

## STUDENT MANAGEMENT SYSTEM WITH RESULT ANALYSIS

WANGCHAOFAN,

DALBIR SINGH A/L VALBIR SINGH

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia, 43600 UKM  
Bangi,, Selangor Darul Ehsan, Malaysia

### ABSTRACT

Student management System is a platform to help students more quickly understand and familiar with the whole University. Student achievement is an important index to measure whether a student has achieved anything in college. However, many University student management systems are developed from the perspective of developers and managers, which will lead to the system not meeting students' needs. For example, the web page's design is too cumbersome to allow students to quickly find the content and functions they need. Moreover, the proposed system does not deeply process much data such as students' grades, which makes it impossible for universities to intuitively understand students' information through charts and other means. The back end of this proposed system uses java as the main development language, MVC as the development framework, MYSQL database, and the front end uses HTML, CSS, and JAVASCRIPT to develop web pages. The system adopts the waterfall model, which can find and solve problems timelier. In this system, students can register courses can also delete the registered courses, but also can provide students with laboratory appointments and book borrowing management pages. The results of students will be directly displayed through charts and other ways to show the learning status of students. Students can more easily solve the problems encountered and the services provided by the University. Lecturers can also get a more complete picture of students based on their performance charts. The system can promote collaboration between lecturers and students.

Key word: University, student information management system, JAVAWEB, Result analysis.

## INTRODUCTION

### 1.1 BACKGROUND

With the advent and development of the “Internet+” era. Informatization of educational administration is an important part of building an intelligent University (Education Informatization 2.0 Action Plan 2018). In University, educational administration is a tedious thing. Every year, new students are admitted, and they are faced with the problems of registering for courses, viewing the course room, generating the schedule, adjusting the course, checking the final grade, and making an appointment for the lab or study room. And with the popularization of higher education, more and more people go to university to study. The traditional educational management mode has been unable to meet the current function of upgrading and maintenance (M. N. Habib, W. Jamal, U. Khalil, and Z. Khan 2021). As a part of the computer application, the educational management system has many people’s manual incomparable advantages. For example, quick retrieval, easy query, high efficiency, good reliability, large storage capacity, strong confidentiality, long life, and low-cost advantages. Whether the evaluation of old students or personnel changes, the University management system can greatly help new students understand the University and University things. For student grades and lecture evaluations, the system generates charts based on scores and check-ins. University administrators can make horizontal and vertical comparisons visually according to the chart, manage and analyze the information, and help University and lectures grasp the situation of students. For example, on Sistem Maklumat Pelajar(SMP) website, the only useful function for students is to register for courses and check grades. Students could not check the sign-in status of each course or apply for leave to the lecturer. There is no way to visually view the form of charts, and analyze their scores and subjects, as well as the need for progress. The problem with the educational administration system is that its function is not sound enough. Many

systems are designed from the perspective of managers, who cannot provide support for student-related activities, and it will be difficult for administrators to supervise and ensure the needs of students (R. G. Hadgraft and A. Kolmos 2020). Therefore, it is very necessary to add the appointment laboratory and lectures in the educational administration system to check the sign-up performance for this semester and to show the results with charts.

## **1.2 PROBLEM STATEMENT**

Student management is a very important aspect of the University system. A college student management system is necessary for improving students' knowledge levels, providing good education, and updating students constantly. However, if the process is done manually, there will be big problems between managing students, educating students, managing lecturers, and managing staff. It takes a lot of energy and a considerable amount of time for college students to understand a very small piece of information. However, if the process is done manually, the person concerned is likely to miss the information.

The integration of digital technology with education influences teaching and learning practices and allows access to data, primarily from online learning environments, that can be used to enhance student learning (Viberg, Hatakka, Balter, & Anna, 2018). The traditional educational management system has not been applied to the data information of many students. Because the data is not deeply processed, the managers cannot dig out the hidden knowledge from the data. Due to the lack of visual data processing in the face of their results, many students do not know their strengths and weaknesses, and cannot fully understand themselves, so they do not know the future research or work direction. The information will help to identify overall trends in the University's performance and provide constructive feedback for individual students (Cambridge Assessment International Education 2023).

It makes sense for students and administrators to categorize, correlate, predict, evaluate, and visualize data such as attendance rates, final grades, homework grades, and quiz scores. Through the above analysis process, students and administrators can be provided with correct information to help students understand themselves and help

administrators to provide higher education quality. In addition, it can be used to provide tailored learning pathways or assessment materials for individual students, based on their past achievement patterns or affinity (Tomasevic et al., 2020).

### **1.3 OBJECTIVE**

Create a student management system website to allow students to register and cancel courses and generate charts to display student grades and build the lecture's side and the administrator's side for the whole system.

Functional and non-functional testing of the system to ensure that the system can perform the planned work and that the interface is clean and aesthetically pleasing.

### **1.4 SCOPE**

Based on all the functions designed and planned to provide, the system is divided into two modules, the administrator module, and the student module. Each module has a different SCOPE, as shown below:

#### **a) Administrator module**

Has the highest level of access other than the developer to manage the entire site. Admin can add colleges, majors under the colleges, and add different classes for each major. Admin can add or delete students and lectures. On top of that, courses can be added and deleted, and lectures can be assigned which classes to go to. On the home page of the administrator's end, you can also see statistical charts that reflect in real time the number of students corresponding to different majors in the University. For the site's infrastructure, backend data, operational structure, etc. Administrators are not authorized to view and modify.

#### **b) Lecture module**

The extent of a lecture's authority is to be able to view the courses he or she has been assigned by the administrator for the semester. It is possible to view how many students are enrolled in each course. At the end of the semester the lecture can grade each student enrolled in the course.

#### **c) Student module**

Students have relatively low privileges and are not eligible to modify data on the website. Students can view relevant content on the website, check enrolled

courses and test scores. One week prior to the start of the University year, students may register for and cancel courses. Students may not change grades or courses.

### **METHODOLOGY**

The system will adopt the Waterfall Model for development. The reason is that the system in the design is small and medium-sized, and in the development process the demand changes will not be too big, the function and requirements are clear, so more in line with the development needs of the system. You can make a development plan before the development, and then build the system step by step according to the order of the model. If you run into any problems during development, this model ensures that changes will not go wrong when they do. Therefore, Waterfall Model is the best development model integrating various constraints and restrictions.

The existing student management system is largely underutilized and unable to meet the needs of students. The main purpose of the requirements analysis activity is to analyze the gathered requirements to remove all ambiguities, incompleteness, and inconsistencies from the gathered customer requirements and to obtain a clear understanding of the software to be developed ( Webeduclick, 2022). Since the main users of this system are students, as much as possible, the user requirements will be considered from the students' point of view.

Functional requirements are essential to your product because, as they state, they provide some sort of functionality (Lane & Krüller, 2022). functional requirements are essential to your product because, as they state, they provide some sort of functionality (Lane & Krüger, 2021). Non-functional requirements are also extremely important as they define the quality of the system and ensure the safety, availability, and effectiveness of the entire system (Leffingwell, 2021).

After listing the requirements of the system, this chapter will define the entire system using an object-oriented approach. The overall architecture of the system and the interfaces between each function are defined using example diagrams and specific requirements. A timing diagram is used to define the constraints of the system and the logic between functions. Finally, the entire flow of the system is defined using the activity diagram.

Software design specification is a detailed plan for developing a piece of software (How to Create Software Design Documents, 2020). This chapter will plan and design the student management system in detail. Complete the goals mentioned in Chapter 1 and abide by the constraints. Combined with the second chapter to compare and improve the existing system proposed solutions. Completed the architectural design and

database design of the system. And use algorithms that can further optimize system performance. Finally, the interface of this system is designed and combined with the previous conclusions obtained by comparing the system.

This system will use javaweb as the core technology of system development. It gives the power to launch full-featured applications with a single click from your Web browser (Sobral, 2019). Use Spring MVC to build the structure of the system. On this basis, this chapter will also build the database and optimize the algorithm of the system.

## RESULTS AND DISCUSSION

Figure 1 login page shows the landing page for the project. There are three options available Administrator, Student & Lecture. And need to fill in the correct username and password and captcha to enter the page after successful verification.

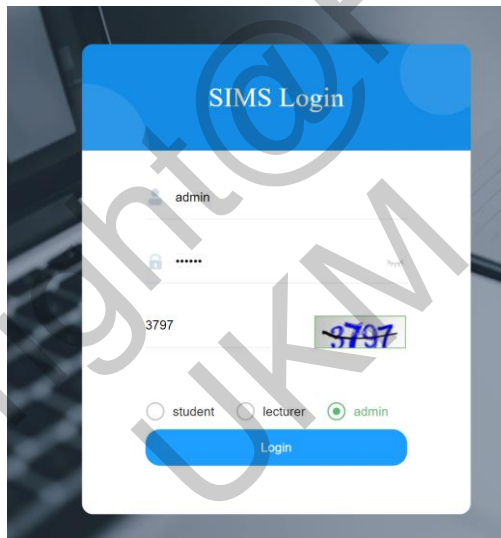


Figure 1 login page

Figure 2 shows the user's menu. The menu pops up when you click on the user's name in the upper left corner. The user can choose to log out or change their password. When clicking on Change Password, the Change Password screen will pop up and the user will be prompted to change the password successfully.

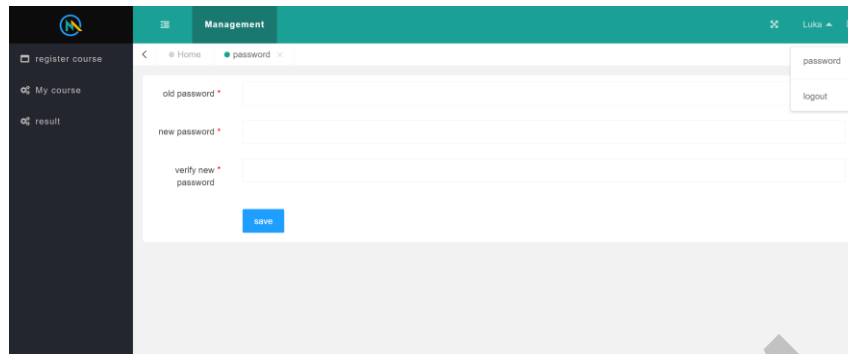


Figure 2 change password

Figure 3 shows the administrator page after a successful login. In this page the administrator can add colleges of the University, based on the majors below the colleges. Classes based on majors. Students and lectures can also be added. In the course management module, you can add courses, and assign lectures to teach them. In the home page, the whole page is in black color, and you can see the statistical charts based on college, major, students and lectures.

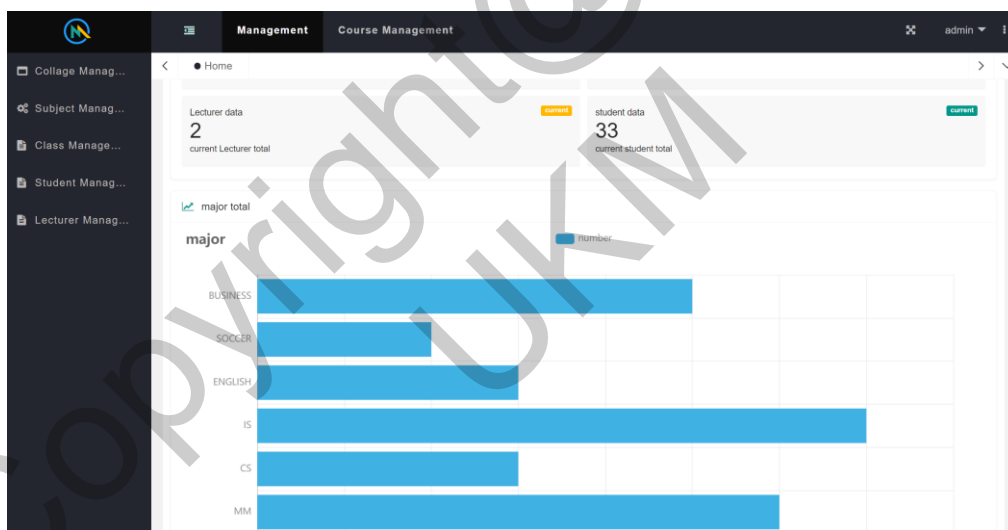


Figure 3 admin menu

Figure 4 shows one of the administrator's management pages. Students can be added to this page. It is also possible to modify the information of existing students. Students who have graduated or withdrawn can be deleted.

Input student name

Matric	Name	Gender	Phone	Id/Passport	Intake year	Collage	Major	Class	
a176845	Luka	male	+6018-21...	EG74455...	2019	FTSM	IT	IT1	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
a177155	cai	male	123456	EG35111	2019	FTSM	IT	IT1	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
a987654	liu	male	123456	123456	2023	FEP	BUSINESS	count	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
a456123	yang	female	123456	123456	2018	FEP	BUSINESS	count	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
a123456	wang	female	123456	123456	2020	CITRA	ENGLISH	english	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Figure 4 student management

Figure 5 shows the pop-up window for adding a new student. To add a student, you need to enter the correct student information and select the correct college, program, and class.

Input student name

**New student**

Matric Number:

Student Name:

Select Class:

Password:

Gender:  Male  Female

Figure 5 new student

Figure 6 The administrator can assign courses to lecturers, and you can see basic information about each lecturer on this page.

Input course name

**assign lecturer**

Input student name

Number	Name	Gender	Phone	Describe		
<input checked="" type="radio"/>	d123	Dr.Tom	female	123456	good at it	<input type="button" value="Assign Lecturer"/>
<input type="radio"/>	a456	Dr.good	male	123456	good at pr...	<input type="button" value="Assign Lecturer"/>

Figure 6 assign lecturer



Figure 7, the lecture's entire page is in purple, and the lecture can view the courses assigned to them by the administrator. It is also possible to grade students' grades.

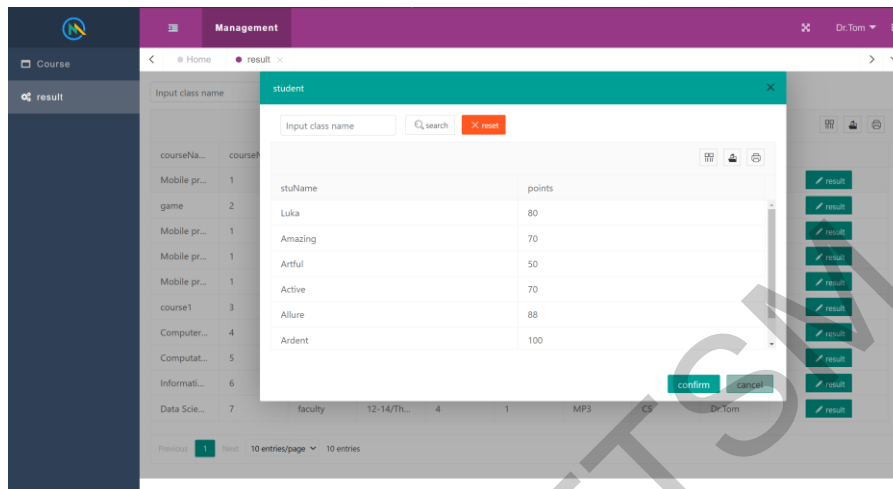


Figure 7 lecturer page

Figure 8 shows all the classes that a lecture is required to teach this semester. The lecture can see the time and place as well as the classes from above.

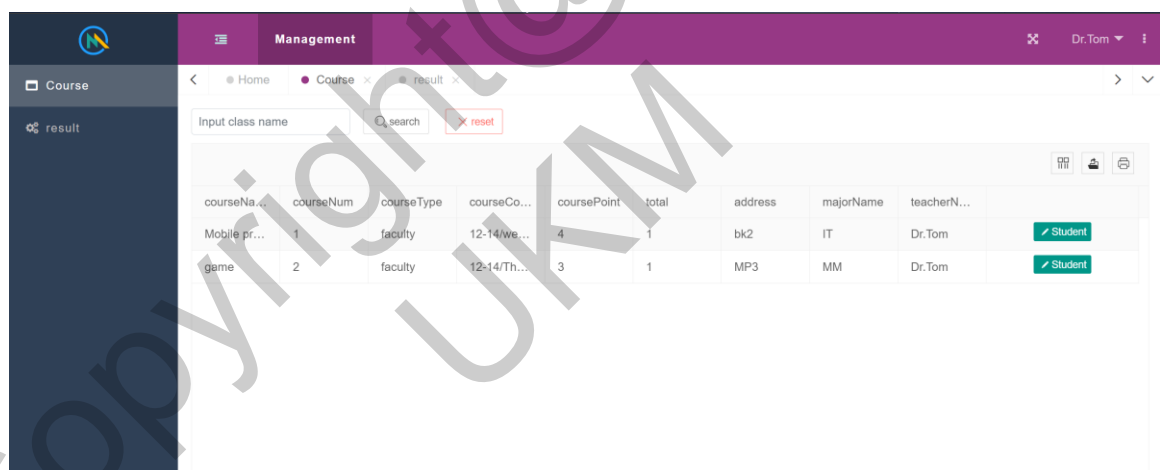


Figure 8 lecturer course

Figure 9 When the lecture clicks on the student button next to the course. Basic information about all the students who have chosen this lesson will be displayed.

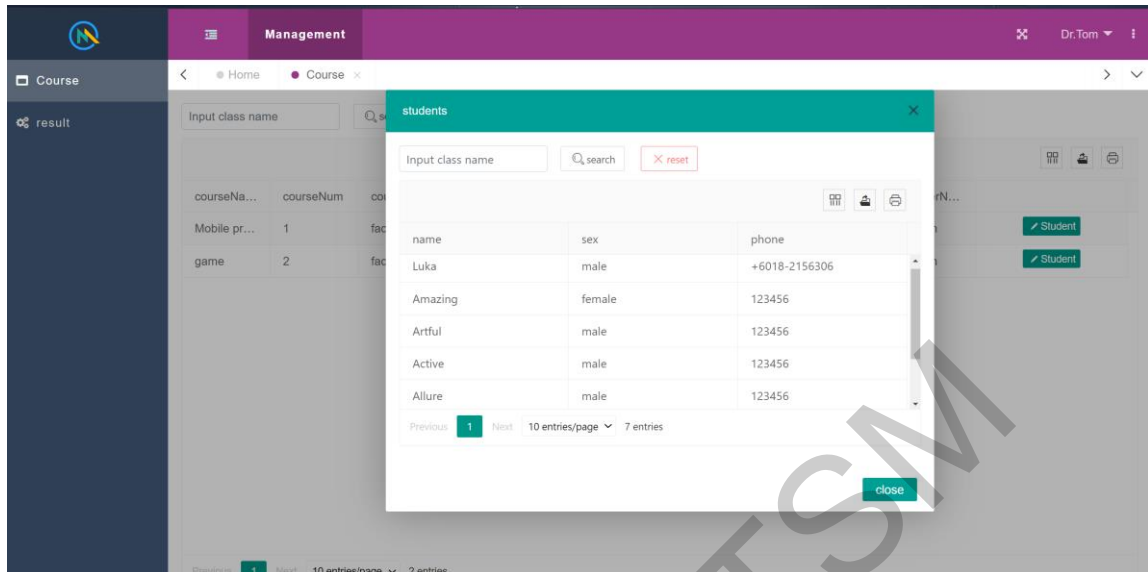


Figure 9 student list

Figure 10 student page is primarily green in color, and in the home page students can see graphs of their grade-based analysis. Students can make course selections, view the courses they have chosen for the semester, and their grades at the end of the semester.

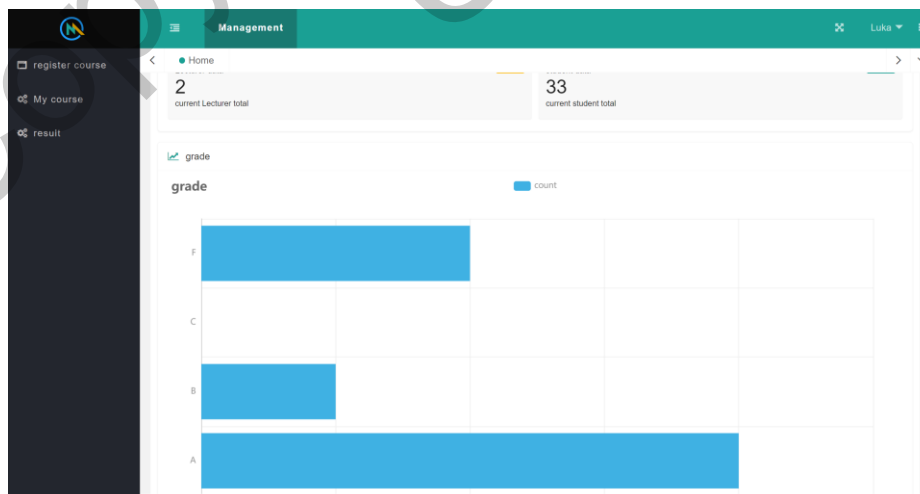


Figure 10 student page

Figure 11 shows the process of registering a student for a course. Students will see all the courses for the semester when they enter the Register for Courses page, and they can select the courses they need to register for.

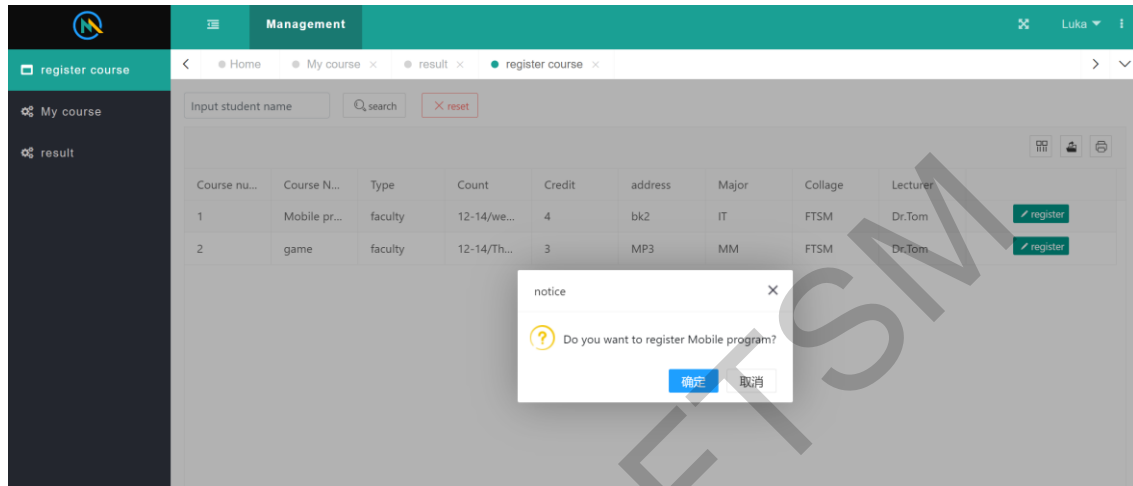


Figure 11 register course

Figure 12 Student's Schedule of Classes, showing basic information about the student's selected classes, class times, and locations.

The screenshot shows a web interface for viewing selected courses. At the top, there's a navigation bar with 'Management' and a user profile 'Luka'. Below that, a search bar is labeled 'Input collage Name'. A table lists two courses:

courseName	courseNum	courseType	courseCount	coursePoint	address	majorName	teacherName
Mobile program	1	faculty	12-14/wednesd...	4	bk2	IT	Dr.Tom
game	2	faculty	12-14/Thursday	3	MP3	MM	Dr.Tom

Figure 12 course table

Figure 13 shows the page where students can view their grades. When the lecture assigns a grade, students can check it through the page.

courseName	mark
Mobile program	80
game	50

Figure 13 result page

### CONCLUSION

Overall, this project is challenging for developers, first of all from the aspect of architecture, designing a good architecture can help developers to develop the whole project more easily. Secondly, there are software and language issues, because the language and framework of JAVAWEB are not covered in the University program. So all the languages and technologies need to be relearned. The server problems occurred by using TOMCAT server. The most difficult part was the use of LAYUI and how to coordinate the parameters. the features of the JavaWeb Student Management System were thoroughly tested. Overall, most of the features passed the test and met the expected design requirements. The system performs well in the three roles of administrator, instructor and student, and the interface is well designed and easy to use. However, some problems were found during the testing process, mainly focusing on the need to further optimize and fix the function of assigning lecturers.

### APPRECIATION

I would like to express my sincerest gratitude to all those who have helped and supported me in completing this student management system project.

First and foremost, I would like to express my special thanks to my supervisor Dr. Dalbir Singh A/L Valbir Singh, whose expertise and experience has been an

invaluable asset to me. Your careful guidance and patience throughout the course of the project enabled me to overcome various difficulties and challenges and ensured the smooth progress of the project. I have benefited greatly from your trust and support, which have given me the opportunity to utilize my creativity and abilities in the project.

At the same time, I would like to thank all those who provided me with resources and assistance. Whether it was in data collection, technical support, program review, or other areas, your help was crucial to me. Thank you for your generous sharing and selflessness, which enabled me to successfully complete this project.

In addition, I would like to thank my classmates and team members. Your unity and joint efforts during the collaborative process of the project laid a solid foundation for its success. Your friendship and support made me feel the warmth of home and the sense of belonging to the team.

Finally, I would like to thank my family and friends. During the course of this project, you have given me endless support and understanding, encouraged me to persevere and cheered me on. Without your support, I could not have achieved what I have today.

#### REFERENCE

- Yan, S., & Yang, Y. (2020). Education Informatization 2.0 in China: Motivation, Framework, and Vision. *ECNU Review of Education*, 4(2), 410–428. <https://doi.org/10.1177/2096531120944929> [5 October 2022]
- Habib, M. N., Jamal, W., Khalil, U., & Khan, Z. (2020). Transforming universities in the interactive digital platform: case of city university of science and information technology. *Education and Information Technologies*, 26(1), 517–541. <https://doi.org/10.1007/s10639-020-10237-w> [7 October 2022].
- Hadgraft, R. G., & Kolmos, A. (2020). Emerging learning environments in engineering education. *Australasian Journal of Engineering Education*, 25(1), 3–16. <https://doi.org/10.1080/22054952.2020.1713522> [12 October 2022].
- Petrovskaya, A.,

- Pavlenko, D., Feofanov, K., & Klimov, V. (2020). Computerization of learning management process as a means of improving the quality of the educational process and student motivation. *Procedia Computer Science*, 169, 656–661. <https://doi.org/10.1016/j.procs.2020.02.194> [5 October 2022].
- ALAMERI, I., & RADCHENKO, G. (2017b). Development of Student Information Management System based on Cloud Computing Platform. *Journal of Applied Computer Science & Mathematics*, 11(2), 9–14. <https://doi.org/10.4316/jacsm.201702001> [11 October 2022].
- Tomasevic, N., Gvozdenovic, N., & Vranes, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, 143, 103676. <https://doi.org/10.1016/j.compedu.2019.103676> [5 October 2022].
- Cambridge Assessment International Education. (n.d.-d). Results Analysis. <https://www.cambridgeinternational.org/exam-administration/results/results-analysis/> [7 October 2022].
- Martin, M. (2022, November 19). What is Waterfall Model in SDLC? Advantages and Disadvantages. *Guru99*. <https://www.guru99.com/what-is-sdlc-or-waterfall-model.html> [5 October 2022]

Wang Chaofan(A176845)  
Prof. Madya Dr. Dalbir Singh A/L Valbir Singh  
Fakulti Teknologi & Sains Maklumat,  
Universiti Kebangsaan Malaysia