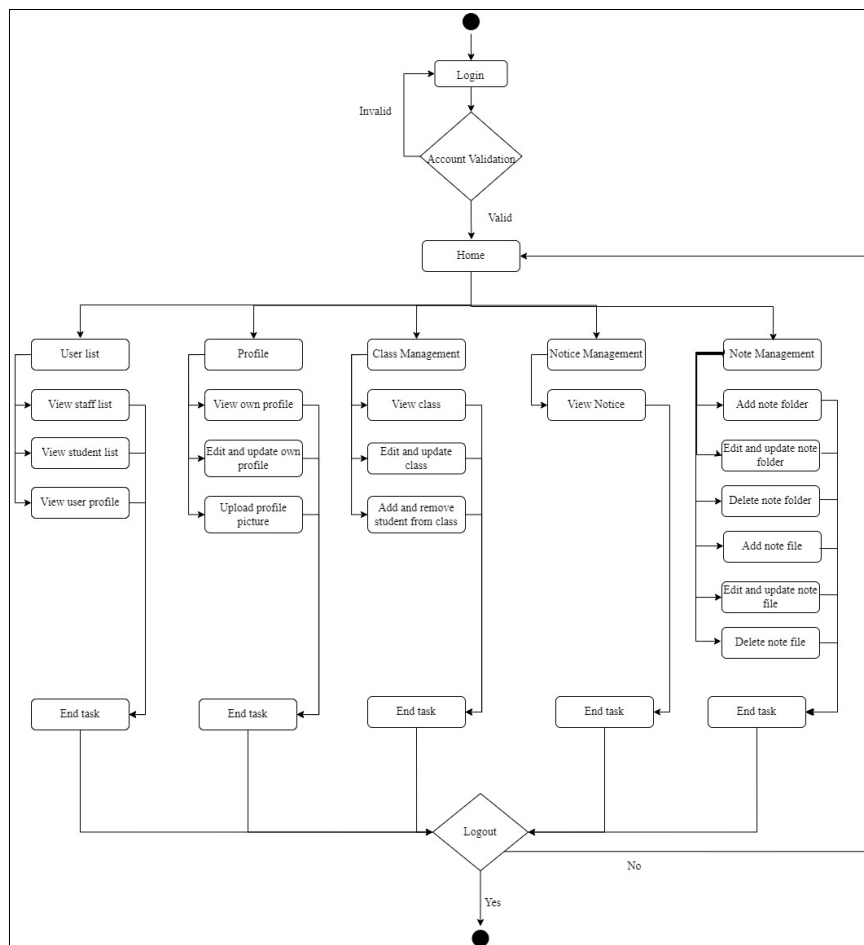


Figure 4: Activity diagram of tutor

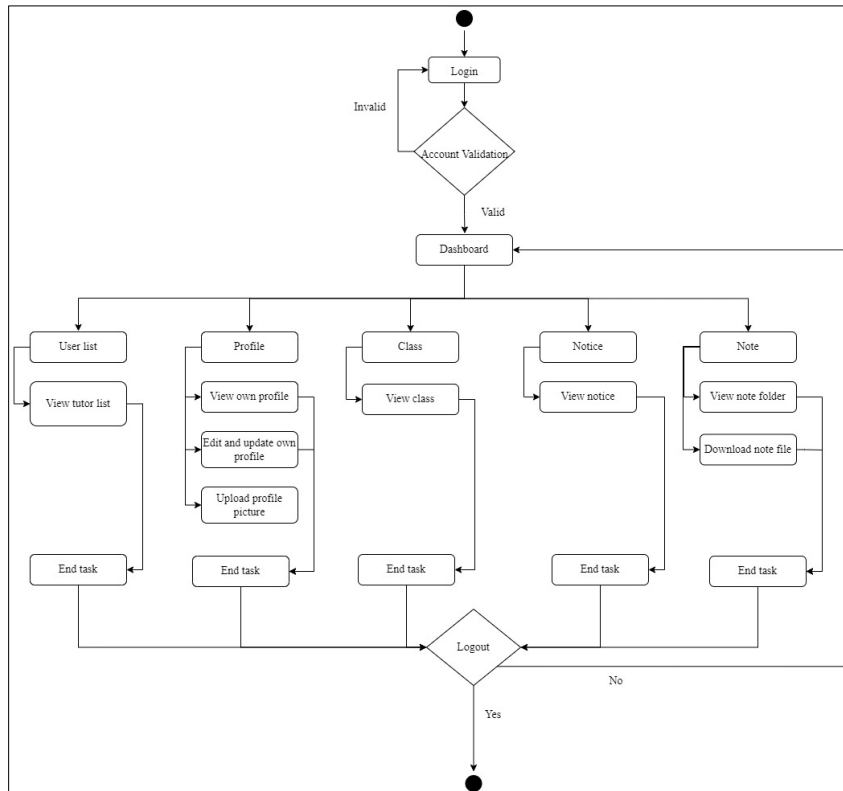


Figure 5: Activity diagram of student

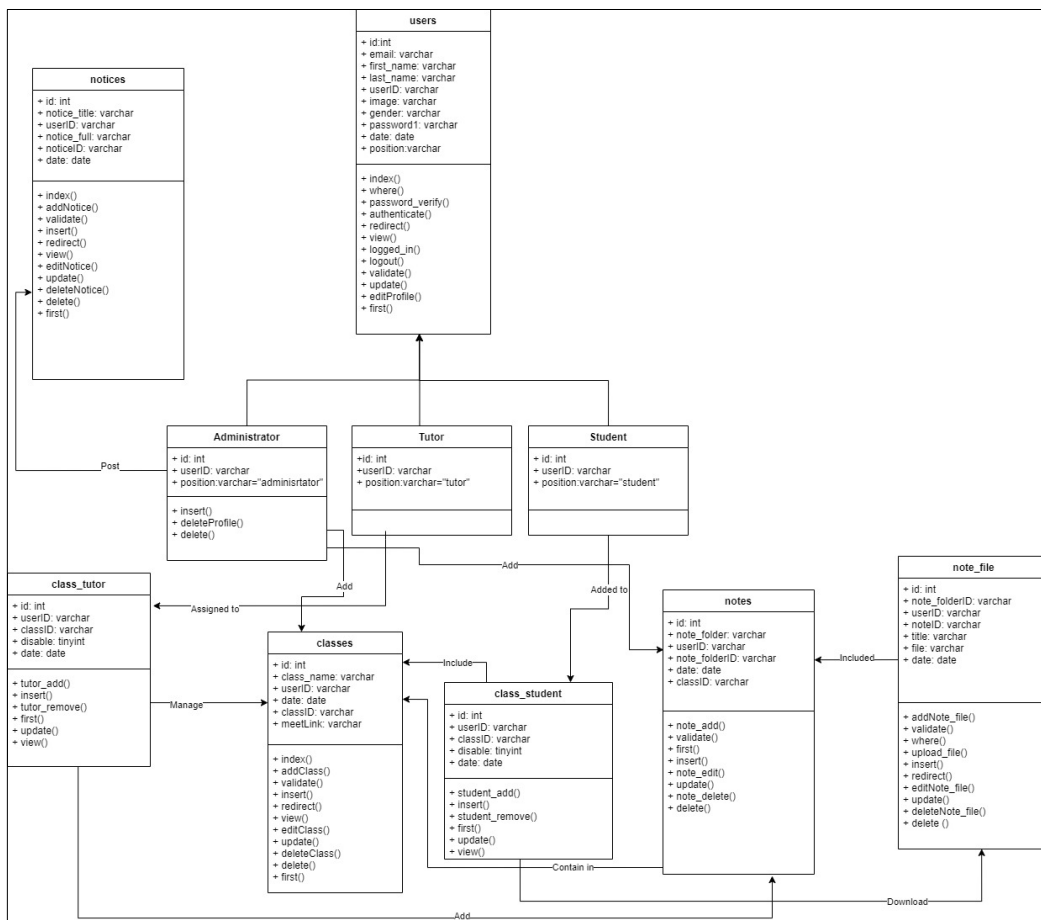


Figure 6: Class diagram of proposed system

The design or construction of a system’s user interface is referred to as user interface design. Each interface’s wireframe will be drawn in the user interface design process. The wireframe of the proposed system is illustrated from Figure 7 to Figure 12.

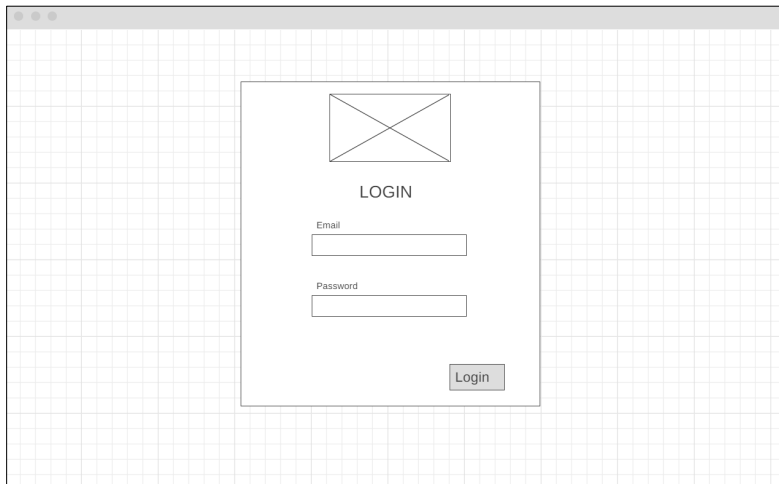


Figure 7: Wireframe of login interface

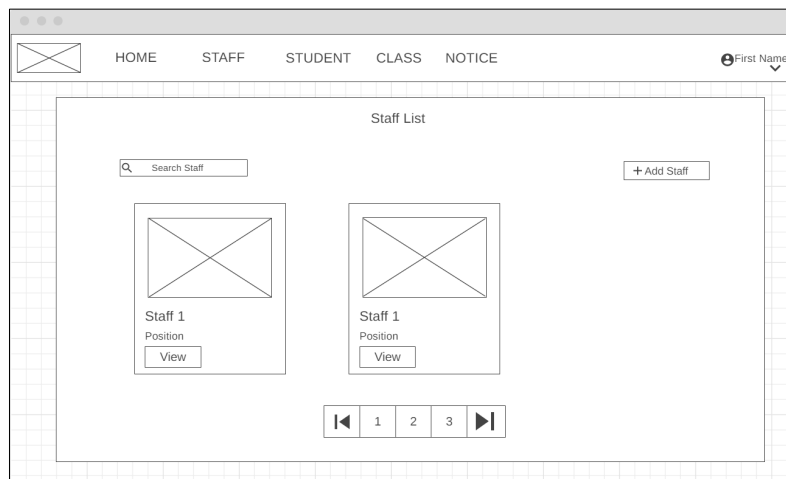


Figure 8: Wireframe of user list interface

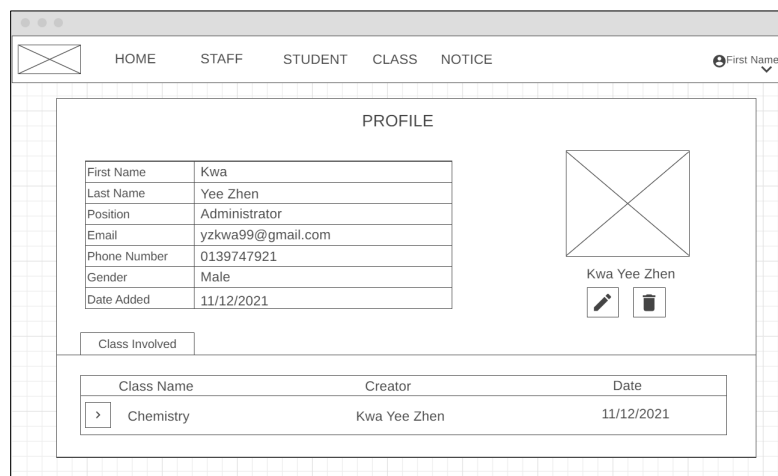


Figure 9: Wireframe of user profile interface

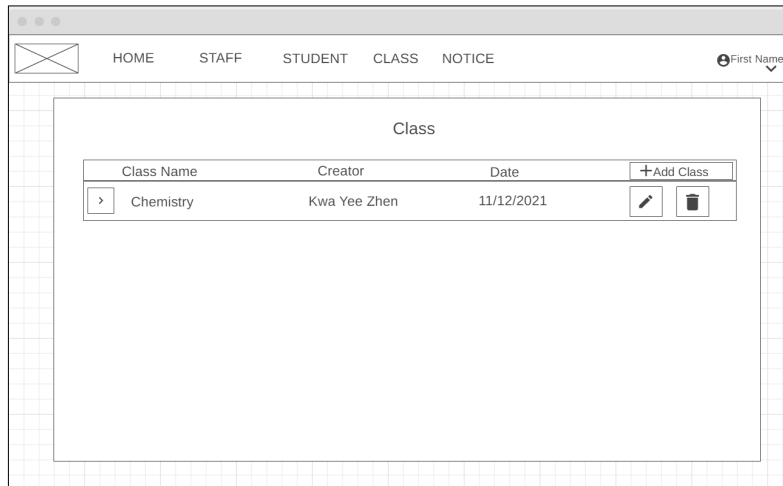


Figure 10: Wireframe of class management interface

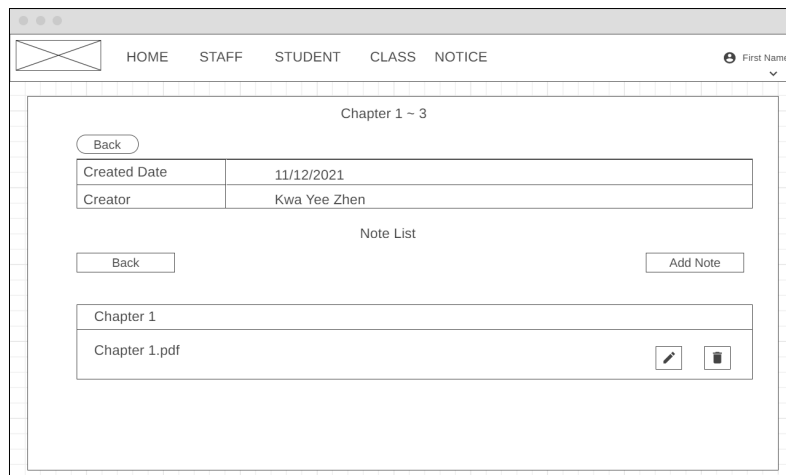


Figure 11: Wireframe of notes management interface

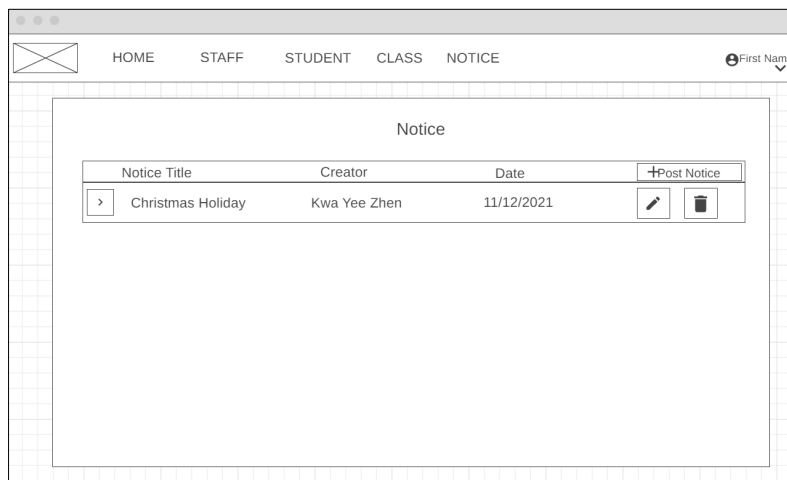


Figure 12: Wireframe of notice management interface

3.4 Implementation phase

In this phase, the implementation and development of the proposed system will be conducted. The modules of the proposed system will be developed based on the requirements and design that determined on the previous stages. PHP programming language is used to develop the back end of the

proposed system. The database will be built by using MySQL and will be connected to the back end of the proposed system. The front end of the proposed system will be developed via HTTP, CSS, and JavaScript. Figure 11 to Figure 18 shows the interfaces of each module in the proposed system.

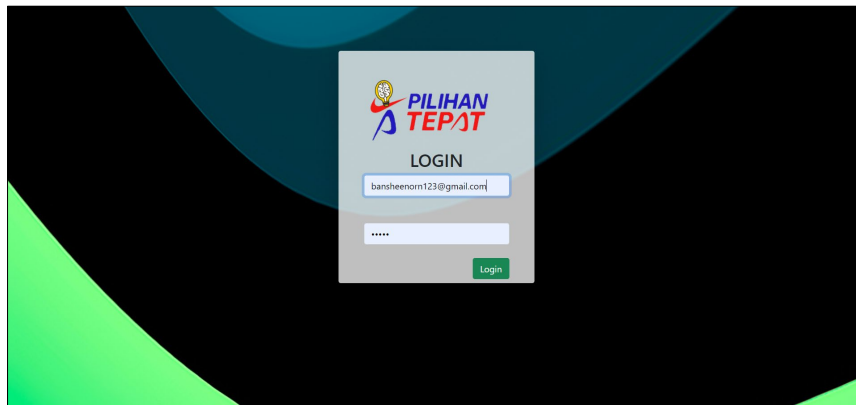


Figure 13: Login interface of the proposed system

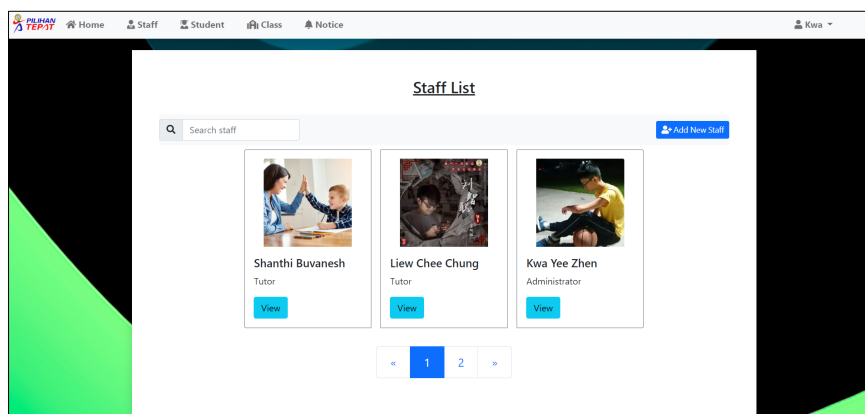


Figure 14: User list interface of the proposed system

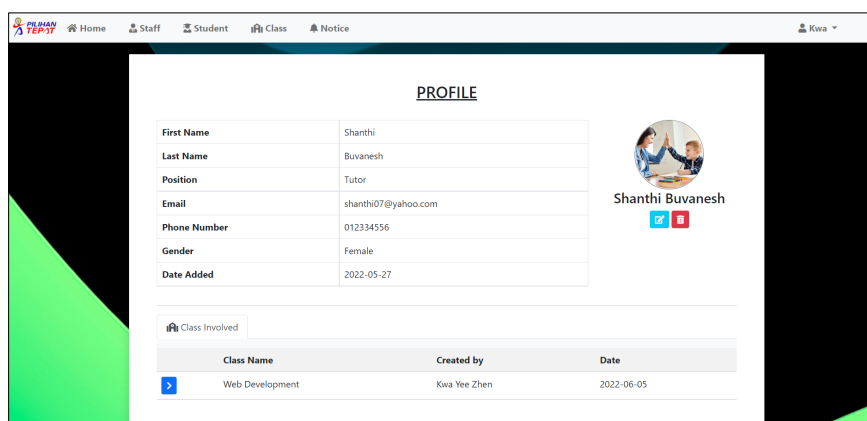


Figure 15: User profile interface of the proposed system

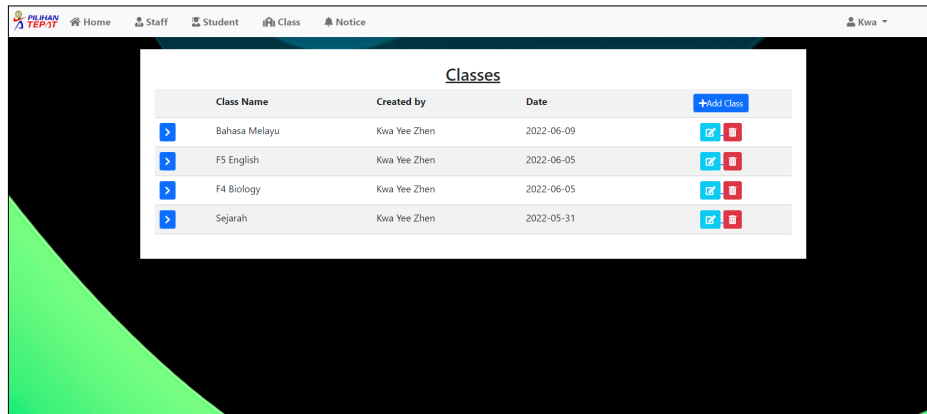


Figure 16: Class management interface of the proposed system

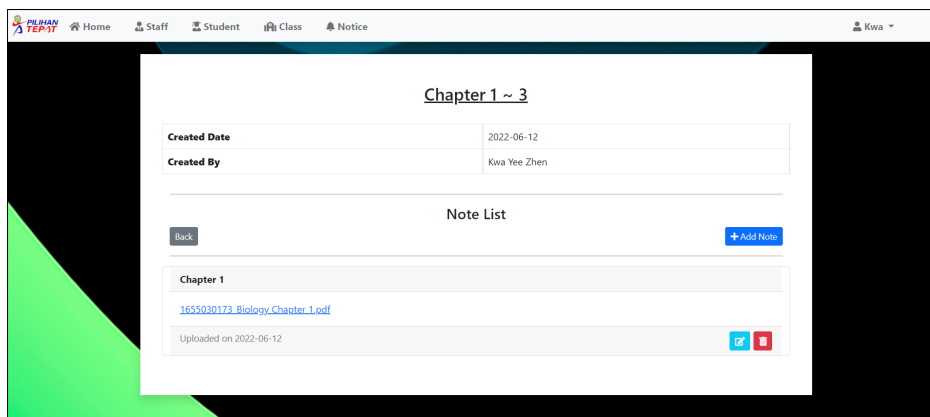


Figure 17: Notes management interface of the proposed system

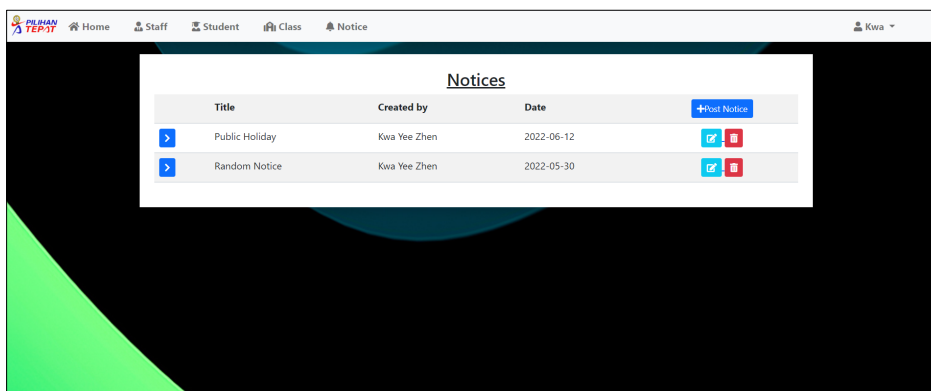


Figure 18: Notice management interface of the proposed system

4. Results and Discussion

There will be two types of testing conducted in the testing phase of this project, which are the functional testing and User Acceptance Testing. These testing must be carried out to ensure that the proposed system meets all of the requirements identified during the analysis phase.

4.1 Functional Testing

The functional testing is a testing that involves comparing the system's output to the requirements and specifications to confirm that it fulfils all the criteria. The results of functional testing for each module of the proposed system will be shown in Table 4 to Table 10.

Table 4: Functional testing result for user login module

Functions	Test Case	Expected Result	Actual Result
Login	Empty or incomplete data input.	An error message will display if any input field is empty.	Pass
	Complete data input with invalid email or password.	An error message will display if the input is invalid.	Pass
	Complete data input with correct email and password.	Successfully login and redirect to the homepage.	Pass
Logout	Press the logout in navigation bar.	Successfully logout and redirect to login interface.	Pass

Table 5: Functional testing result for user management module

Functions	Test Case	Expected Result	Actual Result
Add new user	Empty or incomplete data input.	An error message will display if any input field is empty.	Pass
	Complete data input with invalid input.	An error message will display if the input is invalid.	Pass
	Complete data input with valid input.	Successfully added new user.	Pass
User list	Search for user with invalid name.	A message will display to show that no user was found.	Pass
	Search for user with valid name	The searched user will display in user list.	Pass
	Press on View button in each user	Redirect to each user profile.	Pass
Profile management	Display the personal information and class involved by the user in the profile.	The profile will display personal information and class involved.	Pass
	Edit and update profile with invalid input	An error message will display if any input is invalid.	Pass
	Edit and update profile with valid input	Successfully edit and update and the updated personal information will be display.	Pass
	Upload profile picture	Uploaded profile picture display in profile.	Pass
Delete user	Delete specific user account	The user will be deleted from the database and redirect user to user list.	Pass

Table 6: Functional testing result for class management module

Functions	Test Case	Expected Result	Actual Result
Class list	Press on view button of each class	Redirect user to the specific class interface	Pass
Add new class	Invalid input	An error message will display if the input is invalid.	Pass
	Valid input	Successfully added the class and redirect user to class list.	Pass
Edit class	Invalid input	An error message will display if the input is invalid.	Pass
	Valid input	Successfully edit and update and the updated class information will be display.	Pass
Delete class	Delete specific class	The class will be deleted from the database and redirect user to class list.	Pass
Assign tutor	Search for a non-existing tutor	A message will display to show that no tutor was found.	Pass
	Search for a tutor with empty input	An error message will display if the input is empty.	Pass
	Search for an existing tutor	The searched tutor will display.	Pass
	Press the assign button on the searched tutor	Successfully assigned the tutor to the class.	Pass
Remove tutor	Search for tutor that not assigned to the class	A message will display to show that no tutor was found.	Pass
	Search for tutor that assigned to the class	The searched tutor will display.	Pass
	Press the remove button on the searched tutor	Successfully remove the tutor from the class.	Pass
Add student	Search for a non-existing student	A message will display to show that no student was found.	Pass
	Search for a student with empty input	An error message will display if the input is empty.	Pass
	Search for an existing student	The searched student will display.	Pass
	Press the add button on the searched student	Successfully added the student to the class.	Pass
Remove student	Search for student that not added to the class	A message will display to show that no student was found.	Pass
	Search for student that added to the class	The searched student will display.	Pass

Table 7: (cont)

Functions	Test Case	Expected Result	Actual Result
Remove student	Press the remove button on the searched student	Successfully remove the student from the class.	Pass

Table 8: Functional testing result for notes management module

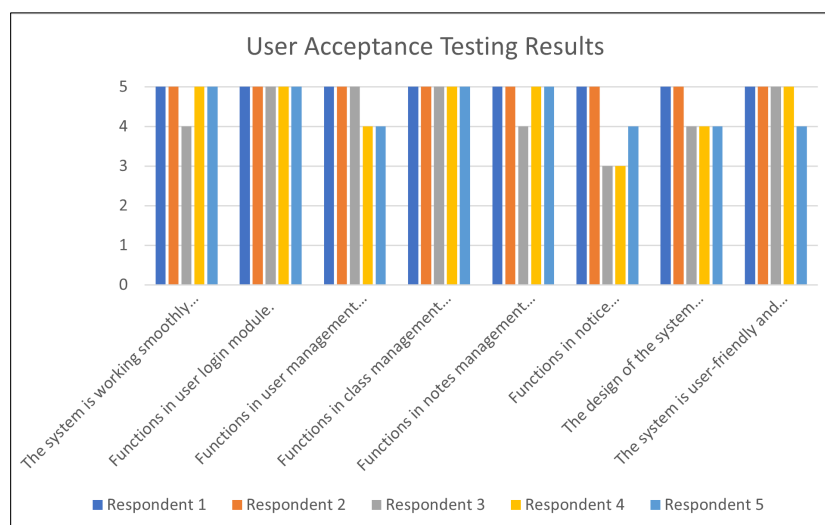
Functions	Test Case	Expected Result	Actual Result
Note folder list	Press on view button of each note folder	Redirect user to the specific note folder interface.	Pass
Add new note folder	Empty input	An error message will display if the input is empty.	Pass
	Valid input	Successfully added the note folder and redirect user to note folder list.	Pass
Edit note folder	Empty input	An error message will display if the input is empty.	Pass
	Valid input	Successfully edit and update and the note folder. The updated note folder information will be display.	Pass
Delete note folder	Delete specific note folder	The note folder will be deleted from the database and redirect user to note folder list.	Pass
Add note file	Empty input	An error message will display if the input is empty.	Pass
	Valid input with the upload of note file	The note file is successfully added and redirect user to specific note folder interface.	Pass
Edit note file	Empty input	An error message will display if the input is empty.	Pass
	Valid input	Successfully edit and update the note file. The updated note file information will display.	Pass
Delete note file	Delete specific note file	The specific note file will be deleted from the database.	Pass
Download note file	Press on the file name of note file	Successfully download the note file.	Pass

Table 9: Functional testing result for notice management module

Functions	Test Case	Expected Result	Actual Result
Notice list	Press on view button of each notice	Redirect user to the full notice interface	Pass
Post new notice	Empty input	An error message will display if the input is empty.	Pass
	Valid input	Successfully posted the notice folder and redirect user to notice list.	Pass
Edit notice	Empty input	An error message will display if the input is empty.	Pass
	Valid input	Successfully edit and update and the updated notice information will be display.	Pass
Delete notice	Delete specific notice	The specific notice will be deleted from the database	Pass
View full notice	Display the content of the notice in full notice interface	The full notice interface will display the full content of notice	Pass

4.2 User Acceptance Testing

The User Acceptance Testing occurs after functional testing and act as the final step in the testing phase. The purpose of User Acceptance Testing is to enhance the quality of the proposed system and determining the level of user acceptance. Figure 19 shows the bar chart for the result of User Acceptance Testing while the table for the result of User Acceptance Testing is shown in Appendix A.

**Figure 19: Bar chart for the User Acceptance Testing Results**

Based on the result, we can conclude that the majority features of the developed proposed system were reach the level of acceptance of users. No user has expressed dissatisfaction with the features of the proposed system. The functions in the notice management module, on the other hand, do not totally satisfy users.

5. Conclusion

In conclusion, the developed proposed system has met and exceeded the stated objectives, project scopes, expected results, and project significance. Even though the system has been successfully developed, there are still some limitations and disadvantages in the system. Lack of online fee payment management functionality will be one of the limitations. Due to this limitation, students are unable to pay their tuition fees via online transactions and the administrator is also unable to manage the payment record of the students. Furthermore, there is no notification counter functionality which will display notification counter on the navigation bar if any new notice is published. Next, the users of the proposed system are unable to categorized by year due to the lack of year management functionality.

There will be several future works to improve the developed proposed system. At first, an online fee payment management functionality should be implemented in the future to allow the students to make their monthly tuition fee payment via online transaction. Next, notification counter functionality can be included to the system in the future to assist the users in staying informed about the Tuition Centre’s most recent notices and information. Lastly, the proposed system should have year management functionality to improve the documentation process of the user’s information.

Acknowledgment

I would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Appendix A

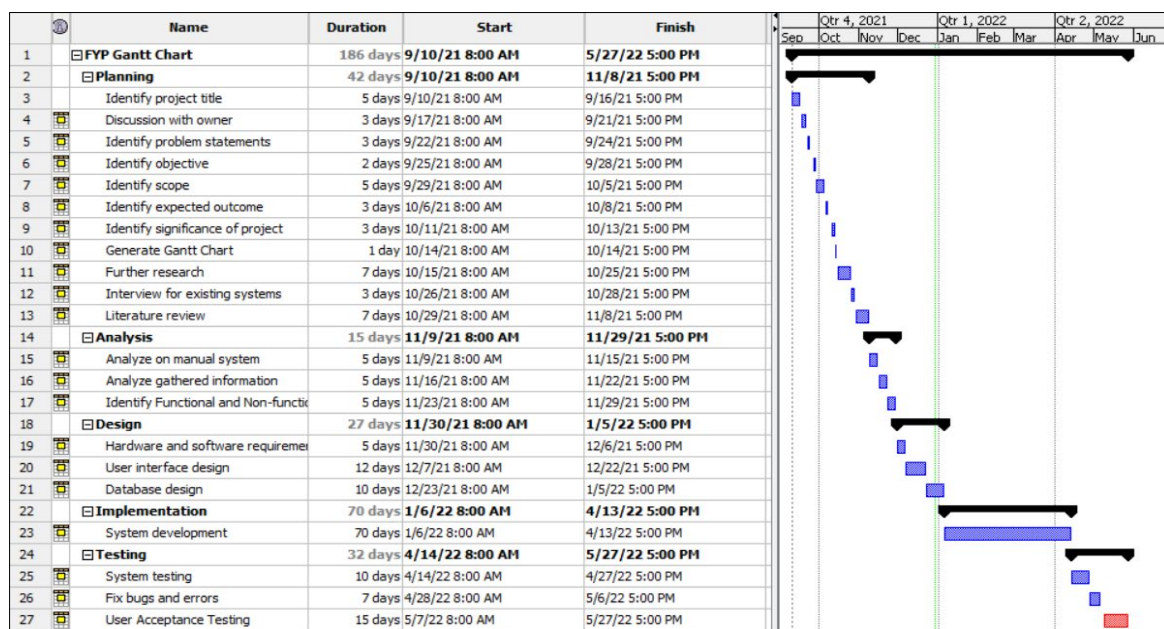


Figure 20: Gantt Chart of the proposed project

Table 11: Functional testing result for notice management module (continue)

Features	Rating (Scale from 1 to 5)				
	1	2	3	4	5
The system is working smoothly in all functions	0	0	0	1	4
Functions in user login module	0	0	0	0	5
Functions in user management module	0	0	0	2	3
Functions in class management module	0	0	0	0	5
Functions in notes management module	0	0	0	1	4
Functions in notice management module	0	0	2	1	2
The design of the system interface	0	0	0	3	2
The system is user-friendly and straightforward	0	0	0	1	4

References

- [1] Shah, M. (2014). Impact of management information systems (MIS) on school administration: What the literature says. *Procedia-Social and Behavioral Science*, 116, 2799-2804.
- [2] Chai, W. C., & Mostafa, S. A. (2021). Bright Kids Tuition Centre Management Information System. *Applied Information Technology and Computer Science*, 2(2), 937-957.
- [3] Bharamagoudar, S. R., Geeta, R.B., & Totad, S.G. (2013). Web based student information management system. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(6), 2342-2348.
- [4] Haryani Ibrahim. (2007). Tuition Centre Management System.
- [5] Zarifah Mohd Pahmi. (2014). Tuition Centre Management System (TCMS).
- [6] McCormick, M. (2012). Waterfall vs. Agile methodology. *MPCS*, N/A.
- [7] Jacobson, L., & Booch, J. R. G. (2021). The unified modeling language reference manual.