Multidisciplinary Applied Research and Innovation Vol. 4 No. 1 (2023) 139-145

MARI

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/mari
e-ISSN:2773-4773

Course Distribution System for Bachelors in Information Technology

Nurhidayatul Husna Hajakamal* and Aniza Mohamed Din

School of Computing, Universiti Utara Malaysia,Sintok, Kedah 06010, Malaysia

*Corresponding Author Designation

DOI: https://doi.org/10.30880/mari.2023.04.01.026
Received 15 October 2022; Accepted 30 November 2022; Available online 15 January 2023

Abstract : The Course Distribution System for Bachelors in Information Technology (CDSBIT) is a web-based system designed to help the head of the department obtain a list of open slots for each course in an efficient and organized manner. As a timetable must be generated for every semester, the university's academic affairs office needs to be informed of the number of classes to be opened for each course. Usually, the number of classes for each subject can be determined by dividing the number of students enrolled in the course with the number of students allocated for the course. Due to the large number of courses available each semester, this is not an easy task. The current course distribution process relies heavily on Microsoft Excel. The head of department calculates the number of classes by entering the number of students and choosing the correct formula to perform the calculation. Switching between computers will also make it difficult to keep track of the latest files. Hence, a course distribution system is proposed to facilitate the task of distributing the classes quickly and efficiently. This system aims to assist the head of the department in providing reports and storing data in the database for future review. By using this system, the format will be standardized no matter where it is used. A waterfall model was used as the methodology for this project, which took four months to complete. The system was tested and evaluated based on user acceptance and ease of use by the head of department, deputy dean, and program coordinator in the School of Computing. It can be concluded that CDSBIT can meet the department head's requirements, implying that it can generate reports and obtain a list of slots efficiently.

Keywords: Course distribution, Slots, Class

1. Introduction

The problem of slots or classes availability in School of Computing courses between students is known as course distribution. Each course has a limit to the number of students who can register in every semester for the lecturer to devote proper attention to every individual student [1]. As requests for some courses exceed the limit, it is logical to wonder which students must be permitted to register for each course [2] and a suitable number of slots or classes to be opened for each course. First-come, first-serve registration is a common practice among educational institutions [3]. However, this could result in unfair outcomes: a student who just so happens to be available when registration opens will be

*Corresponding author: nurhidayatul_husn@soc.uum.edu.my

able to register for all the most popular courses, whereas a student who was a bit late may discover that all the popular courses have already been filled and will only be able to register for the less popular courses. As some of the courses are prerequisites for other courses, some students may get desperate to enroll in those courses. Thus, a proper planning of course distribution is required. Most schools employ manual allocation techniques as requirements for each course are different between courses.

University course arrangement in various Indonesian institutions is mainly done manually through the administrative department of each faculty [4]. As a result, this procedure may require some time since the officer must synchronize all calendars from across all units in that faculty. Furthermore, classroom and time slot constraints might cause subject and lecturer overlap, as well as mismatch in between number of rooms and the number of students permitted to attend class in the classroom. Thus, every semester, a timetabling procedure must be completed, which is an intensive and time-consuming task [5]. The distribution of time - slots and rooms must consider the hard and soft limits provided in one semester in order to avoid conflicts. Hard restrictions should not be breached under any situations inside the university course scheduling dilemma; soft constraints should also be avoided as much as possible [6].

Every semester, the university's academic affairs office generates a timetable of each student's courses. To generate a timetable, the academic affairs office requires a list of the number of slots or classes that must be opened for each course in all programs offered by each school in the university. This task is usually carried out manually by the head of department. As there are numerous classes available for all programs each semester, this will increase their burden. As a result, a Course Distribution System for Bachelor Science in Information Technology (CDSBIT) is proposed to be used to automate the procedure and hence assist the Head of Department in obtaining a list of slots or classes that are open for each course. This system was created utilizing the SDLC process.

2. Material and Methods

The waterfall model is proposed as the project methodology because it can produce high-quality systems that meet or exceed customer expectations based on customer requirements. This methodology, as shown in **Figure 1**, consists of five phases, which are planning, analysis, design, implementation, and maintenance and support. For the planning, a project is given by the supervisor regarding the problem encountered by the Head of Department for each school. A synopsis of this project is given to learn a little about the project to be developed. The initial focus of this project is on the courses offered in the School of Computing, Universiti Utara Malaysia. A draft of the proposal is sent to the supervisor for review of the system that will be developed. At the next stage, the requirements for this subject need to be identified and analyzed.

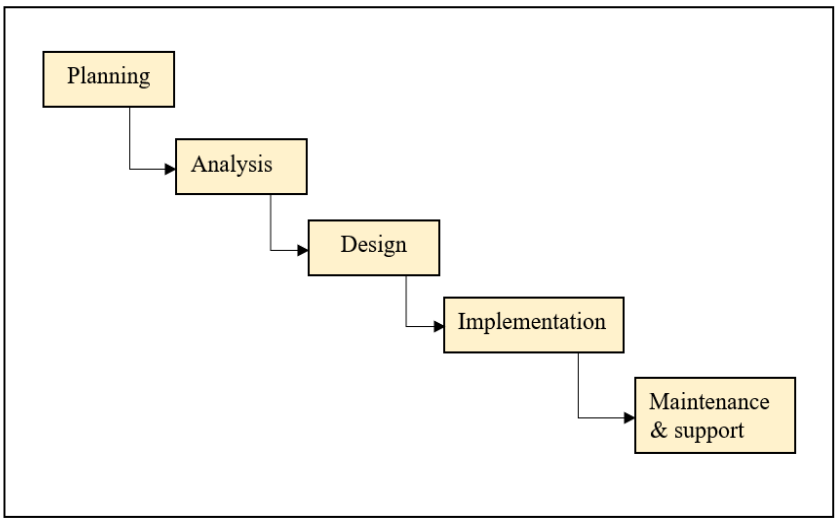

Figure 1: Waterfall Model

In the analysis phase, for user requirements, an interview with the head of department for the School of Computing is conducted to get some ideas of the current problem. The input from the interview is then used as the requirements for this system and becomes the focus for the whole development process. The outcome of this phase is a list of user requirements for the system. The information collected and analyzed are then used as a logical model to design the architecture to support the new information system. This phase includes designing the storyboard and interface of the movement system. The reason for creating the design is to prevent the lack of system performance. To calculate the number of classes/slots for each course, the number of students per semester needs to be identified and a suitable number of students for each course needs to be determined. Different courses sometimes have a specific number of students per slot or class based on course types and the classrooms available. Then, for each course, divide the total number of students per semester by the number of students per slot or class. In the implementation phase, the user requirements are transformed into a user model and constructed using the PHP programming language. Visual Studio Code served as the main integrated development environment (IDE) tool. Additionally, crucial services like database storage are optimized using the Laragon development platform. The outcome of this phase is a fully functional system that is ready to be tested. The system is tested based on updates and responds to bugs, new features and adjusted based on the business environment. The developed system was then demonstrated to and tested based on user acceptance and ease of use by the head of department, deputy dean and program coordinator from the School of Computing. **Figures 2 to 4** show the main page of the system, the course distribution page and report generation.

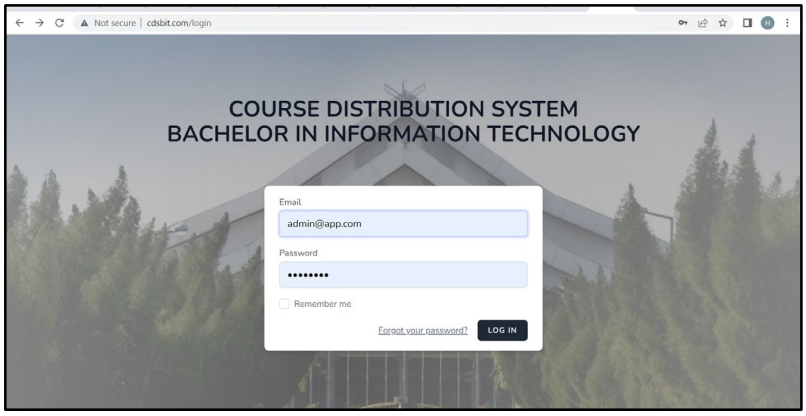

Figure 2: The main page of CDSBIT

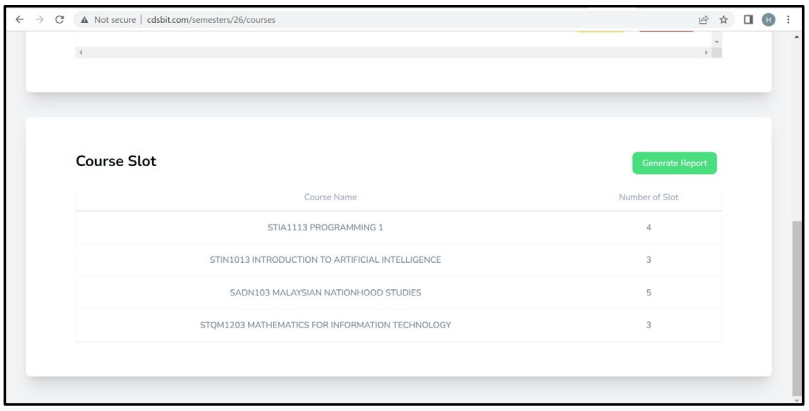

Figure 3: The page for course distribution

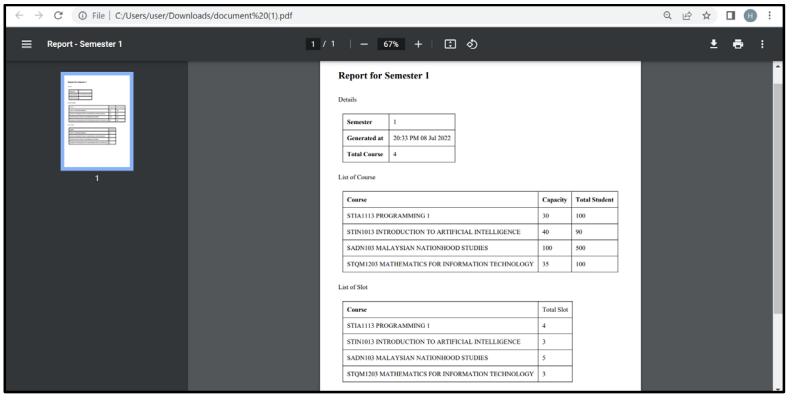

Figure 4: The page for report generation

3. Results and Discussion

A usability evaluation was done on three responders from the School of Computing, consisting of the head of department, deputy dean, and program coordinator. The CDSBIT was demonstrated, and respondents were allowed to test the system themselves. They were then required to fill in a questionnaire that focused on user acceptance and ease of use. The questionnaire requested their opinions on CDSBIT on a 5-point Likert scale, with one representing strongly disagree and five representing strongly agree. For the evaluation, the respondents followed the steps outlined below: (1) approved a written consent; (2) interacted with the CDSBIT in accordance with the experiment method; and (3) answered a post-task questionnaire.

3.1 The Usability of CDSBIT

The replies of respondents on a post-task questionnaire were analyzed. This section assessed respondents' perceptions on CDSBIT usefulness and simplicity of use. It also assessed user satisfaction with CDSBIT. The frequency and average of the replies were represented in **Tables 1 to 3**. Four or five of the reply measures were scored by respondents for the three characteristics of usefulness. Neither of the respondents gave a score of one or two. Only a few were graded as neutral.

The evaluation findings indicated that CDSBIT is beneficial and simple to use. Furthermore, respondents stated that they were pleased with the system since it saved users time when they utilized it to obtain a list of available slots. In terms of user interface, it was stated that CDSBIT was simple to use and that they might readily recall how to use the system without written instructions. Furthermore, respondents were pleased with the system's aesthetics and stated that they utilized it effectively every time. Respondents also strongly believe that this system is fantastic and enjoyable to use.

Table 1: The respondents' responses on the usefulness of CDSBIT

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I can Login to the system with email and password.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
All the button can function well.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
I can create semester and courses without any problem.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
I can update semester and courses without any problem.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
I can edit semester and courses without any problem.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
I can delete semester and courses without any problem.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
It saves my time when I use this system to get list of slots.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
This system meets my needs.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
This system is useful in overall.	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)

Table 2: The respondents' responses on the ease of use of CDSBIT

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
This system is easy to use	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
The user interface of this system is friendly	0 (0.00)	0 (0.00)	0 (0.00)	2 (6.67)	1 (3.33)
This system is easy to learn how to use it	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)
I can use this system without written instruction	0 (0.00)	0 (0.00)	1 (3.33)	1 (3.33)	1 (3.33)
I can easily remember how to use this system	0 (0.00)	0 (0.00)	0 (0.00)	2 (6.67)	1 (3.33)
I can use this system successfully every time	0 (0.00)	0 (0.00)	0 (0.00)	2 (6.67)	1 (3.33)

Table 3: The respondents' responses on their satisfaction of CDSBIT

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I am satisfied with this system	0 (0.00)	0 (0.00)	0 (0.00)	2 (6.67)	1 (3.33)
This system works the way I want it to work	0 (0.00)	0 (0.00)	0 (0.00)	2 (6.67)	1 (3.33)
This system is wonderful and pleasant to use	0 (0.00)	0 (0.00)	0 (0.00)	1 (3.33)	2 (6.67)

4. Conclusion

A course distribution system for information technology bachelor's degrees has been created, and it can retrieve a list of slots for each course. Further improvement can be made on the user interfaces by taking into consideration the lab or tutorial requirements for each course. The system can also be further improved with a feature to assign a lecturer to a certain slot in order to make sure that the slots can have the appropriate lecturers.

Acknowledgement

Many thanks to University Utara Malaysia (UUM), all staff at UUM's School of Computing, and Dr Nor Hazlyna Binti Harun for their advice and helpful suggestions.

References

[1] T. A. Nugraha et al., "University Course Timetabling with Genetic Algorithm," UMY (JET-UMY)., Vol. 1, pp. 100–105, June. 2017.

[2] M. A. Al-Jarrah et al., "A Course Timetable System for Academic Departments Using Genetic Algorithm," Jordanian Journal of Computers and Information Technology., Vol. 3, pp. 120-134, April. 2017.

[3] A. Jamili et al., "A Comprehensive and Multi-Objective Mathematical Model for University Course Timetabling Problem," Paris, France, July. 2018.

[4] S. S. A. Alves et al., "A Novel Educational Timetabling Solution through Recursive Genetic Algorithms, Computational Intelligence (LA-CCI)," Latin America, Vol. 2, pp. 13-16, Oct. 2015.

[5] R. Chen and H. Shih, "Solving University Course Timetabling Problems Using Constriction Particle Swarm Optimization with Local Search," Taiwan, pp.227-244, April. 2013.

[6] M. C. Torres et al., "Course Allocation Problem Using Integer Programming Model," University of the Philippines Los Baños, Laguna 4031 Philippines., pp. 679-689, August 2021.

[7] J. S. Tan et al., "A survey of the state-of-the-art of optimisation methodologies in school timetabling problems," 165:113943, 2021.

[8] A. Gülcü and C. Akkan. "Robust university course timetabling problem subject to single and multiple disruptions," European Journal of Operational Research, 283(2):630–646, 2020.

[9] D. S. Vianna et al., "Hybrid VNS-TS heuristics for University Course Timetabling Problem," Brazilian Journal of Operations & Production Management, 17(1):1–20, 2020.