Applied Information Technology And Computer Science Vol. 3 No. 2 (2022) 356-369

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

Design and Development of Online Hostel Management System to Improve Application and Management Process

Wong Pei Ling[1], Hairulnizam Mahdin*

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

DOI: https://doi.org/10.30880/aitcs.2022.03.02.024
Received 14 June 2022; Accepted 26 September 2022; Available online 30 November 2022

Abstract: UTHM Hostel Management System is developed especially for Universiti Tun Hussein Onn Malaysia (UTHM) students and the university college hostel management. This project is proposed to assist the hostel management to manage hostel activities more efficiently. The current process is more complex and inefficient, with a time-consuming complaint process, and causes inconvenience to both students and hostel management. The project is developed based on the Waterfall model. The programming languages used to develop the system are PHP, Javascript, and SQL. MySQL acts as the database for the proposed system. The system users are the hostel admin and the student. This system allows the admin to manage notices, students, student applications, room allocation, and student complaints. The student can view notices from the hostel, submit a new application, view room details, and complaint about the hostel facilities. All the details about the UTHM Hostel Management System are reviewed in this paper.

Keywords: Hostel Management System, Web-based, Waterfall method

1. Introduction

Hostel management can be defined as the management of all hostel activities such as student registration, room allotment, calculating bills and fines, and generating related reports. Usually, the hostel administrator takes responsibility for hostel management. A hostel management system can make jobs much easier for all the users of the application [1]. When applying for the hostel, students can just register and apply online via the system without having to walk into the hostel [2]. The hostel management system is to be used for managing student hostel accommodations for Universiti Tun Hussein Onn Malaysia (UTHM). One hostel can accommodate an estimated 600 students. This hostel management system is important in managing a hostel as it can speed the registration and other processes thus improving efficiency and saving time. Besides, it provides convenience to both hostel admin and students.

*Corresponding author: hairuln@uthm.edu.my

Currently, hostel management is computerized but still requires manual work. The system that they used was the residential college application system (Ehapp) and the Student Information system. (SMPi). The Ehapp system was used by students to apply for the hostel while the SMPi system was used by the hostel management to complete hostel management activities such as allocating rooms to the student. The system admin for the Ehapp system which acts as a moderator will generate the student list from student applications and submit the student list to the hostel principal who is the user of the SMPi system. Then, the hostel principal will select the student who meets the requirements to stay in the hostel and allocate rooms to them through the SMPi system.

There are limited features in the existing system. Students only can apply for hostels and view their application status and information online in the Ehapp system. The moderator needs to collect and arrange the student list and give it to the hostel management. This situation involves additional manpower. While using the proposed hostel management system, the hostel management can directly view the student application and approve or reject the student application for the hostel through the proposed system.

The proposed UTHM hostel management system will be able to store all hostel records in the database with improved features. In this system, the hostel admin can manage students, assign rooms to students, manage the complaints collected from students, and generate student reports. Students can apply to hostels for the new semester, and complain about broken hostel facilities. This system provides great convenience to all people that are from hostel management (hostel admin, hostel manager, and hostel holders) and students.

2. Related Work

2.1 Database

The database used for developing the proposed system is the MySQL database by using phpMyAdmin, which is an open-source administration tool for MySQL [3]. By using phpMyAdmin, various tasks can be performed such as creating, modifying, or deleting databases, tables, rows, or columns. It also can run SQL statements and manage users and their rights.

2.2 Hosting Requirement

The developed system will be hosted on the local web server by using the XAMPP control panel. XAMPP is a short form for cross-platform, Apache, MySQL, PHP, and Perl. XAMPP is an application server for hosting PHP websites. Since the proposed system was to be developed using PHP language, the use of XAMPP is compulsory. It contains a phpMyAdmin administration tool which contains a MySQL database.

2.3 A study on similar systems

Three similar systems and the existing system are selected for comparison for the benchmark of the proposed system. The systems are Ehapp and SMPi, EduSec, and Zuan. A table will be represented later to show their difference and similarity with our proposed application. The Ehapp system has a simple interface where it is easy to see the important function on the page. It has only three pages in the user module. This module provides limited features to the user. The student can only apply for the residential college, view their application status, and view their profile. While in SMPi, its interface was too complex and contained too many elements on one page. The user of the system may be confused when they want to perform an action in the system. On the other hand, EduSec is a fully functional hostel management system. It contains many basic features that should be in a hostel management system. However, its interface contains too much information. There is too much information on a page. This is complicated and the system admin may be confused when they want to perform an action. They

will take a longer time to find where information is on the page. Besides, a too complicated system interface will provide less user experience to the system users.

The Zuan hostel management system is easy to use as its interface is simple. However, some of its interface content is not arranged properly. For example, in the dashboard, the columns are not arranged properly. The space between columns is inconsistent and this causes the interface to look less attractive. Besides, Zuan does not have a user module. This means all actions are to be performed at the admin site. The user cannot register themselves via their mobile or laptop. This causes inconvenience to both the hostel user and the hostel admin. Table 1 shows the comparison between the existing applications and the proposed application.

Table 1: Comparison between the existing applications and the proposed application

Features	System[1]	System[2]	System[3]	System[4]
Online application	Yes	Yes	No	Yes
Room allotment at admin site	Yes	Yes	Yes	Yes
Register at the user site	No	Yes	No	Yes
Complain broken facilities at the user site	No	No	No	Yes
Visitors' management	No	Yes	No	No
Room transfer	Yes	Yes	Yes	Yes
Cashless payment option	No	Yes	Yes	No
Notice module	No	No	No	Yes
Check-in/check-out at the admin site	Yes	Yes	Yes	Yes
Generate student report	Yes	Yes	Yes	Yes

System[1]: Ehapp and SMPi

System[2]: Edusec

System[3]: Zuan

System[4]: UTHM Hostel Management System

3. Methodology

Methodology refers to the framework that is used to improve the management and control of the software development process, structure and simplify the process, and standardize the development process and product through the specifying process of the activities to be accomplished and technologies to be used [4].

3.1 Waterfall model

The Waterfall model has been chosen to develop the UTHM Hostel Management System. The waterfall model is a linear sequential approach in Software Development Life Cycle (SDLC) and is famous in product development. In the waterfall model, the logical progression of the step is emphasized. The progress in the waterfall model is like the water flow direction over the edge of a cliff, in which in each development phase, there are different endpoints, and these endpoints cannot be accessed again

once accomplished. The development process continues to go forwards. Each stage depends on the information from the earlier stage. Since the stages are accurate and rigid, each stage is executed one at a time, enabling it to be easily maintained [5].

There are five major phases in the waterfall model, which are the requirement analysis phase, design phase, implementation phase, testing phase, and maintenance phase [6]. Each phase needs to be done carefully as each such phase will affect the subsequent phases. Therefore, it needs to be reviewed and evaluated before moving on to the next phase. Figure 1 shows the waterfall model diagram.

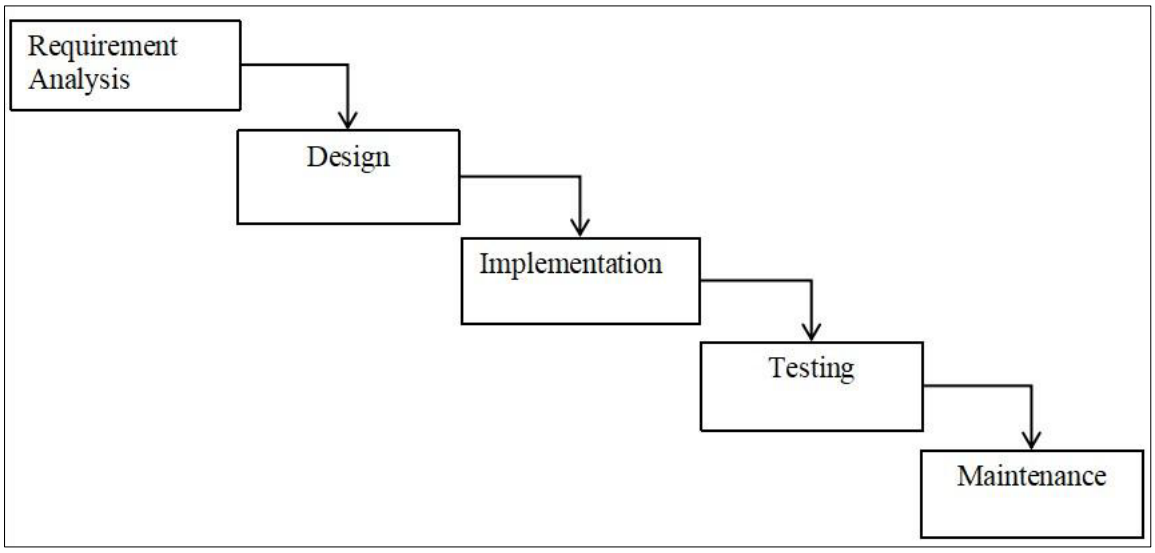


Figure 1: Waterfall model diagram

3.1.1 Requirement analysis phase

At this phase, the user requirements were being gathered and analyzed by conducting an interview session with the user of the existing system which is one of the UTHM hostel assistant hostel managers, Madam Siti Maria Binti Anuar. The interview session was conducted through the Google Meet. During the Google Meet session, all the user requirements of the system were recorded for reference purposes. The requirements were based on how to solve the problem faced by UTHM hostel's existing Ehapp system by creating an improved version of the system to help their business run more fluently. Table 2 shows the functional requirements and Table 2 shows the non-functional requirements.

Table 2: Functional requirements

Module	Functionalities
Login I. Admin II. Student	I. User input valid ID and Password. II. System alert for any invalid input.
Dashboard	III. Admin input notice to be displayed on the student dashboard.
Profile I. Admin II. Student	I. Admin can edit his/her details. II. Students can edit their details.
Application I. Admin II. Student	I. Students can apply for the hostel. II. System admin can approve or reject student applications.
Room allocation I. Admin II. Student	I. Admin can allocate rooms to students. II. Admin can vacate the room allocated. III. Students can view allocated room details.
Report	I. Admin can generate student reports in each hostel.

Module	Functionalities
Register I. Student	I. Students can register as new users to the system.

Table 2: (cont)

Change password and forget password I. Admin II. Student	I. Admin can change his/her password. II. Students can reset the password via reset password email in case they forget the password. III. Students can change their passwords.
Complaint I. Admin II. Student	I. Students can complain about broken facilities in the hostel or other complaints about the hostel's faultiness. II. Admin can view and update the complaint status.

Table 3: Non-Functional requirements

Module	Functionalities
Operational	I. The proposed system should work on any web browser. II. The system should be available when there is an internet connection.
Security	I. Users can only access the system with the correct username and password. Passwords must be a combination of alphabet and number to meet strong password requirements. II. The password should be saved with the bcrypt password-hashing function to enhance the security.
Privacy	I. Admin is not allowed to view and edit student information such as passwords.
Performance	I. The response time of the system should be less than three seconds.

3.1.2 Design phase

The design phase plays a vital role which aims to design all system requirements derived from the requirement analysis phase. The system prototype was developed to provide an overview of the functions that need to be implemented by the proposed system. A flowchart was designed as it is the best way to describe a system in terms of its process in sequential order [7]. A Data Flow Diagram (DFD) was designed as a reference in designing a database for the system. Furthermore, the Entity-Relationship Diagram (ERD) was designed to visualize how data in the system is connected. A Data dictionary was generated to ease the design of the database. Besides, a test plan was generated to enable testing of the proposed system based on the system's functionality and user acceptance. Figure 2 shows the Level 0 Data Flow Diagram for the proposed system.

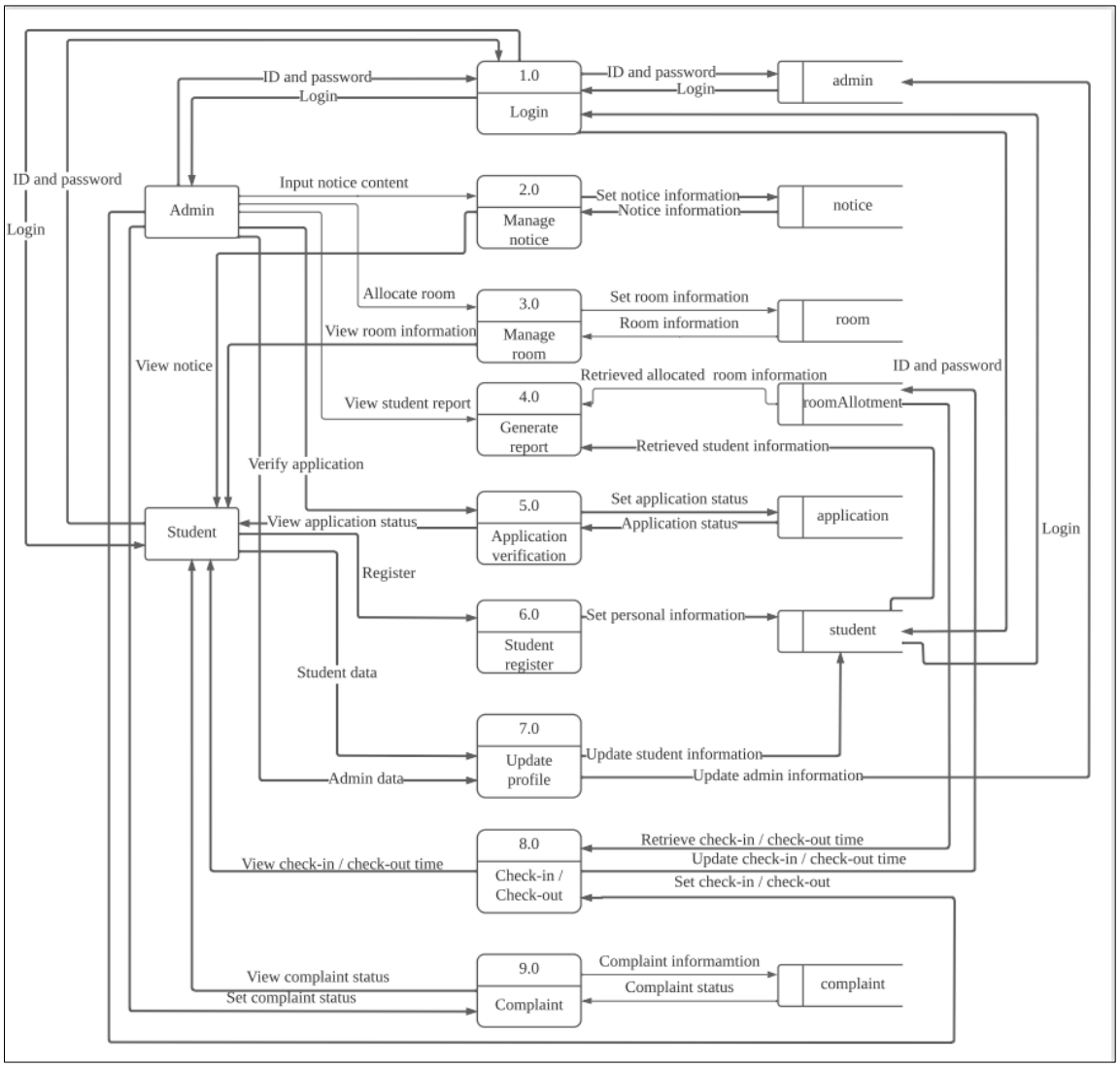


Figure 2: Data Flow Diagram Level 0

3.1.3 Implementation phase

During the implementation phase, the UTHM Hostel Management System was built and tested to see if the system contains the functions as it was designed in the design phase. The web development tools used to develop the system were Hypertext Preprocessor (PHP) language, Hypertext Markup Language (HTML), Javascript, Cascading Style Sheets (CSS), and Bootstrap.

3.1.4 Testing phase

The testing phase is important to test the usability of the UTHM Hostel Management System to ensure user acceptance of the system interface suitability. The system was tested based on its functionalities, user interface, and system testing with the users. During this phase, the developer can know if the developed system has met the objectives and scope that were stated in the requirement analysis phase. Furthermore, from this testing phase, it can be identified whether the developed system has met the user needs and requirements.

3.1.5 Maintenance phase

This phase is the last phase of the waterfall model. After the testing phase, the feedback result from the user is collected. Bugs and weaknesses of the developed system are identified. Then, a possible solution to the bug and improvement to the developed system is identified.

4. Results and Discussion

There are several details to be discussed which are the flowchart, entity-relationship diagram, interface design, and the user acceptance test of the proposed system development.

4.1 Flowchart

Figure 4 shows the flowchart of the proposed system. There are two types of users for the system, which are the admin and the student.

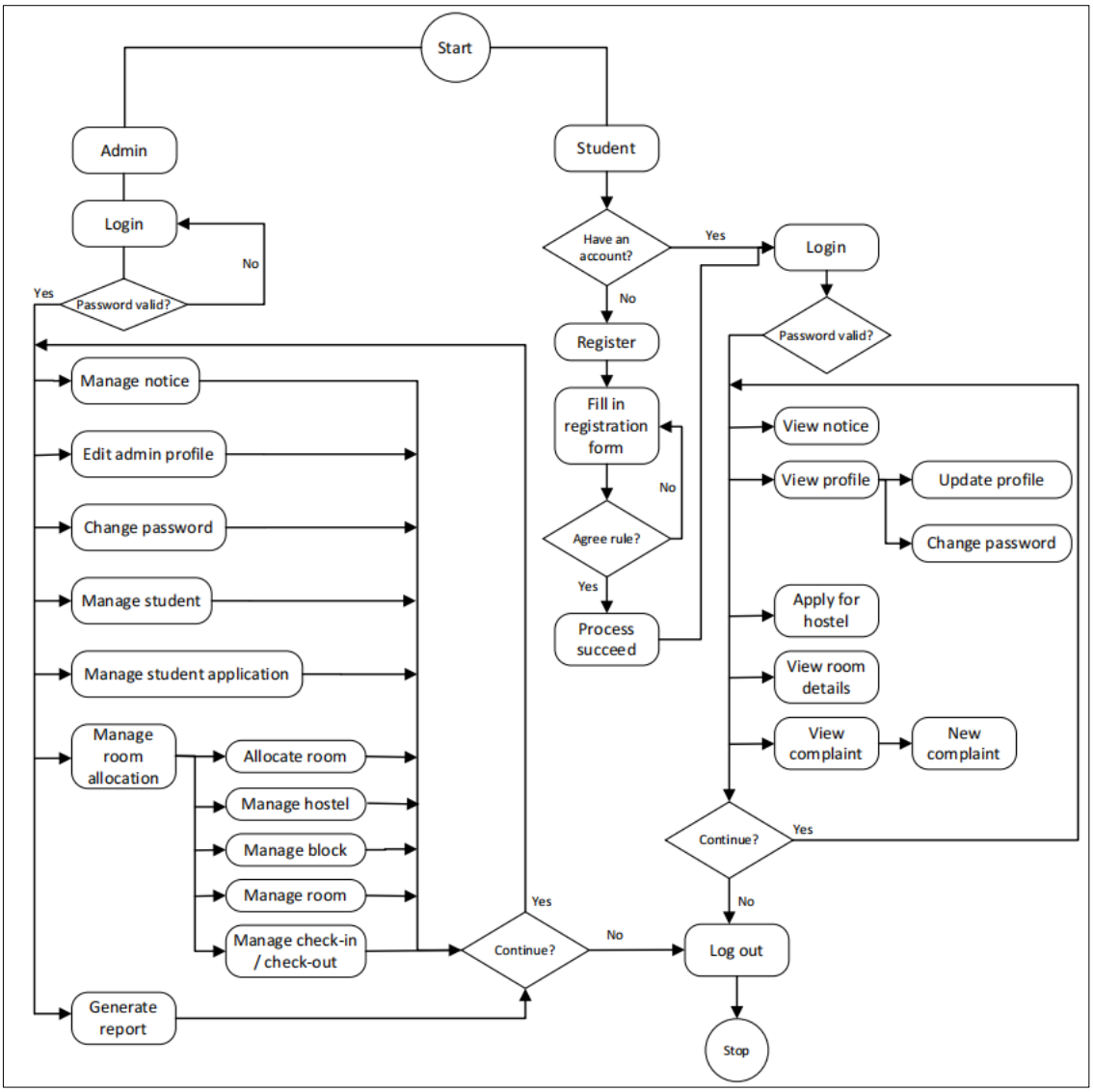


Figure 4: Flowchart

4.2 Entity-Relationship Diagram

Figure 5 shows the entity-relationship diagram of the proposed system. ERD is a visual representation of entities in a system and how the entities are related to each other [8].

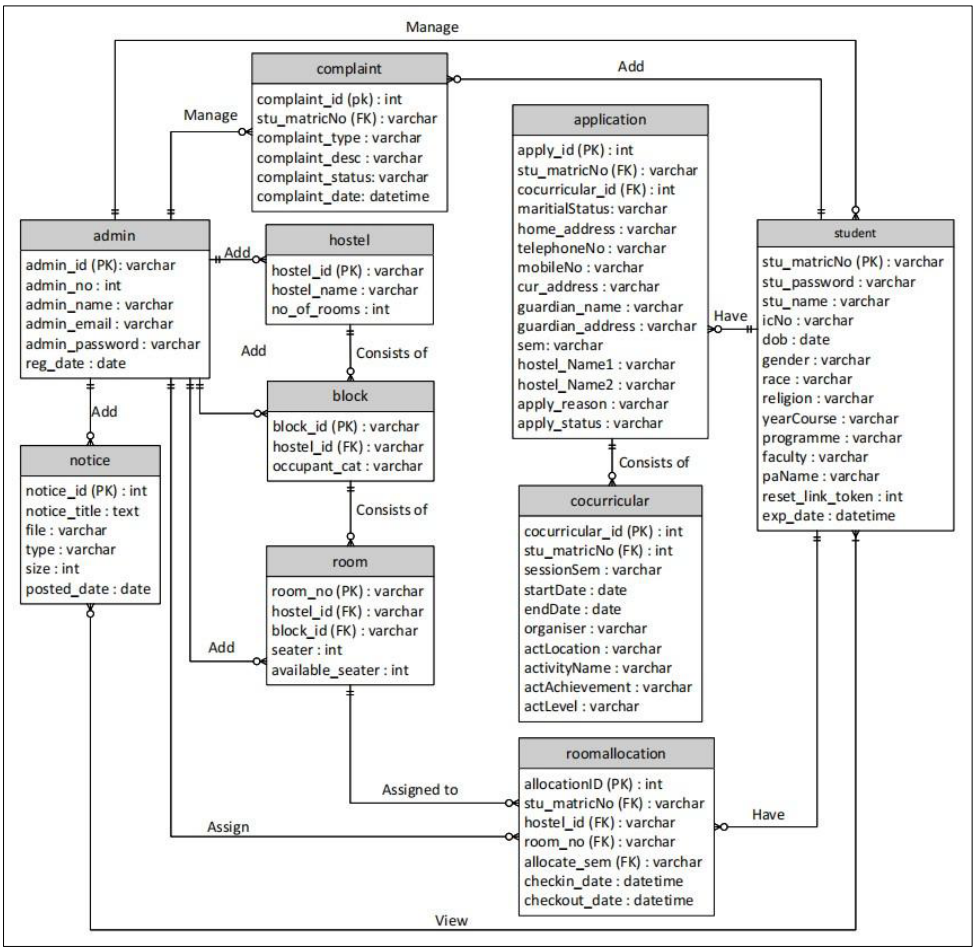


Figure 5: Entity-Relationship Diagram

4.3 Interface design

Figure 6 shows the admin manage notice page. The admin can create, view, update, or delete the notice.

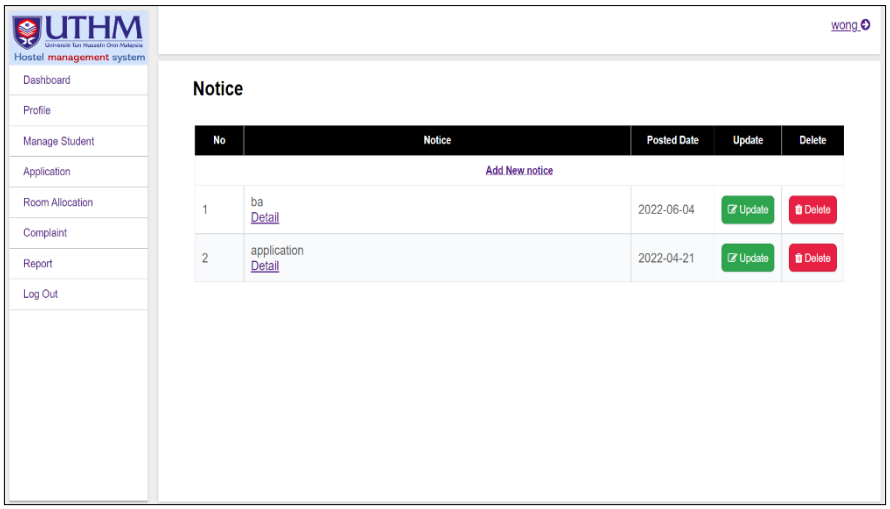


Figure 6: Admin manages notice page

Figure 7 shows the admin room allocation page. The admin can select to manage student room allocation, manage hostels, manage block, manage room, or manage student check-in / check-out. While

using manual system, the admin needs to refer to manual form to view student check-in / check-out time and key into the system. By using the system, the information are managed efficiently.

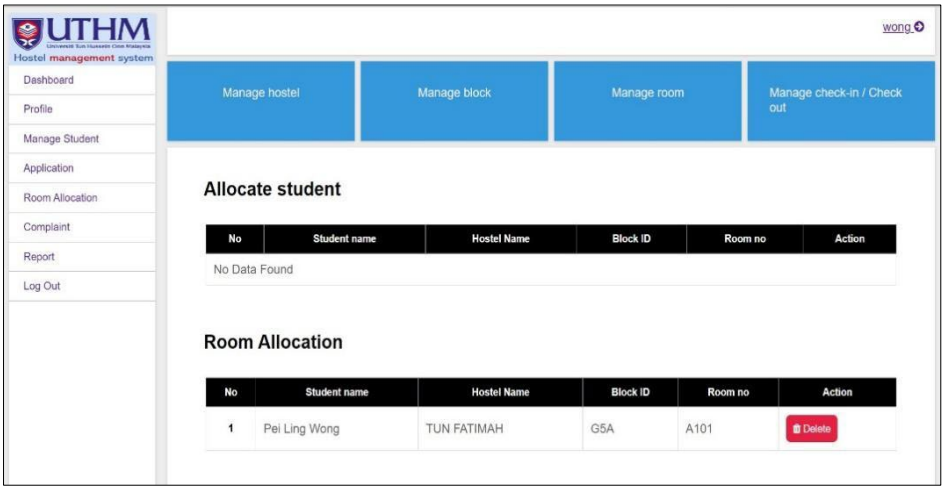


Figure 7: Admin room allocation page

Figure 8 shows the admin manages the complaint page. The admin can view the complaint submitted by the student. Then, the admin can update the complaint status. This complaint module makes the complaint process in the hostel more efficient and timesaving.

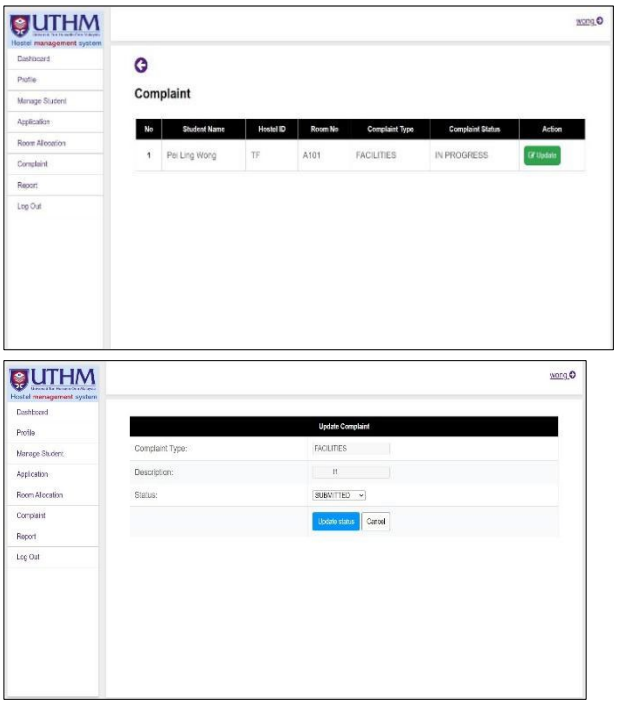


Figure 8: Admin manages the complaint page

Figure 9 shows the student complaint page. The student can view their submitted complaint and add a new complaint about the hostel facilities on this page.

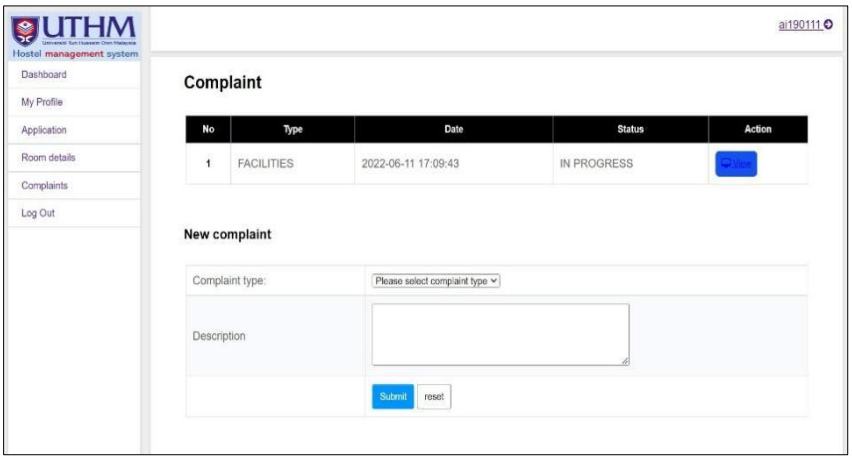


Figure 9: Student complaint page

4.4 Test plan result

Table 4 shows the test plan result for the UTHM Hostel Management System student site functional requirements.

Table 4: System Functionality Test Plan Result (Student)

Section	Test Plan	Expected Result	Actual Result
Register	Submit form without complete data input	Message prompt out: Please fill out all fields	Pass
	Fill in the existed student matric number	Message prompt out: This matric no already exists	Pass
	Submit with the incorrect format of matric number, password, full name, ic number, year course, or academic advisor name	Alert message prompt out	Pass
	Submit the form with all correct inputs	Message prompt out: Register success	Pass
Login	Submit form without data input	Message prompt out: Matric number and Password fields are empty	Pass
	Submit form without complete data input	Alert message prompt out	Pass
Dashboard	View notices	Show all notices	Pass
Profile	View profile	Show profile details	Pass
	Edit profile detail	Message prompt out: Data successfully updated	Pass
	Change password with incomplete data input	Error message prompt out	Pass
	Change password with wrong current password	Message prompt out: Current password is not correct	Pass
	Change password with complete data input	Message prompt out: Password successfully updated	Pass
Application	Submit form without data input	Message prompt out: Please fill out all fields	Pass
	Submit form without complete data input	Message prompt out: Please fill out all fields	Pass

	Submit the form with the incorrect format of telephone number, mobile number, or guardian name	Alert message prompt out	Pass
	Submit the form with all correct inputs	Message prompt out: Application sent	Pass
	View application	Show application details	Pass
Room detail	View room details	Show room details	Pass
Complaint	View complaint record	Show complaint records	Pass
	Submit the form with incomplete data input	Alert message prompt out	Pass
	Submit the form with all correct inputs	Message prompt out: Complaint submitted	Pass

Table 5 shows the test plan result for the UTHM Hostel Management System admin site functional requirements.

Table 5: System Functionality Test Plan Result (Admin)

Section	Test Plan	Expected Result	Actual Result
Login	Submit form without data input	Message prompt out: Admin id and Password fields are empty	Pass
	Submit form without complete data input	Alert message prompt out	Pass
Dashboard	View notices	Show all notices	Pass
	Add new notice	Message prompt out: New notice created successfully	Pass
	Update notice	Message prompt out: Notice Successfully Updated	Pass
	Delete notice	Show confirm alert: Do you want to delete this notice? Clicking 'Yes', the selected notice is deleted from the list.	Pass
Profile	View profile	Show profile details	Pass
	Edit profile detail	Message prompt out: Data successfully updated	Pass
	Change password with incomplete data input	Error message prompt out	Pass
	Change password with wrong current password	Message prompt out: Current password is not correct	Pass
	Change password with complete data input	Message prompt out: Password successfully updated	Pass
Manage student	View student	Show student details	Pass
	Delete student	Show confirm alert: Are you sure to remove this student? Clicking 'Yes', the selected student isdeleted from the list.	Pass
Application	View student application	Show student application details	Pass
	Approve student application	Show confirm alert: Are you sure to approve this application?	Pass

		Clicking 'Yes', the selected student is updated to room allocation list.	
	Reject student application	Show confirm alert: Are you sure to reject this application? Clicking 'Yes', selected student application status updated.	Pass
Room allocation	View student to be allocated	Show student list that had not been allocated	Pass
	Allocate room to selected student	Show confirm alert: Are you sure to assign to this room? Clicking 'Yes', message prompt out: Allocated	Pass
	View room allocation	Show student list that had been allocated	Pass

Table 5: (cont)

	Delete room allocation	Show confirm alert: Are you sure to remove this allocation? Clicking 'Yes', selected student deleted from the list.	Pass
	Add hostel, block, or room	Success message prompt out. Hostel, block, or room added to the list.	Pass
	Update hostel, block, or room	Success message prompt out. Hostel, block, or room updated to the list.	Pass
	Delete hostel, block, or room	Success message prompt out. Hostel, block, or room deleted from the list.	Pass
	Check-in or check-out	Show confirm alert: Are you sure to check in / check out this student? Clicking 'Yes', selected student check in / check out time updated.	Pass
Complaint	View complaint	Show complaint details	Pass
	Update complaint status	Message prompt out: Complaint status updated	Pass
Report	Select the hostel name and generate a report	Show student report from the selected hostel. Show print selection	Pass

Table 6 shows the test plan result for the UTHM Hostel Management System non-functional requirements.

Table 6: Non-Functional System Test Plan Result

Section	Expected Result	Actual Result
Operational	The proposed system should work on any web browser.	Pass
	The system should be available when there is an internet connection.	Pass

Table 6: (cont)

Performance	The response time of the system should be less than three seconds.	Pass

5. Conclusion

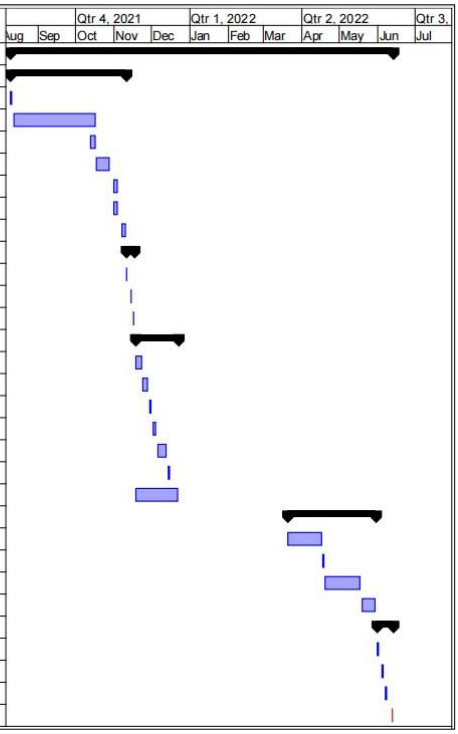
In conclusion, the development of the UTHM Hostel Management System is completed. With the improved functionalities such as the notice module and the complaint module, this system eases the hostel management to manage the hostel activities more efficiently. However, this system has some limitations which can be improved in the future such as the implementation of a push notification function to notify the admin of the student's new application and complaint in real-time.

Acknowledgment

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

Appendix A (Gantt Chart)

	Name	Duration	Start	Finish
1	**UTHM Hostel Management System**	**221 days**	**8/9/21 8:00 AM**	**6/13/22 5:00 PM**
2	**Planning**	**68 days**	**8/9/21 8:00 AM**	**11/10/21 5:00 PM**
3	Propose project title	3 days	8/9/21 8:00 AM	8/11/21 5:00 PM
4	Prepare proposal	47 days	8/12/21 8:00 AM	10/17/21 5:00 PM
5	Define project schedule	3 days	10/13/21 8:00 AM	10/17/21 5:00 PM
6	Complete Chapter 1	10 days	10/15/21 8:00 AM	10/28/21 5:00 PM
7	Existing system review	5 days	10/29/21 8:00 AM	11/4/21 5:00 PM
8	Complete Chapter 2	5 days	10/29/21 8:00 AM	11/4/21 5:00 PM
9	Complete Chapter 3	4 days	11/5/21 8:00 AM	11/10/21 5:00 PM
10	**Analysis**	**5 days**	**11/11/21 8:00 AM**	**11/17/21 5:00 PM**
11	Conduct interview	1 day	11/11/21 8:00 AM	11/11/21 5:00 PM
12	Analyse user requirement	2 days	11/12/21 8:00 AM	11/15/21 5:00 PM
13	Analyse system requirement	2 days	11/16/21 8:00 AM	11/17/21 5:00 PM
14	**Design**	**25 days**	**11/18/21 8:00 AM**	**12/22/21 5:00 PM**
15	Design Data Flow Diagram	4 days	11/18/21 8:00 AM	11/23/21 5:00 PM
16	Design Entity Relationship Diagram	3 days	11/24/21 8:00 AM	11/28/21 5:00 PM
17	Design database, data dictionaries	3 days	11/29/21 8:00 AM	12/1/21 5:00 PM
18	User interface design	2 days	12/2/21 8:00 AM	12/5/21 5:00 PM
19	Develop system prototype	6 days	12/6/21 8:00 AM	12/13/21 5:00 PM
20	Generate test plan	3 days	12/14/21 8:00 AM	12/16/21 5:00 PM
21	Complete Chapter 4	25 days	11/18/21 8:00 AM	12/22/21 5:00 PM
22	**Implementation**	**52 days**	**3/20/22 8:00 AM**	**5/30/22 5:00 PM**
23	Code Implementation 1	21 days	3/20/22 8:00 AM	4/17/22 5:00 PM
24	Review	1 day	4/18/22 8:00 AM	4/18/22 5:00 PM
25	Code Implementation 2	22 days	4/19/22 8:00 AM	5/18/22 5:00 PM
26	Complete Chapter 5	8 days	5/19/22 8:00 AM	5/30/22 5:00 PM
27	**Testing**	**10 days**	**5/31/22 8:00 AM**	**6/13/22 5:00 PM**
28	User Acceptance Testing	3 days	5/31/22 8:00 AM	6/2/22 5:00 PM
29	Identify system improvements	1 day	6/4/22 8:00 AM	6/5/22 5:00 PM
30	Complete Chapter 6	2 days	6/7/22 8:00 AM	6/8/22 5:00 PM
31	System Review	2 days	6/12/22 8:00 AM	6/13/22 5:00 PM



References

[1] Suriya, S., Meenakshi Sundaram, G., Abhishek, R., & Ajay Vignesh, A. B. (2021). Online Hostel Management System Using Hybridized Techniques of Random Forest Algorithm and Long Short-Term Memory. In Advances in Machine Learning and Computational Intelligence (pp. 207-218). Springer, Singapore.

[2] Baffoe, E. A. (2015). Hostel Management System. Saf. Sci, 33(3), 1-6.

[3] Delisle, M. (2012). Mastering phpMyAdmin 3.4 for effective MySQL management. Packt Publishing Ltd.

[4] Russo, N., Wynekoop, J., & Walz, D. (1995). The use and adaptation of system development methodologies. Managing Information & Communications in a Changing Global Environment, Idea Group Publishing, PA.

[5] Kramer, M. (2018). Best practices in systems development lifecycle: An analysis based on the waterfall model. Review of Business & Finance Studies, 9(1), 77-84.

[6] Maulana, F. I., Susanto, V., Shilo, P., Gunawan, J., Pangestu, G., & Budi Raharja, D. R. (2021, September). Design and Development of Website Dr. Changkitchen Diet Catering Using SDLC Waterfall Model. In 6th International Conference on Sustainable Information Engineering and Technology 2021 (pp. 75-79).

[7] Uzun, E., & Buluş, H. N. (2017, October). Object-based flowchart drawing library. In 2017 International Conference on Computer Science and Engineering (UBMK) (pp. 110-115). IEEE.

[8] Creately. (2021). Ultimate Entity Relationship Diagram Tutorial (ER Diagrams). Retrieved from: https://creately.com/blog/diagrams/er-diagrams-tutorial/. [Accessed Dec 26, 2021].