Enhancing Hostel Outing Management through a Usability-Driven Web-Based System: A Case Study of SMK Seri Gading

Siti Aishah Mohd Rahim[1], Rashidah Mokhtar[2,*], Noraisyah Abdl Aziz[2], Mazlyda Abd Rahman[2]

[1] Centre of Computing Sciences, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Terengganu, Malaysia
[2] Centre of Computing Sciences, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Johor, Malaysia

ARTICLE INFO	ABSTRACT
  	Hostel Outing Management System (HOMS) is meant to make it easier to organise student outings at SMK Seri Gading (SEGA) by speeding up the approval and monitoring processes. The current process is manual approach, which uses paper forms and records, has generated problems with tracking, especially during busy times. Therefore, this study's purpose is to design and develop a web-based system to automate outing requests, approvals, and real-time student tracking. Using the adapted waterfall model, a system was developed with PHP, HTML, CSS, and JavaScript, following a secure client-server architecture. The system lets students apply for outings, check the progress of their applications, and see their past records. Guardians get fast notifications and can see the details of their child's outing. Wardens and security staff can easily keep an eye on when students leave and come back and handle outing requests. We used usability concepts to make sure that it was easy to use and that navigation was quick. A mobile app may be added in the future to make things easier to use.

1. Introduction

Schools that provide hostel facilities have a big responsibility in making sure their students are safe, especially when students leave the hostel for outings. SMK Seri Gading (SEGA), a secondary school in Batu Pahat, also faces this responsibility and understands the importance of managing student outings in a proper and organized way. To help make this process easier and safer, a Web-Based Hostel Outing Management System (HOMS) was developed specially for SEGA. This system replaces the traditional manual method and offers a faster and more reliable way for students to apply for outings and for staff to keep track of them. As mentioned by Nawi *et al.,* [1], web-based systems are very helpful in schools because they reduce the amount of paperwork and make it easier for teachers and staff to manage student activities.

With this system, students can apply for outings online, without needing to wait in line or meet the warden in person. Their outing request is sent straight to the warden, who can approve

---

* *Corresponding author.*
E-mail address: rashi271@uitm.edu.my

or reject it through the system. After a decision is made, both the student and their guardian will get a notification. Guardians can also log in to view full details of the outing such as the purpose, time, and expected return. This makes it easier for parents to know where their children are and gives them peace of mind. According to Vijitha *et al.*, [2], systems like this not only improve student safety but also help parents feel more involved and informed.

Recent research shows how important it is for schools to use digital tools. Diyaolu *et al.*, [12] said that digital hostel systems, such as the E-College Monitoring System, make it easy to keep track of students' movements in real time, immediately notify guardians, and reduce the number of mistakes made while managing large groups of students. Anisyah *et al.*, [16] stated that using information technology makes school administration much better by improving attendance, scheduling, communication, and reporting. This lets administrators make better decisions faster.

Ghavifekr and Wong [17] research shows that strong technology leadership by school leaders in Malaysia makes it easier to use digital resources effectively, which leads to more engaged teachers and better student outcomes. A comprehensive review of the literature done in 2023 backs up the idea that digital tools might help students stay motivated, get parents more involved, and keep an eye on how well schools are doing. Bhat *et al.*, [18] came up with a method that uses RFID-based ID cards to automate attendance at hostels and improve access control. This greatly increases the efficiency and safety of student accommodations.

The system also benefits school staff. Wardens can manage outing requests faster, check student history, and even add demerit points if students return late. Security guards can verify students more quickly, and administrators can generate reports easily. All this helps make the outing process smoother, more accurate, and less stressful for everyone. As explained by Nurshaza [3], digital outing systems help reduce human errors and save time, especially when many students are involved. By introducing this system, SMK Seri Gading has taken an important step in improving hostel management through technology, making it safer and easier for both students and staff.

This system helps wardens manage student movements more easily. They can see outing histories, approve requests faster, and act if students come back late. Security guards can quickly confirm which students have permission to leave, and system administrators can monitor user activity and generate reports. HOMS makes the entire outing process more efficient, secure, and easier for everyone involved. By using this system, SMK Seri Gading can improve student safety, reduce paperwork, and improve communication between students, wardens, and guardians.

1.1 Key Concern

The manual process used for student outing management at SMK Seri Gading is inefficient, inaccurate, and mostly not secure. Students must complete the outing card requests and wait for a manual approval from a warden. If it is a busy time, such as a school holiday, where many students are processing out the same day, it can be delayed even longer. If the warden is absent or not busy, students must wait until they are present at the office. This may lead to increased waiting and frustration, especially when students must go out for urgent matters such as extra class attendance or their issues. According to Nawi *et al.*, [1], a manual method for providing approvals affects accuracy and efficiency of the system while also creating a system in which delays are unavoidable. If the process for approvals is in the hands of only one person, such as a warden for example, delays in movement will become significant.

Additionally, security guards track student movements by physically checking them in and out while making a record of each check-in and check-out in a logbook, which substantially increases

the chances of lost data or mistakes. Vijitha *et al.,* [2] highlighted the problems that can arise in paper-based methods for collecting data, showing that when data is lost or inaccurate, a breach occurs in the safe record-keeping of students, thus posing a risk to their safety. The manual intervention can sometimes become cumbersome, especially in emergencies when the warden must glance through physical logbooks to extract the relevant information.

This manual reporting mechanism slows down the process and may come with human errors that may impede the actual timely response to any matter. Additionally, the lack of real-time tracking of the leave system makes it really challenging to track the whereabouts of some students [4]. Based on the items discussed above, an upgrade of the current practice is needed, potentially systematizing the process to enhance data security, enhance approval, and ultimately facilitate timely and improved management of student leave requests.

2. Methodology
2.1 Adapted Waterfall Model

A project management methodology is a group of principles, tools, and techniques to plan, execute, and control your projects. Project management methodologies are established methods that help project managers manage projects by directing teamwork, organizing tasks, and promoting teamwork and collaboration [5]. In software development, there is a waterfall method that reflects a sequence of closely ordered stages. Each stage must be completed before the next stage can begin, to ensure that a stage does not have to be done again [6].

The Waterfall Model was developed in 1970 by Royce [9] and consists of six phases mainly system requirements, software requirements, analysis, program design, coding, testing, and operations. The original Waterfall Model was modified to fit the specific needs of the project for the Web-Based HOS for the SEGA project. In particular, a documentation phase was added because the project lasted for one year (two semesters), which will allow for comprehensive documentation that may be used in the future for other similar projects. The Adapted Waterfall Model proposed by Mokhtar *et al.,* [7] consists of planning, analysis, design, implementation, testing, and documentation as in Figure 1.

The planning phase of the project involves gathering, analyzing, and organizing information. In this phase, an interview with the school warden was conducted to understand the current process of hostel outing management. A flowchart was created to help visualize the current business processes and to identify areas that require change.

In the analysis phase, a form of approach for system development was picked; consequently, there was an attempt to ensure that all the expressed and written requirements are understood. For this project, the adapted waterfall model was chosen because it is a well-defined and structured approach with a sequence of steps, which provides a clear distinction between expressions of user requirements and expressions of system requirements. According to Przybilski and Tuunanen [8], user requirements, which are expressed in natural language and then provide a description of the system in terms of its services, features, and limitations, and may provide diagrams of the system, are fashioned. In comparison, the system requirements are more formal and provide the features and functional behaviour of the system. The system requirements were divided into functional and non-functional requirements. Functional requirements define the services, inputs, and responses of the system. Non-functional requirements define constraints on the developers, such as performance and security.

The design phase then takes these requirements and converts them into specifications. In order to model the structure of our system and show how it works, we used several tools, namely

the Entity Relationship Diagram (ERD), Data Flow Diagrams (DFD) and Site Map. The next phase, the implementation, will concentrate on coding the system. Following the implementation will be the testing phase to ensure everything appears to work as expected and no issues arise. Finally, there is the documentation phase, in which all the important information is compiled, and the final project report is assembled.
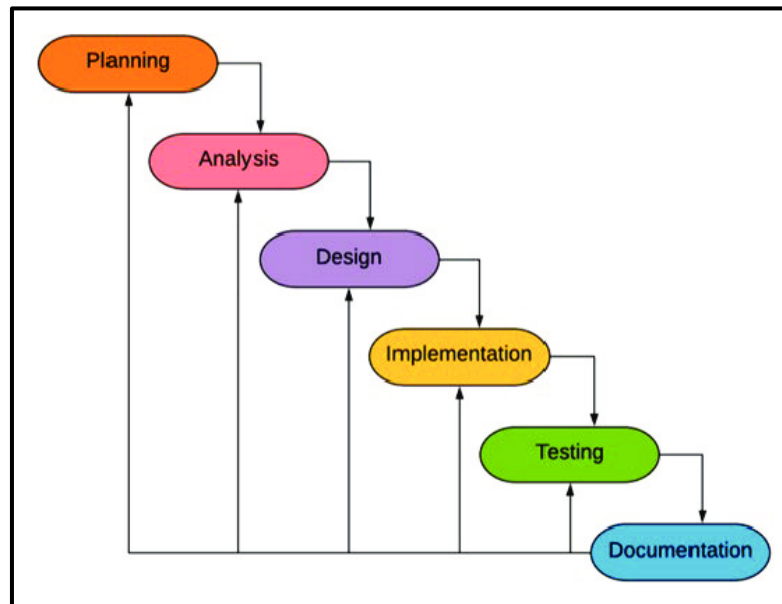

Fig. 1. Adapted Waterfall Model [7]

2.2 Context Diagram

Context diagrams play a critical role in system design as they visually show the interaction between the system and its users in relation to the external users, while indicating the boundary of the system. Ultimately, context diagrams serve as an excellent way to understand the flow of data between the system and users. In the case of the Web-Based Hostel Outing System (HOS) for SMK Seri Gading (SEGA), five key user types interact with the system which are students, wardens, administrators, guardians, and guards.

For students, the system allows them to register, log in, submit outing requests, and view their outing status, history, and demerit records. Wardens handle outing approvals, review reports, and monitor student demerits. Administrators are responsible for managing accounts and system logs. Guardians can register, log in, and track their child's outing status. Guards, on the other hand, only need to input a student's IC number, with the system automatically recording the outing time. User registrations, outing requests, and approvals all have data flows that can be seen in the context diagram as illustrated in Figure 2, which provides an overview of what the system does.
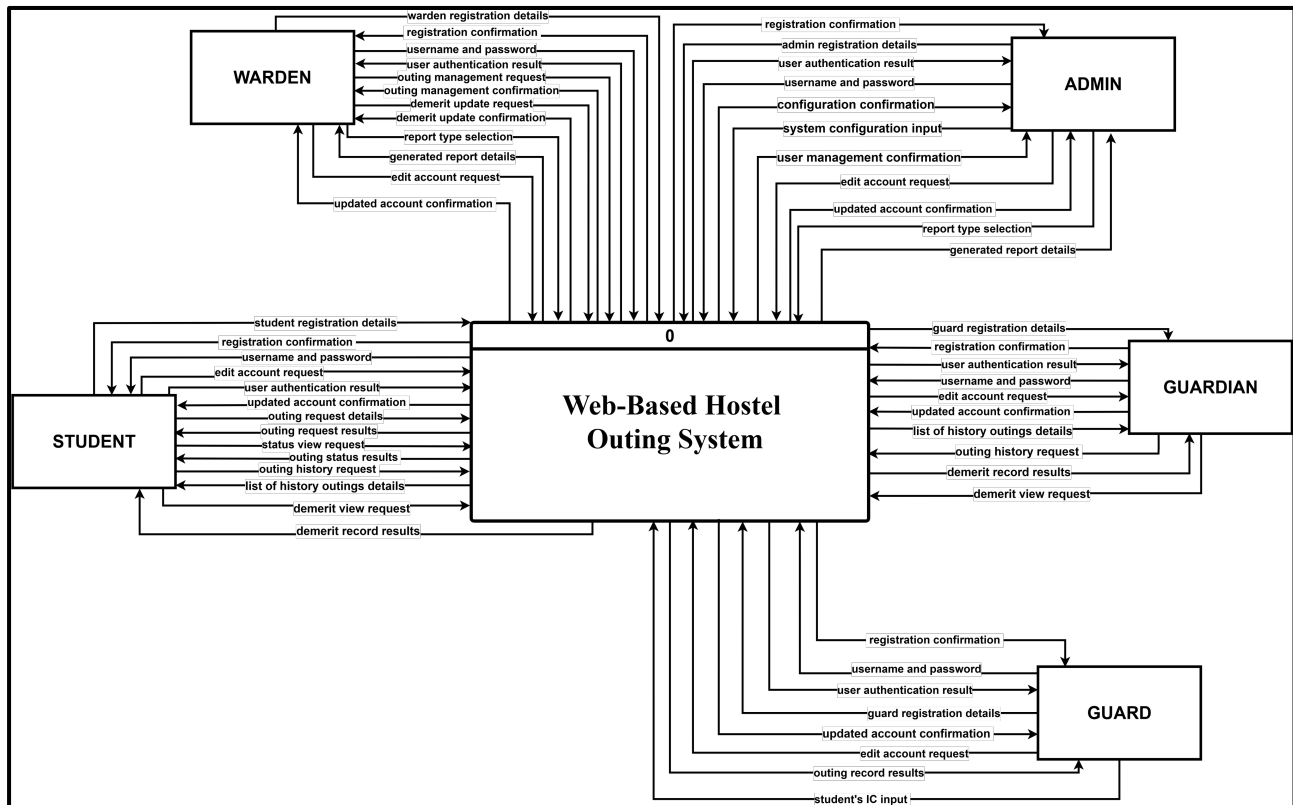
Fig. 2. Context Diagram

2.3 Data Flow Diagram

Data Flow Diagram (DFD), created by Larry Constantine, is a graphical representation of data that describes the flow of data throughout various business processes or functions. DFD is helpful in visualizing how data is processed by a system. The DFD shows processes, data stores, external entities, and data flows and illustrates how they relate to one another [10].

In the Web-Based Hostel Outing System (HOS) for SEGA, the DFD, as illustrated in Figure 3, explains the processes in the system that involve five main users: students, wardens, guards, guardians, and administrators. There are twelve processes and ten data stores that hold important information, such as details of students, wardens, guards, guardians, admins, and outing records.

The process initiates with user registration, where the user's details are captured and stored in the database. Upon successful registration, users that include students, wardens, guards, guardians, and administrators who can log into the system to perform their respective tasks. Students are able to submit outing requests, which are subsequently stored within the system for further processing. Wardens manage the demerit records of students, while guards monitor the students who are on outings. All these records are stored in the outing information database. The system also keeps a history of past outings and generates reports for future reference. Administrators can adjust system settings, fix any problems, and review activity logs. This DFD provides a clear and simple view of how the Web-Based HOS operates, making it easier for everyone involved to manage hostel outings efficiently.
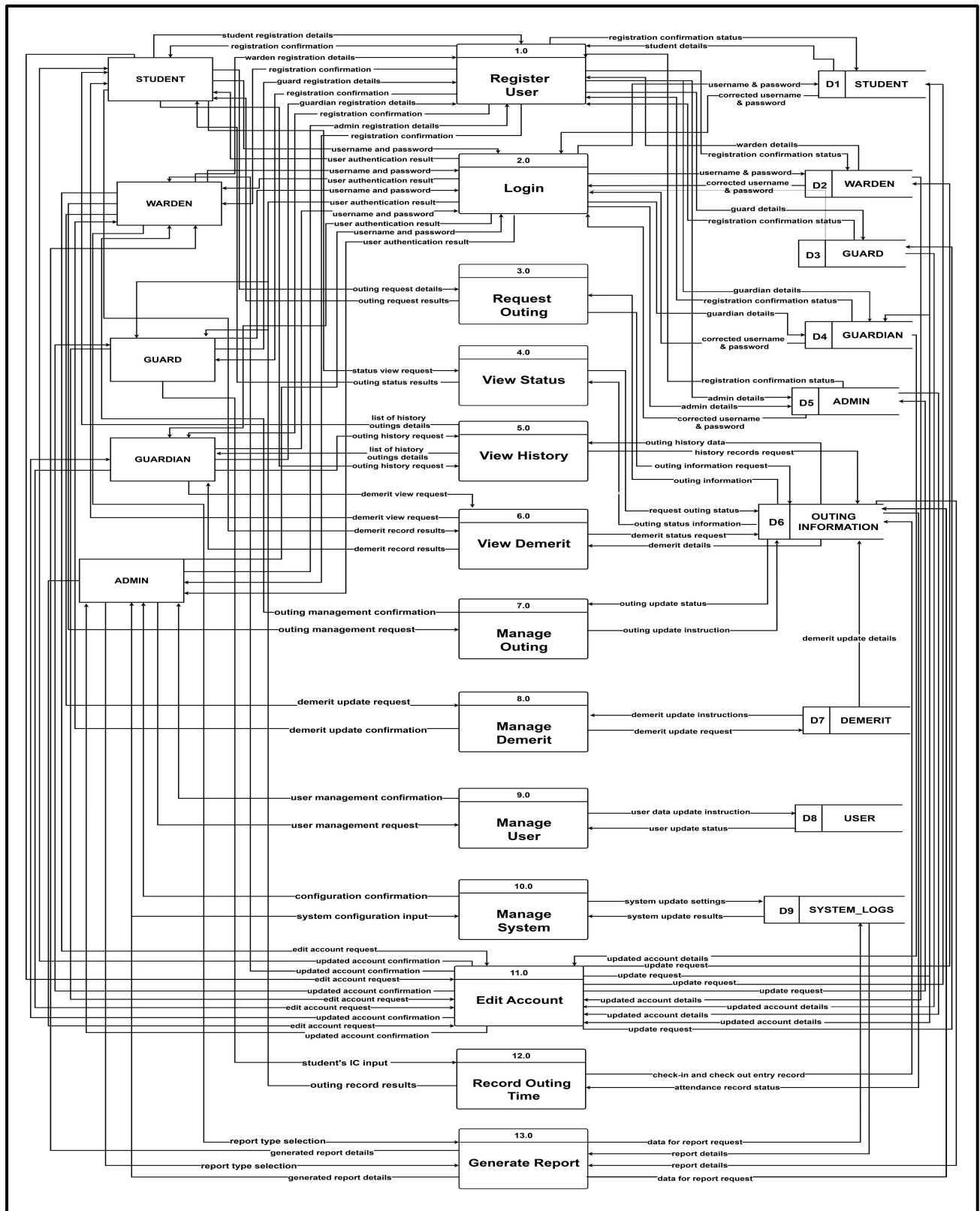
Fig. 3. Data flow diagram

2.4 Entity Relationship Diagram

An Entity Relationship Diagram (ERD) is a commonly used tool to plan how data will be structured. It begins by identifying the key components (called entities), their characteristics

(attributes), and their connections (relationships). Once you have created the diagram, it is mapped into a formal structure known as a relational model, defined by a certain set of rules. This structured model is later used to build the actual database, depending on the software used [11]. For the Web-Based HOS ERD developed for SEGA, as shown in Figure 4, each attribute is specific to its entity and stores particular data. Some of these attributes act as foreign keys, which link different entities but do not directly define the relationship type. These relationships represent how the entities are related, such as one-to-many or many-to-one. The ERD is created to guide the development of the HOS system. The main entities in the diagram include Student, Guardian, Warden, Guard, Admin, Outing Information, Users, System Logs, and Demerit.
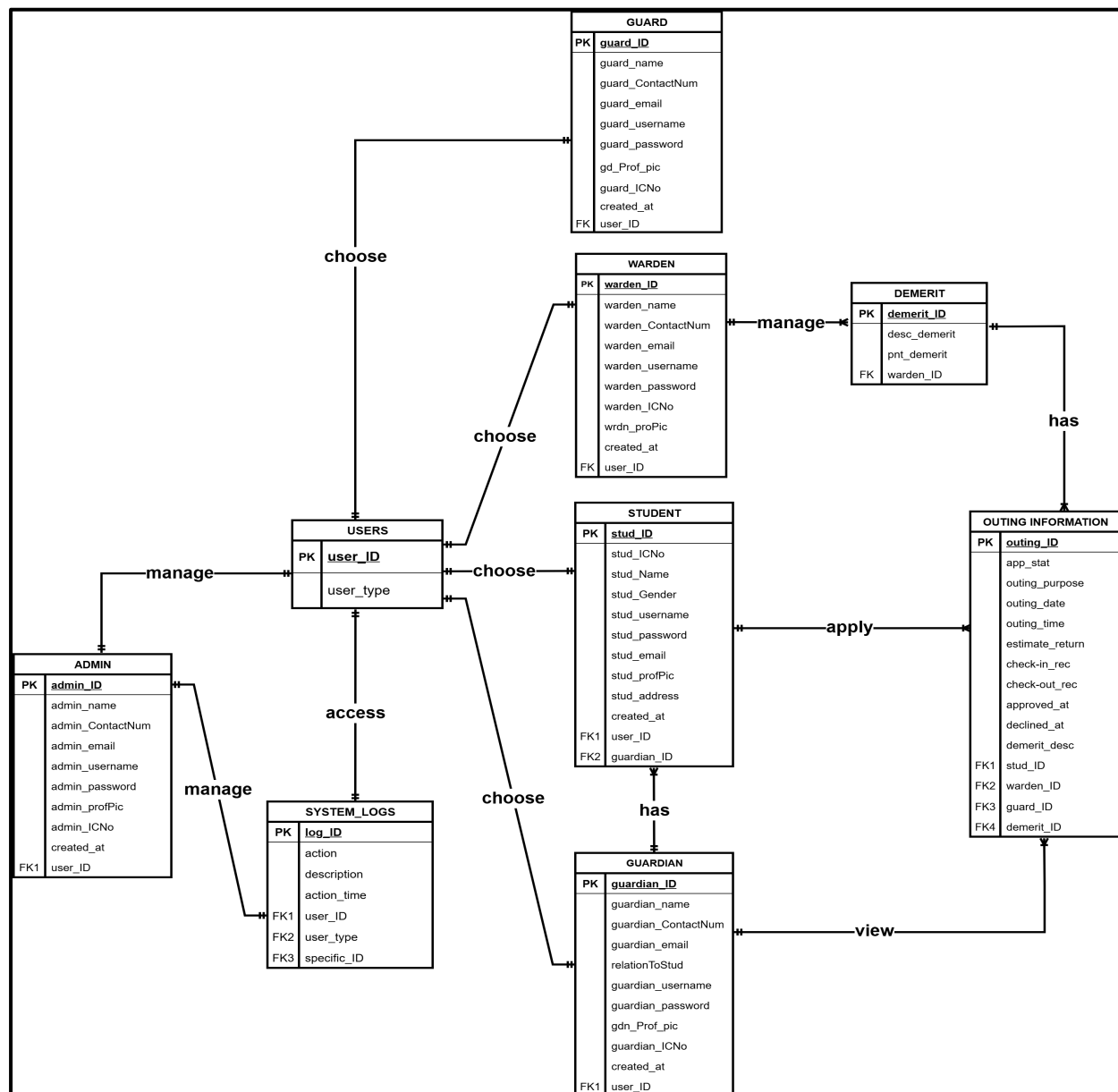

Fig. 4. Entity relationship diagram

2.5 Site Map

A site map is a diagram that represents the structure of a website. Its main purpose is to provide a clear understanding of the website's information structure during its development. It

also partially illustrates the navigation structure. In some cases, site maps are included in the final version of the website, but these often differ significantly from the internal site maps used during development [13]. Figure 5 displays the site map of Web-Based HOS for SEGA that consists of Student, Warden, Guard, Guardian, and Admin.
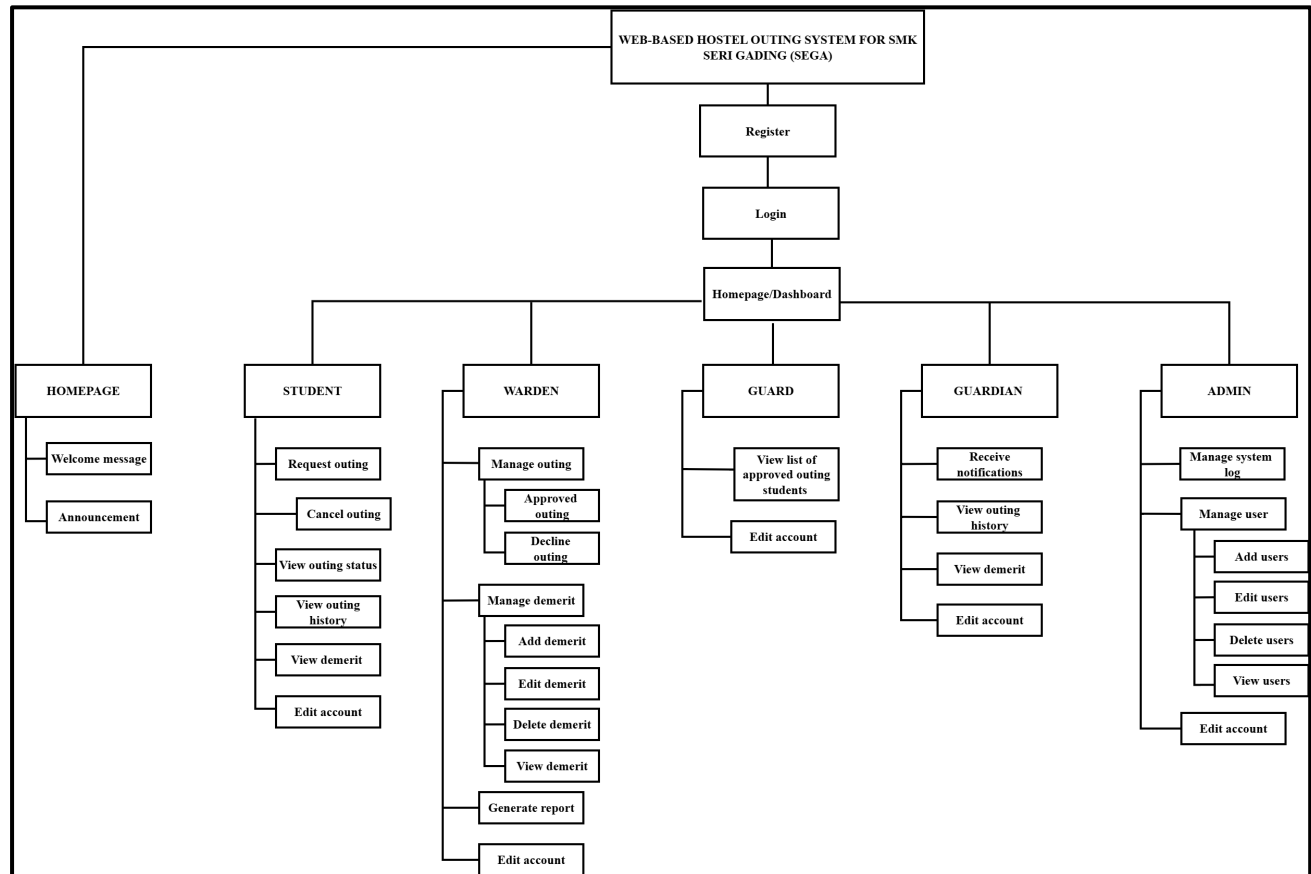

Fig. 5. Site map diagram

2.5 User Interface

The part of a system that users engage with is called the user interface (UI). It is the point of interaction for humans and computers, allowing them to convey ideas and perform the necessary action [14]. The UI is the mediator that allows users to convey their thoughts and interact with the system in an efficient manner, both physically, perceptually, and conceptually [14]. A well-designed and user-friendly interface not only simplifies communication with the system but also encourages users to stay longer and engage more with the platform [15]. For the Web-Based HOS for SEGA, all 10 usability principles were applied in designing the user interface, as illustrated in Figure 6 until Figure13.

Figure 6 shows the public main page is the system's landing page, and it lets users get to the system based on their roles. It has five clearly labelled buttons for students, wardens, guardians, guards, and administrators to use. This layout makes it easy for users to find their access point and lowers the chance of getting lost. The design is minimalistic, applying a clean and organized structure with intuitive visual elements to guide users from the initial stage of interaction.
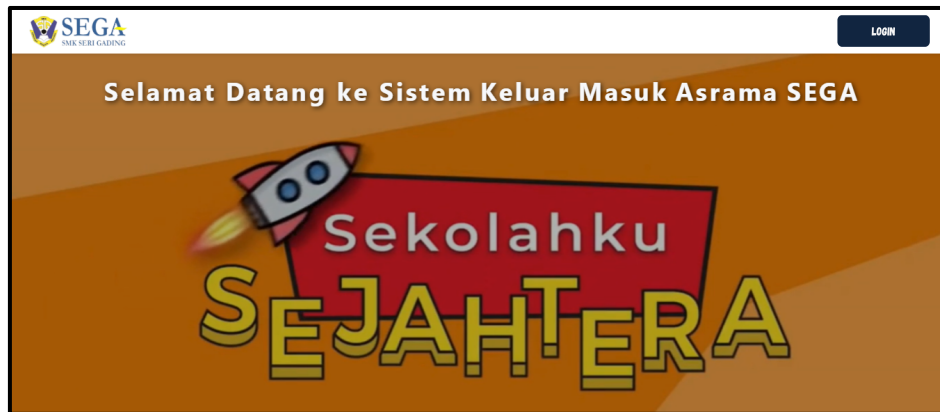
Fig. 6. Public main page interface

The student dashboard as shown in Figure 7 and Figure 8 is a central and easy-to-use place for students to handle their outing applications. Students can use this interface to request outings, check the status of their applications, see their past outings, and keep track of any demerit records. This system gives students more power by letting them see and control their outing records without having to go to the wardens in person or wait for verbal updates. The fact that students can track their own behaviour and punctuality over time helps them develop a sense of responsibility and self-discipline. Also, this digital method protects your privacy more and cuts down on wait times during busy times like weekends and holidays.
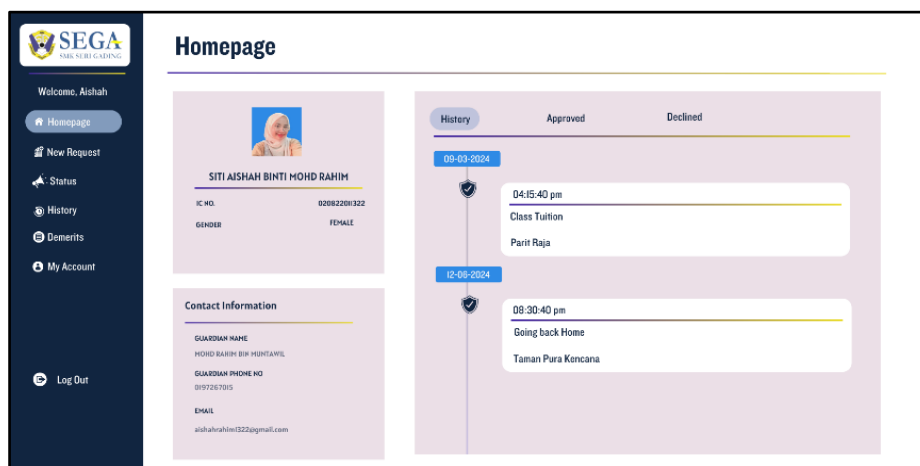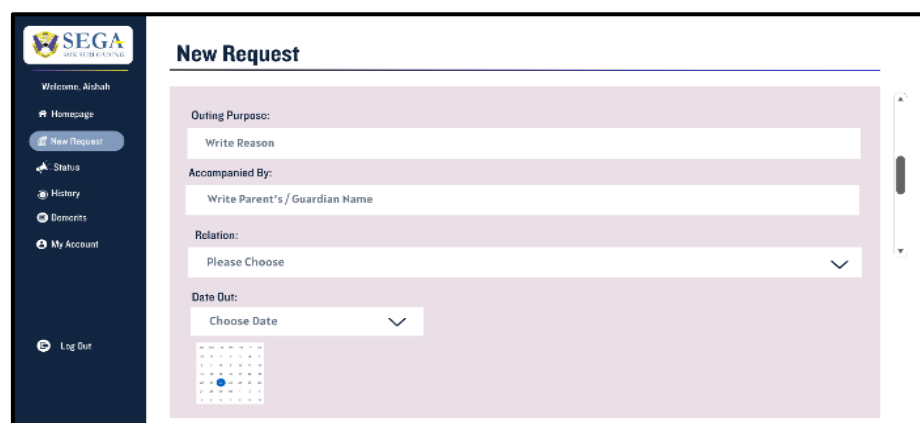

Fig. 7. Student homepage interface

Fig. 8. Student request outing interface

The warden dashboard as illustrated in Figure 9 and 10 is meant to make the approval process easier and more organised, which will help wardens do their jobs better. This interface lets wardens quickly see and respond to outing requests, keep track of outing schedules, and keep an eye on demerit points. Wardens can quickly handle a lot of requests, especially during school breaks or exam times, thanks to structured lists and decision buttons. This is good for wardens because it cuts down on paperwork and mistakes made by hand, and it also makes sure that all approvals are recorded and easy to find. Adding historical records makes it easier to make decisions because wardens can look at past actions or repeated offences when deciding on new requests.
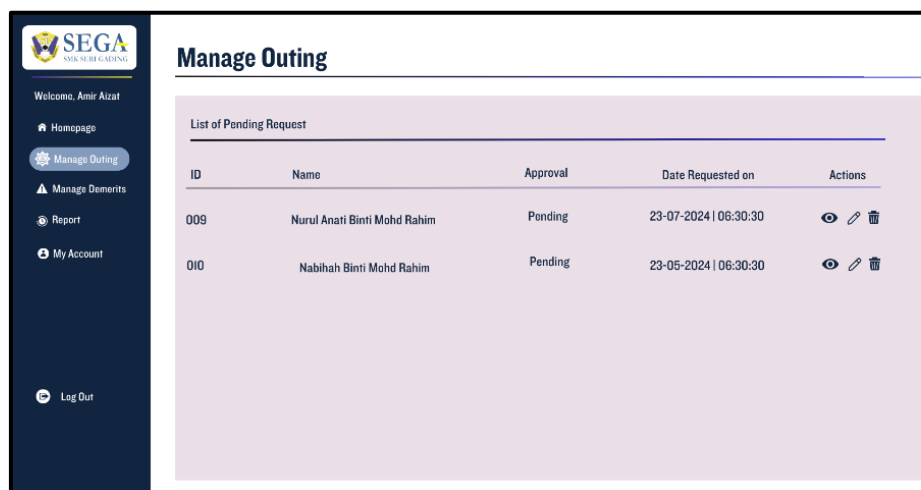

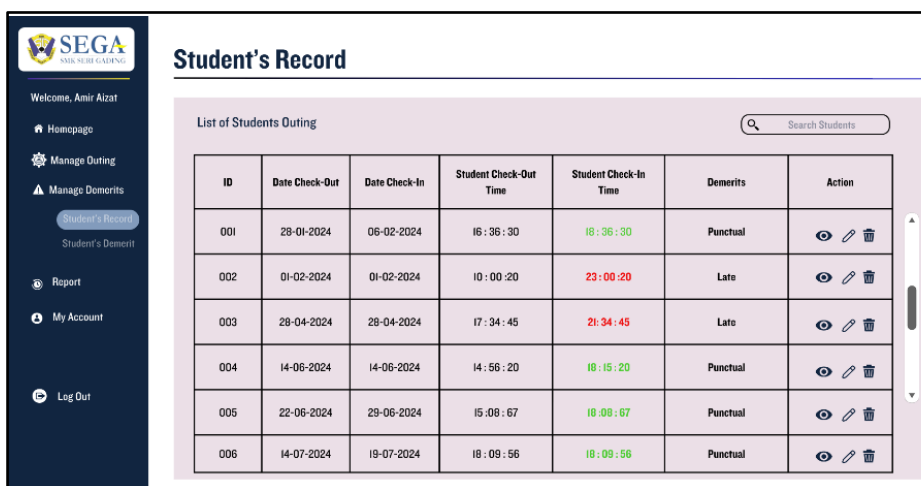**Fig. 9.** Manage outing interface

Fig. 10. Manage demerits interface

In the meantime, the guard interface is very important for keeping students safe when they move around as shown in Figure 11. This module lets guards check if students have permission to go out by typing in their student Identity Card (IC) number. Once the user is verified, the system automatically records the time of check-in or check-out. This digital tracking system takes the place of paper logbooks and lowers the risk of mistakes, unauthorised exits, or lost data. It helps guards because it speeds up the process of checking people in, which is especially helpful when there are a lot of people leaving at once. More importantly, it makes the hostel safer and keeps track of all student movements in real time so they can be looked at later.
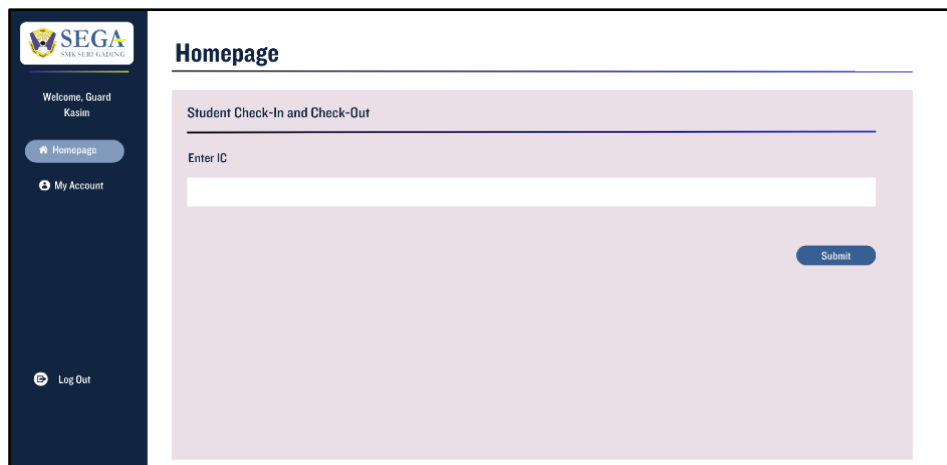
Fig. 11. Guard homepage interface

Figure 12 shows the guardian interface makes it easier for parents to be involved by giving them direct access to information about their child's outings. Guardians can see the approved outings, the reasons for the outings, the times when the kids will be back, and any demerit points that go along with them. Parents feel better about their child's safety when they know what's going on, and this encourages them to get involved in school-related issues. Guardians don't have to rely only on what students tell them anymore, which can sometimes be wrong or late. They can get notifications and see outing data in real time. This feature shows that the school cares about the students' well-being and holds them accountable, which builds trust between the school and the guardians.
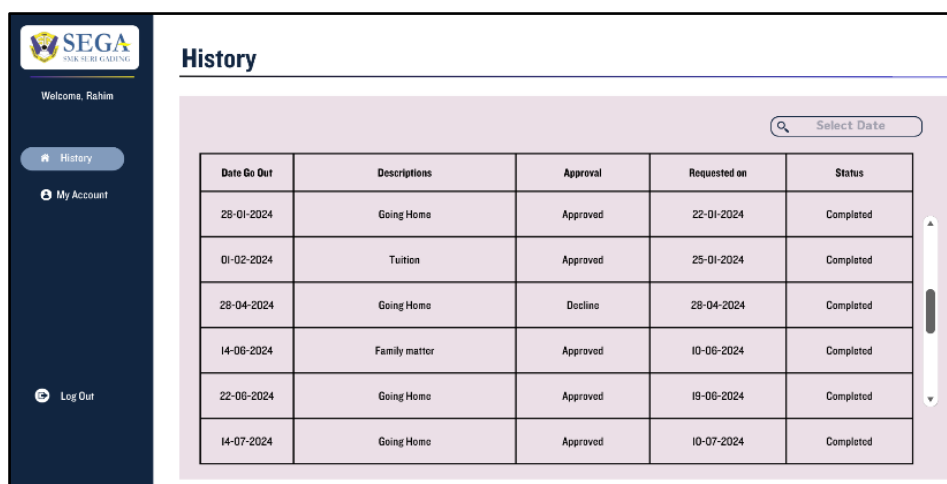

Fig. 12. Guardian page interface

The admin interface as illustrated in Figure 13 gives system administrators full control over the user environment and how the system works. Administrators can add new users, change existing accounts, check system logs, and fix technical problems. This interface lets you manage users in all roles, making sure that access is given and kept up to date. The best thing about this dashboard is that it can keep the whole system safe and secure. The admin makes sure the platform runs smoothly and stops people from making changes they shouldn't by keeping an accurate record of system activities and enforcing access control.
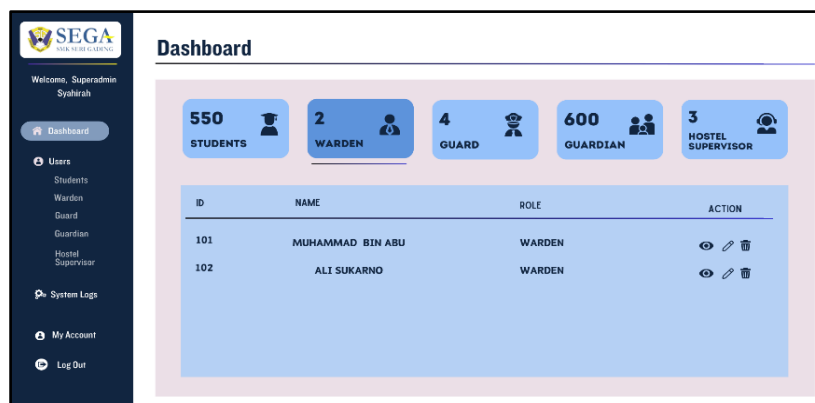
Fig. 13. Admin dashboard interface

In contrast, each interface is carefully made to fit the tasks and information needs of the users. This role-based approach helps make the system more efficient, secure, and user-friendly, which is in line with the main goal of making SMK Seri Gading's hostel outings more modern and better managed.

4. Conclusions

This study was about creating a Web-Based Hostel Outing System for SMK Seri Gading to fix problems with the current manual procedure, like delays, mistakes, and not being able to follow things in real time. The design was based on usability considerations and was put together using the Adapted Waterfall Model to make sure it was clear and organised. The system was designed to work for different types of users, such as students, wardens, guardians, guards, and administrators. Each type of user had their own set of access and functions, such as submitting outing requests, keeping track of approvals, seeing outing history, and making reports. We produced important design elements including context diagrams, data flow diagrams, entity relationship diagrams, and user interface layouts to show how the system will work in a clear way. These technologies helped us see how information flows and make sure the system satisfies users' demands. This full design gives a strong base for future implementation, even though the development phase hasn't started yet. In the future, the system can be further developed and tested to enhance the overall management of hostel outings, ensure greater safety and efficiency for school operations, and enhance communication among users. To make it easier for people to use, a future update might provide a mobile application.

References
[1] Nawi, Nur Ainatul Mardiah Mat, Nur Shaliza Sapiai, Siti Aswani Mohd Ghazali, Nur Hazelen Mat Rusok, Fara Zuliana Zulkifli, and Fatin Masturina Mazlan. 2022. "Developing an E-College Monitoring System as a Web-Based Monitoring Tool Application," 25. https://doi.org/10.3390/proceedings2022082025
[2] Vijitha, B, G Meghana, B Akshitha, and G Naresh. 2024. "STREAMLINING HOSTEL OUTING MANAGEMENT PLATFORM" 14 (02): 198–206.
[3] Nurshaza, Mazlan. 2015. "Boarding School Outing System," no. May, 1–58.

[4]   Kah Wen, Matthew Khoo, Nabihah binti Ahmad, and Siti Hawa binti Ruslan. 2020. "Arduino Based Outing and Attendance System for Boarding School Students." Indonesian Journal of Electrical Engineering and Computer Science 20 (2): 1053–61. https://doi.org/10.11591/ijeecs.v20.i2.pp1053-1061

[5]   Kah Wen, Matthew Khoo, Nabihah binti Ahmad, and Siti Hawa binti Ruslan. 2020. "Arduino Based Outing and Attendance System for Boarding School Students." Indonesian Journal of Electrical Engineering and Computer Science 20 (2): 1053–61. https://doi.org/10.11591/ijeecs.v20.i2.pp1053-1061

[6]   Kah Wen, Matthew Khoo, Nabihah binti Ahmad, and Siti Hawa binti Ruslan. 2020. "Arduino Based Outing and Attendance System for Boarding School Students." Indonesian Journal of Electrical Engineering and Computer Science 20 (2): 1053–61. https://doi.org/10.11591/ijeecs.v20.i2.pp1053-1061

[7]   Mokhtar, Mazita, Nor Hasnul Azirah Abdul Hamid, and Nurul Athirah Ahmad Nazri. "SEATROBS: Development of Sealife Travel & Tour Online Booking System Using Usability Theory."

[8]   Przybilski, Michael, and Tuure Tuunanen. "From rich user requirements to system requirements." (2007).

[9]   Royce, Winston W. 1970. "Managing the Development of Large Software Systems (1970)." Ideas That Created the Future, no. August, 321–32. https://doi.org/10.7551/mitpress/12274.003.0035

[10]   Zhao, Yun, Huayou Si, Yulin Ni, and Hengnian Qi. "A service-oriented analysis and design approach based on data flow diagram." In *2009 International Conference on Computational Intelligence and Software Engineering*, pp. 1-5. IEEE, 2009. https://doi.org/10.1109/CISE.2009.5365568.

[11]   Hingorani, Kamal, Dexter Gittens, and Nicholas Edwards. "Reinforcing database concepts by using entity relationships diagrams (ERD) and normalization together for designing robust databases." *Issues in Information Systems* 18, no. 1 (2017): 148-155. https://doi.org/10.48009/1_iis_2017_148-155

[12]   Diyaolu, Akorede M., Omolara B. Abodunrin, Abdullateef A. Adedamola, Rotimi S. Ogunode, and Oluwatoyin Omoloba. "Development of an E-Based Hostel Management System." *International Journal of Innovative Science and Research Technology* 9, no. 6 (2024): 2456-2165. https://doi.org/10.38124/ijisrt/ijisrt24jun147

[13]   Newman, Mark W., and James A. Landay. "Sitemaps, storyboards, and specifications: A sketch of web site design practice." In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*, pp. 263-274. 2000. https://doi.org/10.1145/347642.34775

[14]   SARDJONO, WAHYU, and WIDHILAGA GIA PERDANA. "Interactive User Interfaces in the Digital World Make the Application Attractive and Easier for User Access." *Journal of Theoretical and Applied Information Technology* 102, no. 1 (2024).

[15]   Pratama, Muhammad Aulia Taufik, and Ahmad Taufiq Cahyadi. "Effect of user interface and user experience on application sales." In *IOP Conference Series: Materials Science and Engineering*, vol. 879, no. 1, p. 012133. IOP Publishing, 2020. https://doi.org/10.1088/1757-899X/879/1/012133

[16]   Nur Anisyah, Taufiq Rohman, Sesti Novalina, Putri Rahmi Virani Lubis, Arsanur Rahman, Rika Aryani. 2025. "The Role of Information Technology in Enhancing the Effectiveness of Education Management Tools." Journal of Financial Technology 14 (4): 21–35.

[17]   Ghavifekr, Simin, and Seng Yue Wong. "Technology leadership in Malaysian schools: The way forward to education 4.0–ICT utilization and digital transformation." *International Journal of Asian Business and Information Management (IJABIM)* 13, no. 2 (2022): 1-18. https://doi.org/10.4018/IJABIM.20220701.oa3

[18]   Bhat, Shravan. "Enhancing Room Security and Automating Class Attendance Using ID Cards." *arXiv preprint arXiv:2307.03926* (2023). https://doi.org/10.48550/arxiv.2307.03926