**AUTOMATED TIMETABLE SCHEDULING SYSTEM USING BACKTRACKING ALGORITHM**

**Hernando Jr. J. Costelo**

**Michelle Emmanuel S. Gaetos**

**Johnuel M. Javier**

**Ren Russel E. Lavilla**

**Lester D. Quijano**

An undergraduate thesis presented to the faculty of the Department of Computer Studies Cavite State University-Cavite College of Arts and Trades Campus, Rosario, Cavite in partial fulfilment of the requirements for the degree of Bachelor of Science in Computer Science with Contribution No. \_\_\_\_\_\_\_\_\_\_\_\_. Prepared under the supervision of Mr. Karlo Jose E. Nabablit.

**INTRODUCTION**

Despite ongoing advancements in technology that have led to the computerization and automation of various aspects of university organization work, the process of creating course timetables still remains a manual and challenging task. The preparation of course timetables had a significant time constraint for academic colleges. It involves addressing multiple constraints and issues, such as classroom availability, teachers' subject preferences, the number of available teachers and classroom capacities, potential conflicts between rooms and courses, and scheduling conflicts between courses and instructors. Generally, course timetabling in many universities is prepared manually and the scheduling committee should consider all available facilities and resources, such as courses, instructors, rooms and laboratories. Moreover, the instructors, course and sections time were important constraints to handle. Therefore, based on all the mentioned constraints, course timetabling is a very exhaustive and time-consuming task. (Al-Jarrah et al, 2017)

Cavite State University CCAT Campus had offered 9 courses, including BS Electrical Engineering, BS Computer Engineering, BS Computer Science, BS Information Technology, BS Hotel Management, BS Business Managements, BS Technical-Vocational Teacher Education, BS Education with 2 different Majors, and BS Industrial Technology with 9 different Majors. As the campus continued to grow, the number of students increased, making the manual creation of course timetables a complex task to accomplish. In CVSU CCAT, the creation of schedules was manually done by the faculty head or scheduling committee of each department. The previous scheduling process consumed more time as the plotting, revising, and arranging of schedules were done separately for each department. Each course had minor and major subjects, and the scheduling committee of each department had to wait for the major subjects to be plotted in the timetable before plotting the minor subjects.

Furthermore, the proponents had come up with an idea and proposed the development of a web-based Automated Timetable Scheduling System to reduce the difficult work of the faculty head or scheduling committee in creating schedules and to save time in the scheduling process. The web application aimed to help CVSU CCAT avoid mistakes when creating schedules and generate an optimal course timetable within a short period of time. Additionally, this web application would assist the university scheduling committee in managing schedules with ease through an intuitive user interface.

**Objective of the Study**

The main objective of the study is to create a dynamic web-based scheduling system capable of generating conflict-free schedules. The system will specifically cater to the scheduling needs of Cavite State University-CCAT Campus especially in Department of Computer Science faculty, effectively eliminating the need for manual scheduling.

Specifically, the study aimed to:

1. Document and analyze the existing process of scheduling classes
2. design and develop a web application that automate the generation of class schedules.
3. test the reliability and functionality of the generated output; and
4. Evaluate the system using modified ISO-IEC 25101.
5. Create an implementation plan for the deployment of the software.

**Significance of the Study**

Through the implementation of a backtracking algorithm, the proposed system made the process of generating a class schedule more efficient. This implementation resulted in a notable reduction in time requirements and a decrease in the likelihood of human errors during the scheduling process. As a result, Cavite State University CCAT-Campus was able to obtain an optimal class schedule. Moreover, the system proved beneficial for the scheduling committee as it simplified tasks related to adding, deleting, and updating vital information for the faculty loading process.

**Scope and Limitation of the Study**

The study was focused on the development of the Automated Scheduling System for Cavite State University - CCAT Campus. The system was designed with features that allowed users to automate the generation of class schedules using a backtracking algorithm. Furthermore, during the study, significant importance was given to the utilization of web-based system development to ensure the ease of accessing data, an intuitive user interface, and reliable storage.

ATS only had a separate view page for generated schedules for teachers, course year, section, and rooms. The output could be generated in PDF format for printing.

ATS was not designed for multiple users, and the scheduling committee is the primary target user of the system.

ATS can have multiple accounts with the same access, such as adding, updating, and deleting data. The target user can also assign subjects to a teacher and then generate the automated schedule and print using PDF.

ATS is specifically designed to generate an automated schedule for the Department of Computer Science only. The generated schedules cannot be updated since conflicts may arise. However, if you want to add, update, or delete data in the faculty loading data, you need to regenerate the schedule to have an optimal schedule without conflicts.

ATS only generates schedules for full-time teachers and regular students. It does not consider irregular students and course schedules.

**Time and Place of the Study**

The study was conducted from May 2022 to May 2023 at the Cavite State University-CCAT Campus.

**Definition of terms**

***Allele*** is the value indicated in the gene (e.g, teacher name, room id, subjects etc.)

***Chromosomes*** is a term used in genetic algorithm and it is a set of parameters that is used to define the solution.

***Course Timetabling*** is the assigning of class schedules to a specific course including teachers, rooms, timeslot, and subjects.

***Crossover*** is a genetic operator; this operator is use in the reproduction phase in genetic algorithm. This is used to mate two set of solution(timetables) and swapping its genes to create new offspring (timetables).

***Fitness Function*** is used to determine if the given timetables are the fittest solution to the problem considering the constraints. This function will give fitness score to each solution(timetables), the lower the score the higher the probability to be chosen for reproduction.

***Genes*** is the variables (e.g., teachers, rooms, subjects) that is joined together to form chromosomes(timetables). It is represented using binary value string of 1s and 0s.

***Genetic Algorithm*** is used to solve optimization problem (Course Timetabling). This algorithm is used to automate the generation of course timetables using the genetic operators (Selection, Crossover, Mutation) and genetic algorithm phases (Initial population, Fitness function, Reproduction, Termination)

***Hard Constraints*** refers to the constraint that the developed system must satisfy (e.g., no subject clashes).

***Initialization of population*** is the initialization of set of solution(timetables).

***Mutation*** is used to apply random changes to the new offspring(solutions) to prevent premature convergence (suboptimal solution).

***Offspring*** is a new set of solution. It is produce using the crossover in genetic operator in which two parents(timetables) were selected and mate to reproduce new solution(offspring).

***Population*** is a collection of chromosomes(solutions).

***Reproduction*** is genetic algorithm phase, in which genetic operators (Crossover and Mutation) are used to reproduce new solution(timetables).

***Selection*** is a genetic operator; it is used to select fittest individual(solution) for reproduction.

***Soft Constraints*** is the desirable constraints that the developed system should satisfies but not essential.

**REVIEW OF RELATED LITERATURE**

In this chapter the developers review studies, articles, software, case studies related to the development of the Automated Scheduling System. It also shows the relevant system, its features and method prior to the development of the automated scheduling system.

**Importance of gathering data**

According to Simple learn (2022), gathering and analyzing data before the software development help us to make better decisions and help us evaluate possible outcome.

According to CQL (2022), gathering data helps us identify existing problems and identify the appropriate approaches on how to solve those problems.

**Importance of using web application**

Web based application offers a wide variety of advantages compared to desktop application. Instead of installing the application or software to each of every computer your company or school has, it could be easily accessed through the use of internet. Maintaining a web application is a simple process since the host server could easily add or update data without the need to upgrade in every computer. (Khamooshi, 2019)

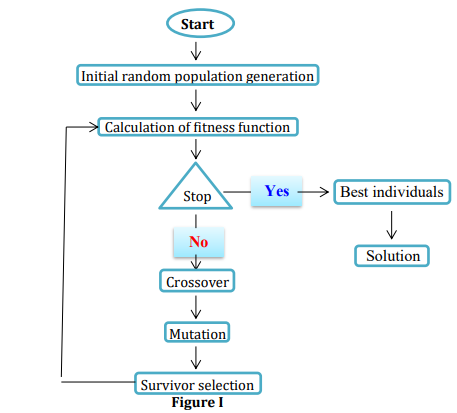
According to Souvik (2022), web application increase efficiency and reduce costs by automating the processes and task and reduce costs by allowing the user to add and update data through online, thus, it eliminates paper based manual data entry.

**Course Timetabling**

Course timetabling main objective is to assign a course to a specific time or set of time without conflicts to another course allotted time. It is also one of the university timetabling problems, it is resolved by meeting the specific constraints. This constraint is divided into two parts, the former is the hard constraints, and the latter is the soft constraints. (Burke and Petrovic, 2002). Hard constraints are the problems that must be satisfied/solved to generate a great timetable while soft constraints are the desirable problems but not essential to be solve. (Houhandi et. al, 2019). According to Ben Moreland (2015), these are the examples of hard and soft constraints. For hard constraints, Room capacity which means the number of students assigned to the room must best equal to the seating capacity of the room. For soft constraints, A preferred teaching week which means the teacher could choose when to teach the subject to specific time and day.

**Genetic Algorithm introduction and how it works.**

Genetic algorithm is a heuristic search method approach based on the Darwinian theory of evolution which means survival of the fittest and it is one of the methods used for timetabling optimization problems, this algorithm starts in creating a random timetable population and evaluate each of chromosomes using fitness function that are based on the soft and hard constraints. The next generation of population should produce better solution or timetables than before using crossover and mutation then it will repeatedly do the process until it satisfies the constraints. (Herath, 2017)



(*Source:* https://how genetic algorithm works?)

Figure 1: Genetic Algorithm Process

Genetic algorithm is evolutionary algorithm that considered five stages such as initial population or initialization, fitness function, selection, crossover and mutation. (Mallawaarachchi, 2017)

Genetic algorithm is a randomly based classical algorithm that initialize population to identify the number of individuals needed inside the population. Each of individual is called solution and each of this solution has chromosomes that are characterized by set of genes or set of parameters. Genes are represented using string 1s and 0s thus if the genes are joined together, it will form a chromosome that will be a part of population. Fitness function results provide a fitness value that represent the quality of the solution and if the fitness value is higher, it is considered as a good solution to the problem and it also has higher probability to be chosen for reproduction. (Gad, 2018). Selection is done during each successive generation, after the fitness function provide fitness value to each of the solution, this process will select the fittest individual or solution for mating to generate new solution. Reproduction is a process of generating a second set of population based on the selected fittest individual, this process uses genetic operators to reproduce new solution such as crossover and mutation thus genetic algorithm will repeat it process until a termination condition has been reached. (Kumar et. al, 2010)

**Genetic Algorithm implementation in Python**

The chromosomes in genetic algorithm have a set of genes and in most cases, genes are represented using strings or list and most of the operation of GA will be done using strings and a high-level programming language like python is needed since it is good in strings processing. Python is the ideal programming language for the implementation of genetic algorithm because it has a good readability, and it is flexible. (Wonja Lee et. al, 2005)

According to Ander Fernandez Jauregui (2022), Python is a recommended programming language for implementing genetic algorithm because it has different library that could be used for the development of genetic algorithm like PyGAD, which simplify the process of genetic algorithm and it also support the different types of crossovers, mutations and selections.

**Python Django web framework**

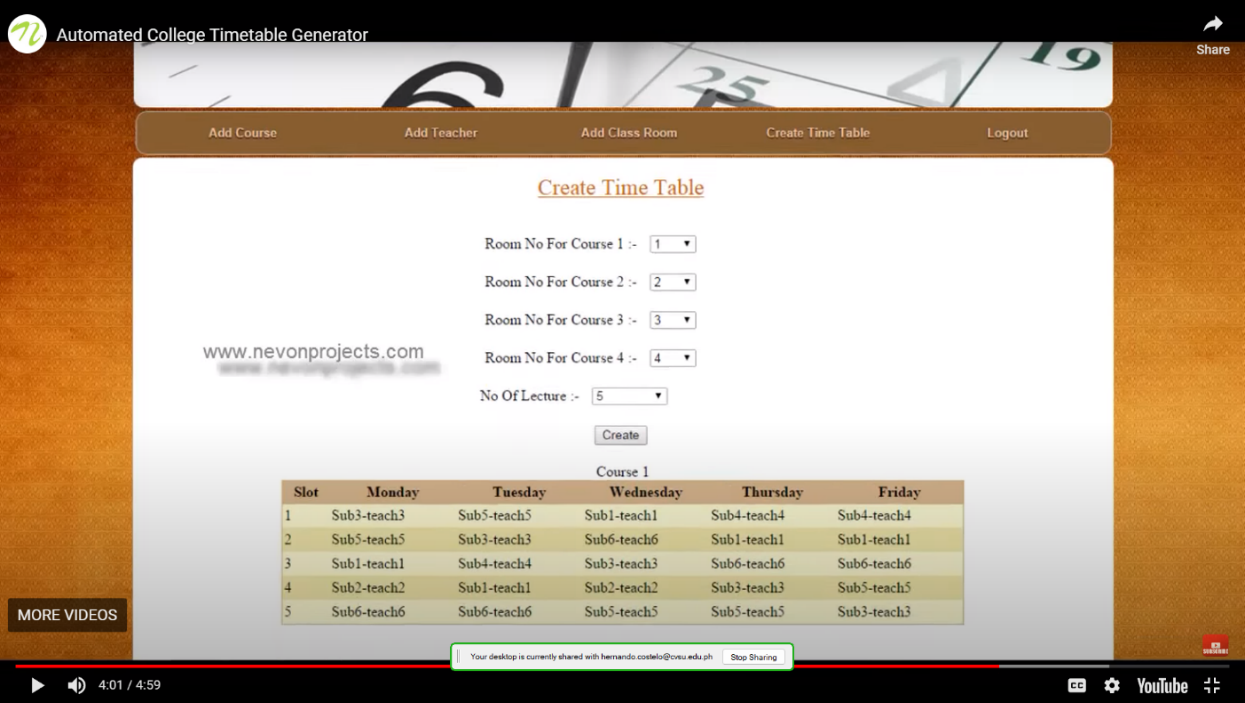
Django is the ideal web framework for python because it supports or encourages rapid development. This framework is designed to provide a rapid development to help developers to develop application as quickly as possible. (“Django Software Foundation, 2022)

**Related System:**

**Automated College Timetable Generator**

This web-based automated college timetable generator is used to automate the college timetable using genetic algorithm. The developed system is a dynamic system that allows user to update data such as add course, teachers, rooms and automate the generation of timetable

The related web application will be used by the developers as reference for the process flow of the system.



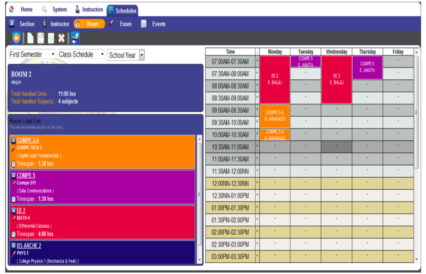
**(***Source: https: Automated College Timetable Generator)*

Figure 2. Create Timetable Tab

**Bohol State Island University Automated Scheduling System Capstone Project**

This study designed an automated class scheduling system for Bohol State Island University and test its functionality in terms of speed, accuracy, data handling, security, stability, and adaptability in making class schedules. This system is a dynamic system because the admin /user could update the data in the database like add teachers, add buildings, add course and adding new user.

The developers will use this related system as a reference in creating a dynamic scheduling system.



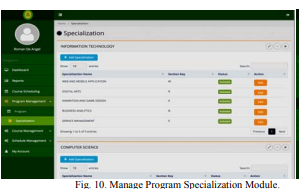
*(Source:* https: Automated Scheduling System Capstone Project)

Figure 3. Automate Schedule Tab

**Web based Course Scheduling System using Greedy Algorithm**

This study is a web-based system that focuses on course scheduling for College of Computer Studies under FEU institute of Technology, Manila Philippines. The developed system manages schedules on courses and assigning to faculty using greedy algorithm and the method of the research was research and development.

This related system will be used by the developers as a reference for creating admin dashboard.



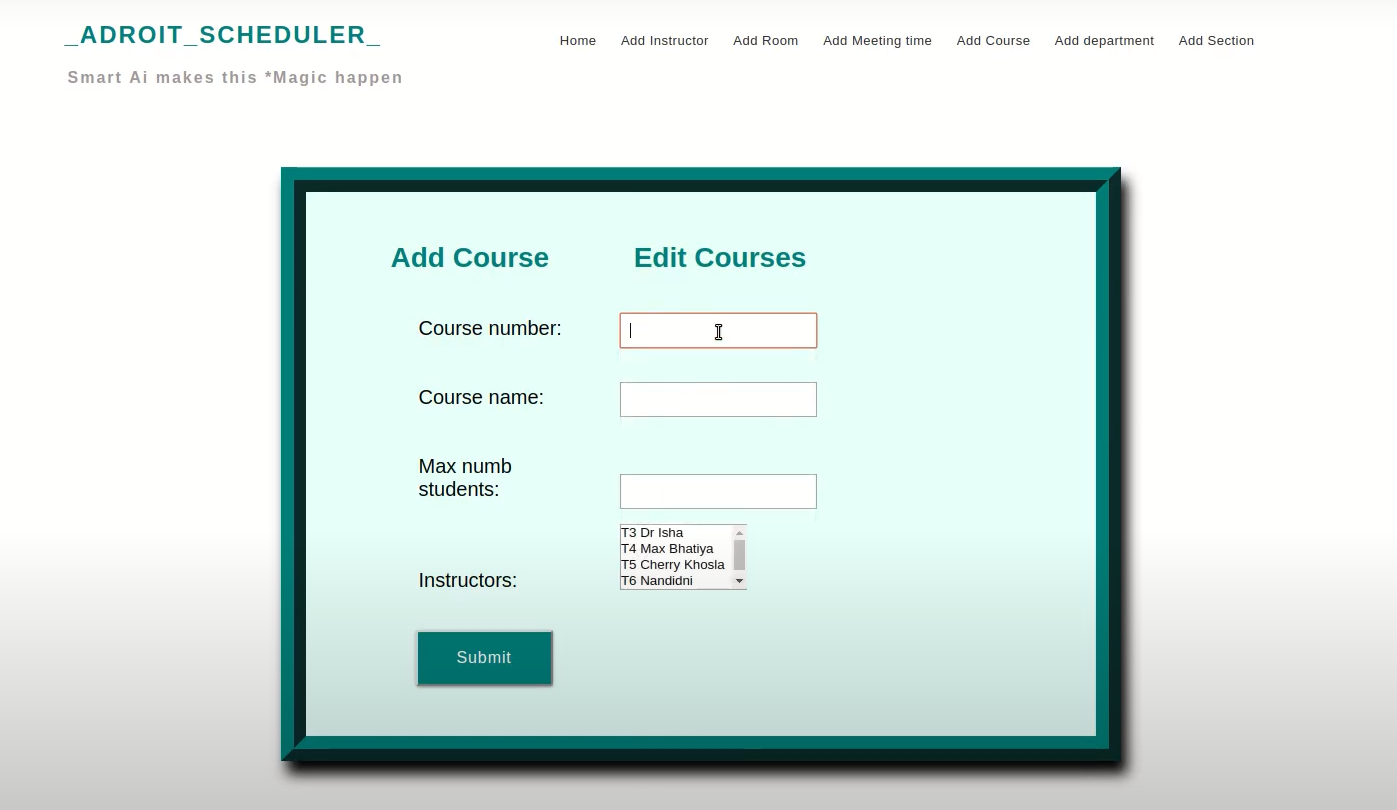
**(***Source*: https://ijssst.info/Vol-20/No-S2/paper14.pdf)

Figure 4. Manage Program Specialization Module

**Intelligent Timetable Generator using Genetic Algorithm**

This related web application uses genetic algorithm to generate university timetable. The development of this project is based on the hard and soft constraints.

The proponents will use this related system as a reference for the implementation of genetic algorithm in python.

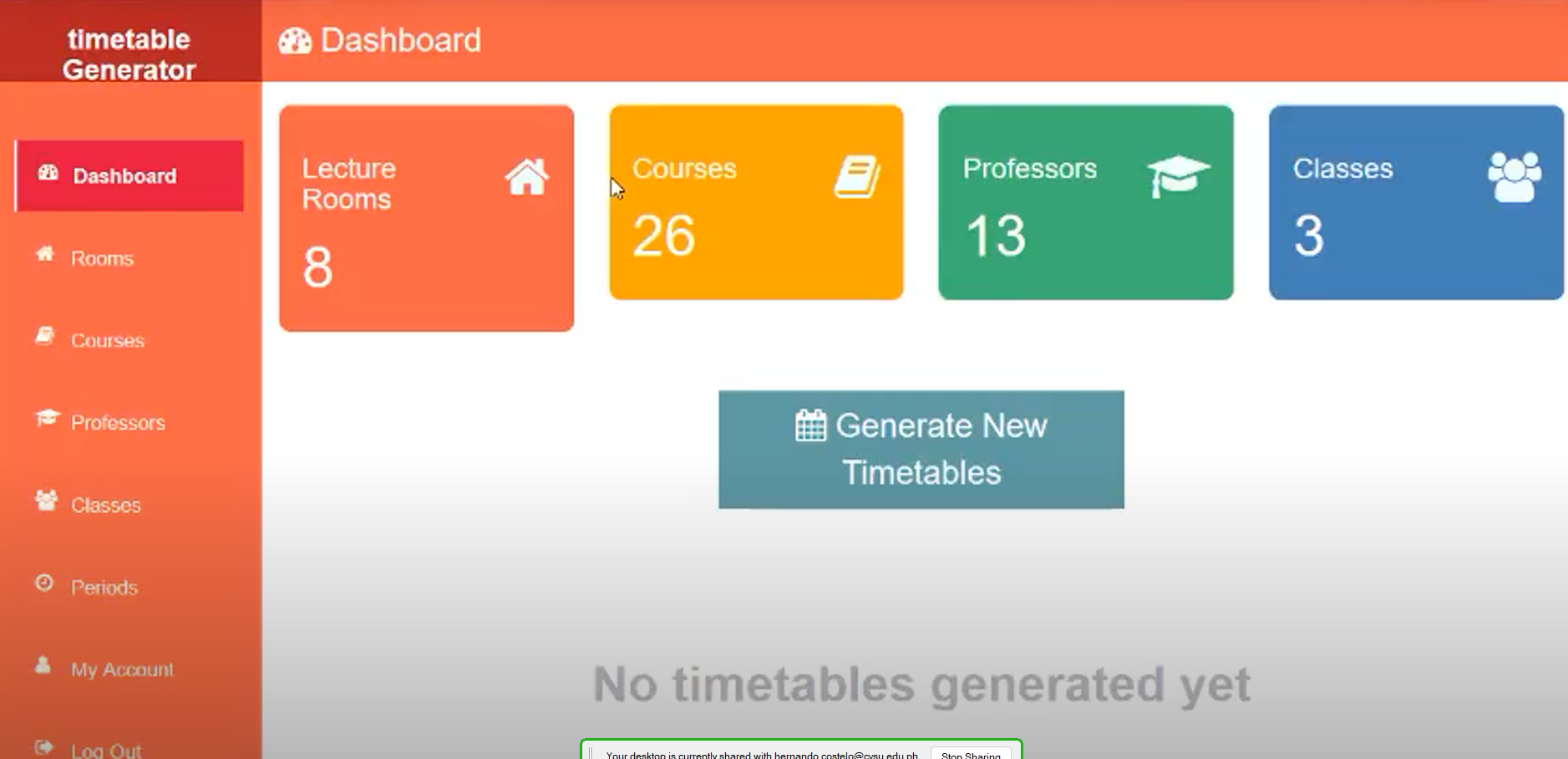
****

(*Source:* http://abhik.pythonanywhere.com)

Figure 5. Add Course Tab

**Timetable Generator using Genetic algorithm.**

Timetable Generator use genetic algorithm to automate the generation of university timetables, the developers of this web application used PHP as their programming language and used Laravel web framework.

****

(*Source:* https://Timetable Generator)

Figure 6. Admin Dashboard

**Synthesis**

To sum it all up, all the information gathered in review of related literature helps the proponents to have an idea on their study. It helped the proponents to identify the different stages or process of genetic algorithm and help them identify the appropriate method that developers should use in the development of the system. Lastly, before the development of the system the developers must identify the hard and soft constraints in regard to the current scheduling process of Cavite State University CCAT Campus to properly design the process flow of the system.

**Table of Comparison**

The proponents compare the features of proposed system to the other related system as shown below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Features** | **Automated College Timetable Generator** | **Bohol State Island University Automated Scheduling System** | **Web based Course Scheduling System using Greedy Algorithm** | **Intelligent Timetable Generator using Genetic Algorithm** | **Timetable Generator using Genetic algorithm.** | **Automated Timetable Scheduling System using Genetic Algorithm** |
| **1. Generation of Reports** | Checkmark with solid fill | Checkmark with solid fill | Checkmark with solid fill | Checkmark with solid fill | Checkmark with solid fill | Checkmark with solid fill |
| **2. Faculty loading** | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Checkmark with solid fill |
| **3. Admin Dashboard** | Close with solid fill | Close with solid fill | Checkmark with solid fill | Close with solid fill | Checkmark with solid fill | Checkmark with solid fill |
| **4. View Page for generated schedule** | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Checkmark with solid fill |
| **4. Generated schedule convert to a PDF file.** | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Close with solid fill | Checkmark with solid fill |

Figure 7. Table of Comparison

**METHODOLOGY**

In this section, the research method used to conduct the study was defined. The developers explained how they addressed the research objective by utilizing the gathered data and information. The Rapid Application Development Model was used by the developers to achieve faster output. This approach focused on prototyping to quickly deliver or introduce the product, rather than spending excessive time on planning.

**Conceptual Diagram**

**Current Scheduling Process Information**

* Process flow of the current scheduling
* Checklist of Programs offered at CVSU CCAT
* Teachers List

**Knowledge**

* Course Timetabling
* Backtracking Algorithm
* Web programming
* Database Management

**Software and Hardware**

* MySQL
* PHP
* HTML
* CSS
* JavaScript
* Boostrap5
* JQuery
* Figma
* TCPDF

**INPUT**

**Requirements Analysis Phase**

* The gathered information about the current scheduling process was analyzed.
* The related articles, books, studies, and systems were studied.
* The criteria for assigning timeslots and rooms were identified.

**Prototype Cycle Phase**

* The use interface design was created using Figma.
* The UI Design was demonstrated and refined

**Developing and Testing Phase**

* The UI was converted into a real web application.
* The User interface of the web application was coded.
* The Backtracking algorithm was implemented into the web application.
* The reliability and functionality of the generated output were tested.

**Deployment Phase**

* An implementation plan was created for the deployment of the system.

.

**PROCESS**

**Automated Timetable Scheduling System using Backtracking Algorithm**

**OUTPUT**

Figure 7. Input Process-Output model of Automated Timetable Scheduling System

**System Analysis and Design**

* **Requirements Analysis phase**

During the Requirements Analysis phase, the developers identified the different requirements needed and analyzed the related systems, studies, articles, and books for the development of the system.

* **Prototype Cycle Phase**

In the Prototype Cycle Phase, the proponents designed and created a prototype, demonstrated it for refining, and repeated the process until it was ready for the development phase.

* **Developing and Testing Phase**

In the Developing and Testing Phase, the developers converted the prototype into a real web application and tested its reliability and functionality. If the web application met all the criteria needed, it proceeded to the next phase. If not, it went back to the first phase until it met the criteria required for the deployment phase.

* **Deployment Phase**

The developers created an implementation plan specifically for Cavite State University CCAT-Campus to deploy the ATS system. The plan outlined the necessary steps for system installation, configuration, data migration, user training, and post-implementation support.

**Software/System/ Product Description or Process Design**

**Hardware Requirements**

In system development, the proponents used an HP LAPTOP with the following specifications: 8 GB DDR4-3200 MHz RAM, AMD Ryzen 5 5500U with Radeon Graphics operating at a speed of 2100 megahertz, and 512 GB PCIe storage, running on a 64-bit Windows 11 operating system.

**Software Requirements**

* **Front-end:**
  + **HTML**–stands for Hypertext Markup Language and it used to structure web page and it also use to organize the contents of the webpage.
  + **CSS** – stands for Cascading Style Sheet, it is one of the core technologies of web development. It is used for the design of web pages such as fonts, layouts, colors, shapes and it also used for responsive web pages to different device.
  + **Bootstrap 5** – it is a free and open-source front-end development CSS framework. It is used to provide responsive web design to different devices **s**uch mobile, laptop, tablet etc.
  + **JavaScript** - it is a scripting language that are used to create an interactive or dynamic website.
  + **JQuery –** It was used in system development to enhance the functionality and interactivity of the web application. It provided features and utilities for tasks such as DOM manipulation, event handling, AJAX requests, and animations, making it easier to implement dynamic and responsive elements in the system.
* **Back-end:**
  + **PHP -** is a widely-used server-side scripting language specifically designed for web development. It is embedded within HTML code and executed on the server to generate dynamic web pages. PHP offers a wide range of functionalities and is popular for building dynamic websites, web applications, and handling server-side tasks. It can interact with databases, handle form data, and generate dynamic content. PHP is known for its simplicity, flexibility, and strong community support, making it a popular choice for web development.
  + **MySQL** – is a relational database management system that used for storing or handling data using Structure Query Language.
  + **XAMPP** – It is a software package that provides a local development environment for building and testing web applications. It stands for "Cross-Platform, Apache, MySQL. XAMPP is a bundled software solution that includes several components necessary for web development, such as the Apache web server, MySQL database server, PHP programming language
  + **TCPDF** - TCPDF is a PHP library used for generating PDF documents. It stands for "TCPDF - PHP class for PDF". TCPDF provides various functions and methods to create PDF files dynamically, allowing developers to generate PDFs from scratch or based on templates.
* **Tools:**
  + **Google Drive -** Google Drive is a file storage, it is used to store research paper, images, diagrams, UI Design and Wireframes
  + **One Drive** – It is a cloud storage that allow you to edit and synchronize file.
  + **Microsoft Teams –** it is used to communicate with one another and schedule a meeting twice a week to report the progress of their task.
  + **Figma –** is an online collaboration tool used for creating digital products. It is used to design user interface of the system.
  + **Lucid Chart** – is a diagramming tool web application, it is used for creating diagrams (e.g., flowcharts, Site map).
  + **Visual Studio Code –** is a code editor and its supports different programming language like Python.
  + **Adobe Photoshop CS6** – is photo editing software, it is used for the design and image of the website.
  + **GitHub -** GitHub is a web-based platform that serves as a version control repository for developers. It provides a centralized location where developers can store, manage, and collaborate on their code projects. The platform uses Git, a distributed version control system, to track changes to files and facilitate collaboration among team members.

**Backtracking Algorithm Process**

The backtrack algorithm is a common approach used to solve problems that involve finding all possible solutions or combinations by systematically exploring different options. It is a recursive algorithm that incrementally builds a solution and backtracks whenever a dead end is reached, undoing the previous choices and exploring alternative paths. The backtrack algorithm in automated scheduling explores different task assignments, checks constraints, and backtracks if necessary to find valid schedules that satisfy all requirements.

**System Interface**

The following are the screenshots of the User Interface of the proposed system:

In login page, the admin needs to input correct username and password to login.

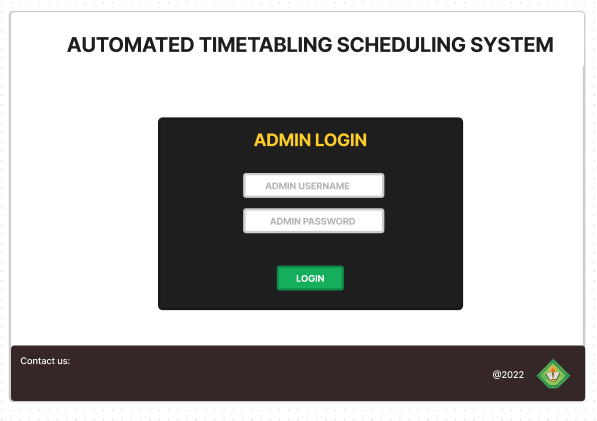


Figure 8. Log In Page

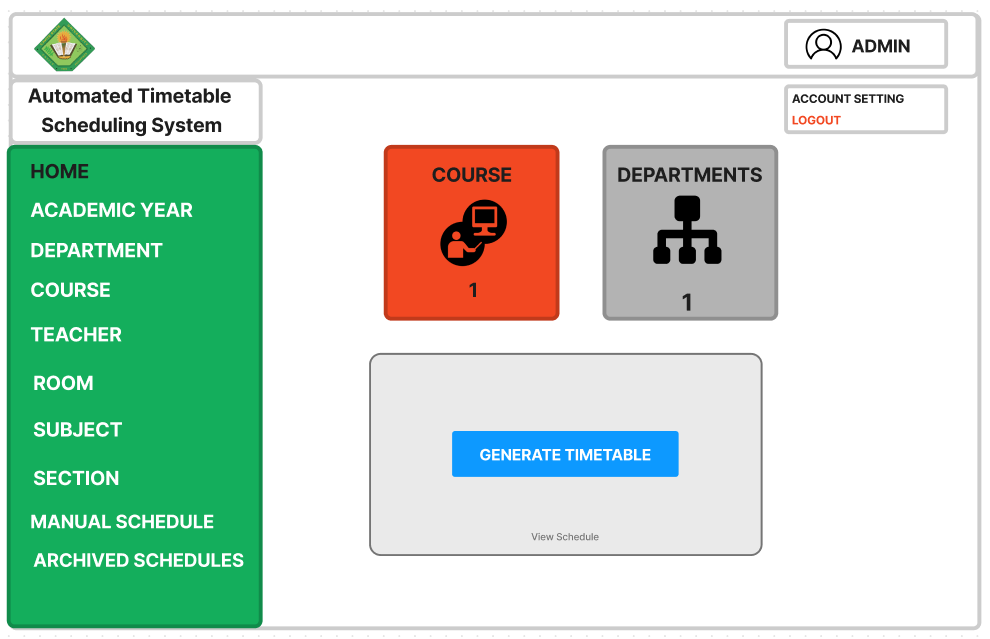
After a successful login, the admin will be redirected to admin dashboard. The admin is also allowed to configure the account setting. As you can see, on the admin dashboard generate timetable button and view timetable link is locked since the admin need to input all the data needed to generate and view the timetables. The Generate Timetable button will be available after the admin add the necessary data before the automation process.

Figure 9. Admin Dashboard

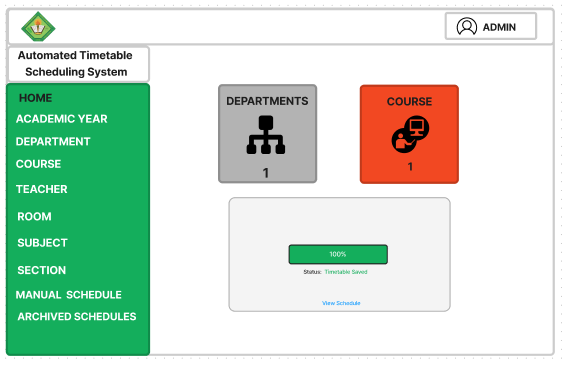
 In this figure, the admin added all the necessary data and clicked the generate timetable button.

Figure 10. Admin Dashboard

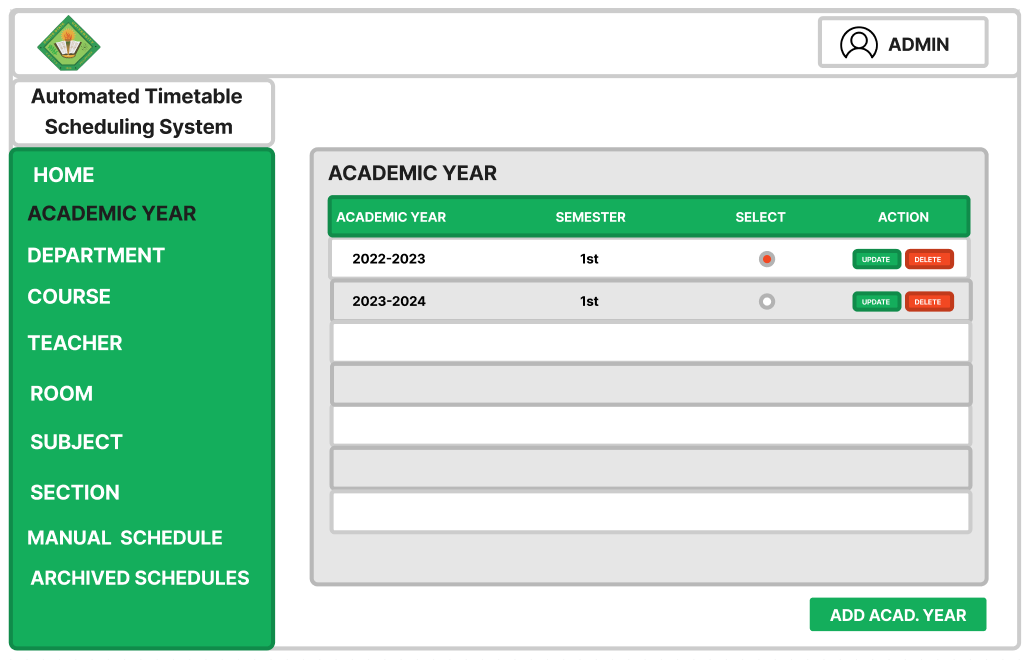
In figure 10, the admin needs to select what academic year will be used in the automation process of generating class schedules. The admin is also allowed to add, update, and delete academic year.

Figure 11. Academic year page list

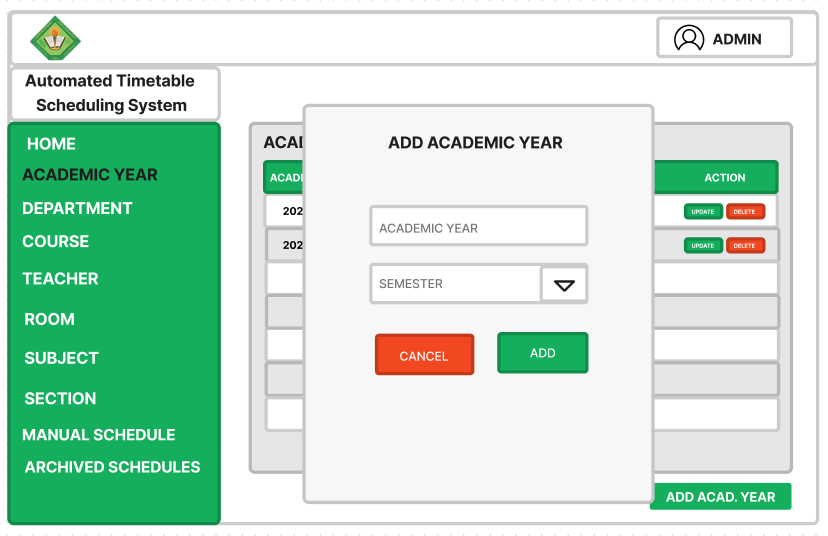
When admin clicked the add academic year button, a form will appear, and the necessary data will be needed to add new academic year as shown in figure 12.

Figure 12. Add academic year page

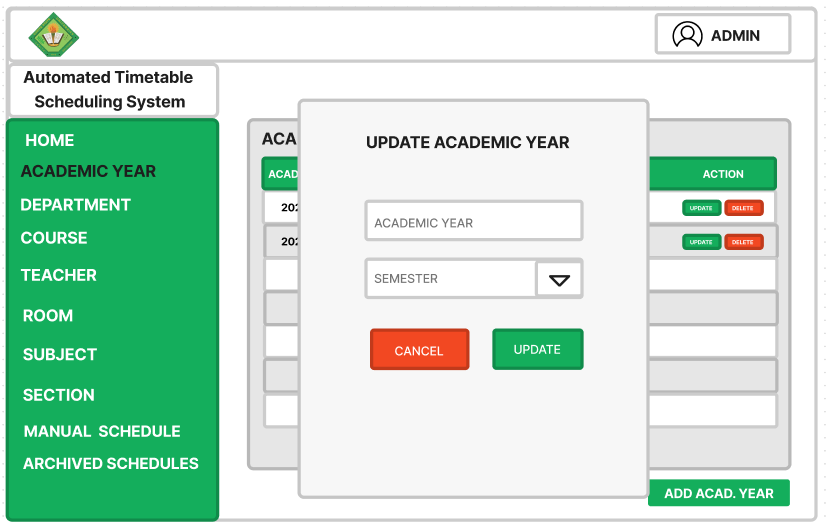
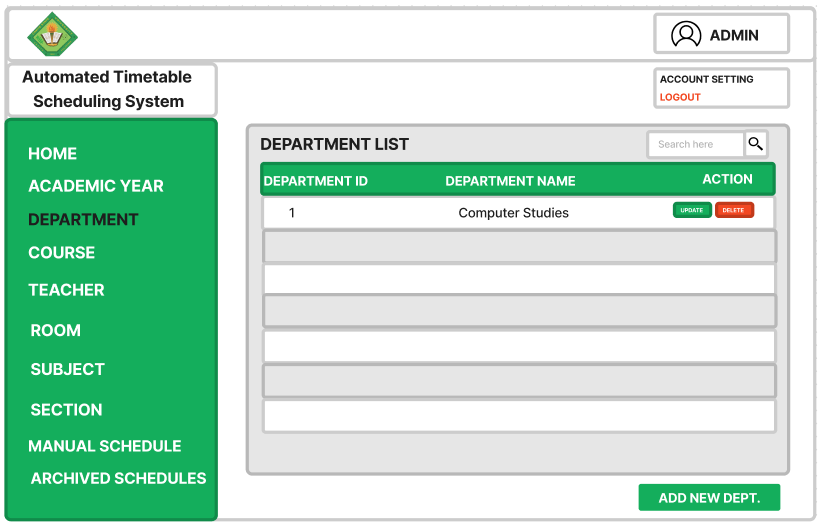
In this page, the admin can update specific academic year by updating the following information the academic year and the semester.

Figure 13. Updating academic year page

In the department list page, there is a list of available departments in the university, the admin can add, update and delete the department.

Figure 14. Department page list

Adding a new department will be needed to create a dynamic scheduling system, in this page the admin is allowed to add new department.

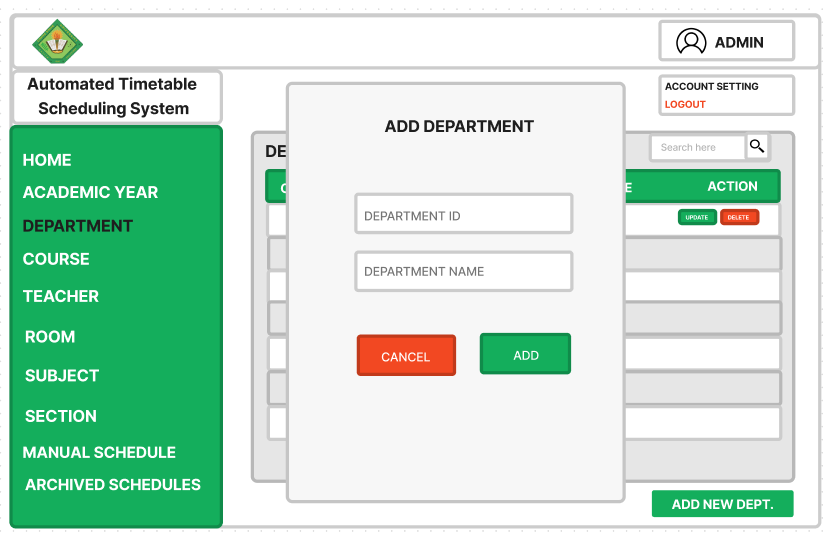


Figure 15. Add Department page

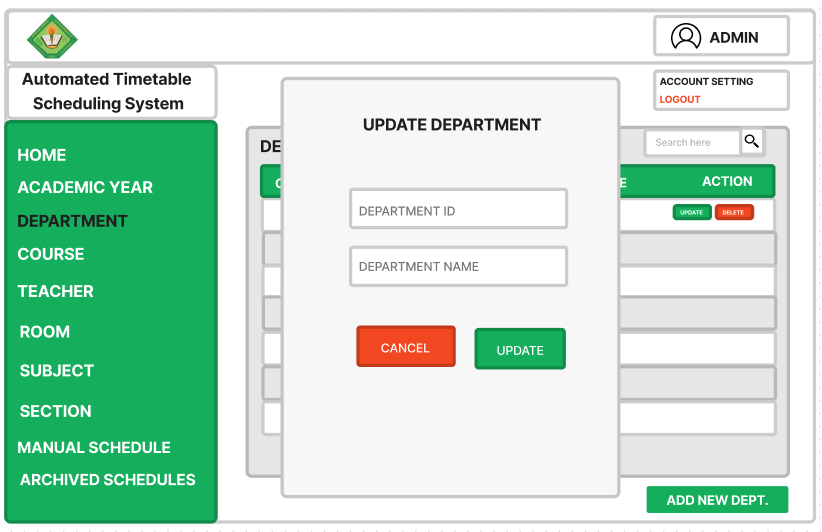
In this page the user can update a specific department, if there are some changes it can be done in update department

Figure 16. Updating department page

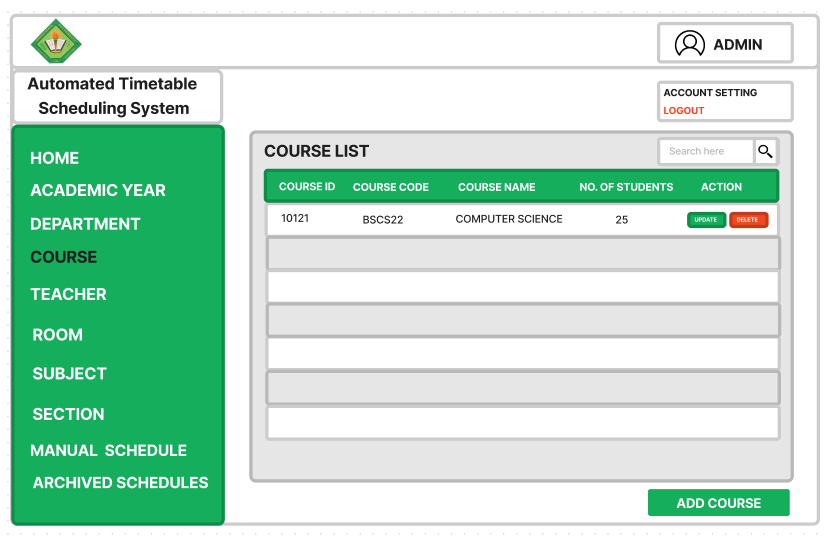
In course list page, the admin is allowed to view, add, update, and delete course by clicking the corresponding buttons.

Figure 17. Course list page

When the admin clicked the add course button, a form will appear and the admin need to input the necessary data to add course such as course id, course code, course name and the maximum number of students can be enrolled in the specific course.

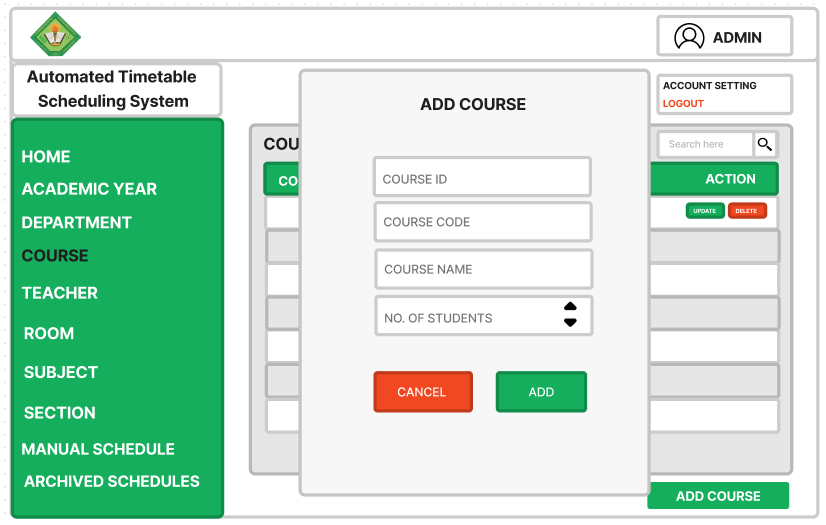
****

Figure 18. Add course list page

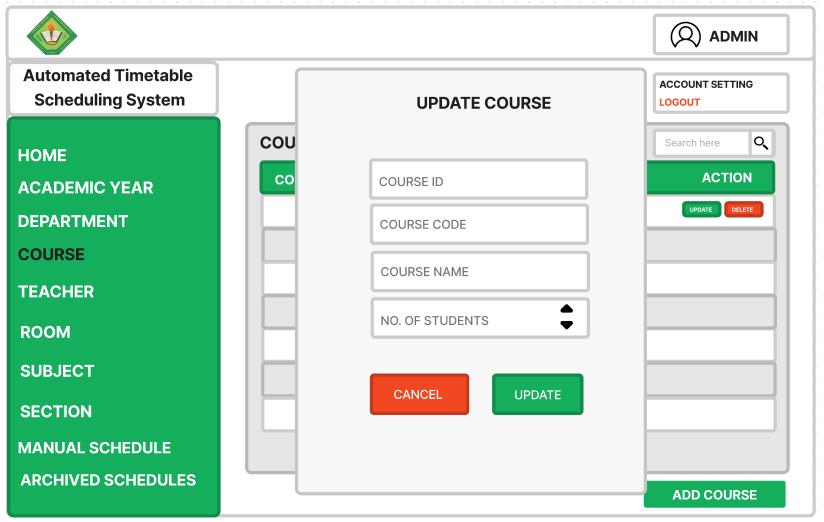
****Updating of courses is also available, if the course can now accommodate more students, it can be update by clicking update course

Figure 19. Updating course list page

In this page the admin can view the list of the teachers in the University with the following information the teachers id, name, and the department where the teachers belong. Adding, updating and deleting teacher is also available.

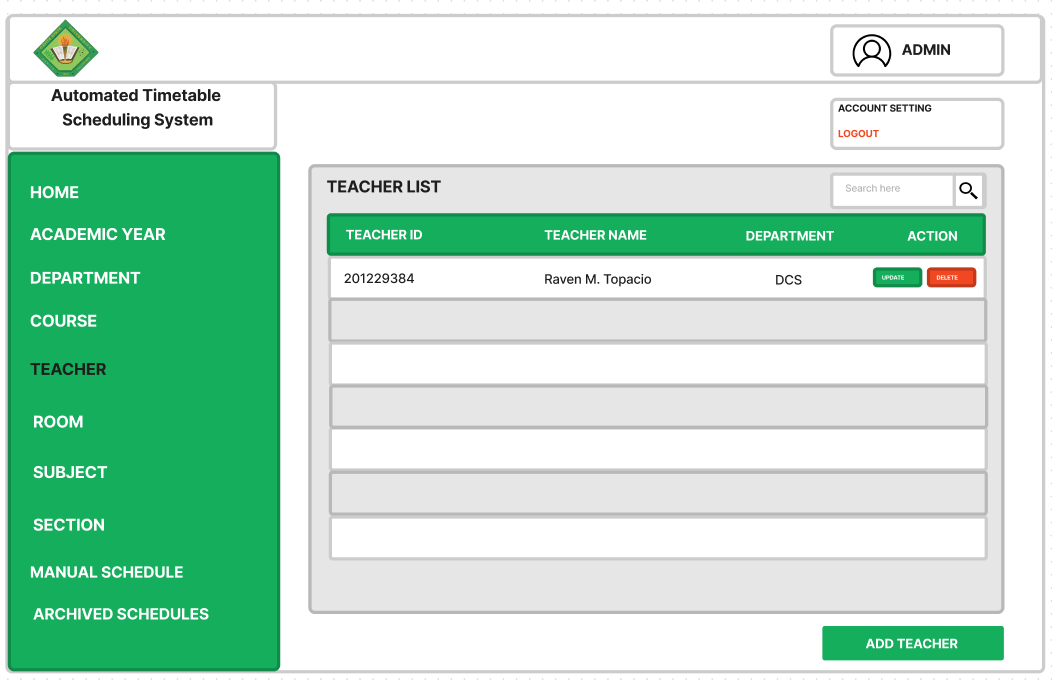


Figure 20. Teacher list page

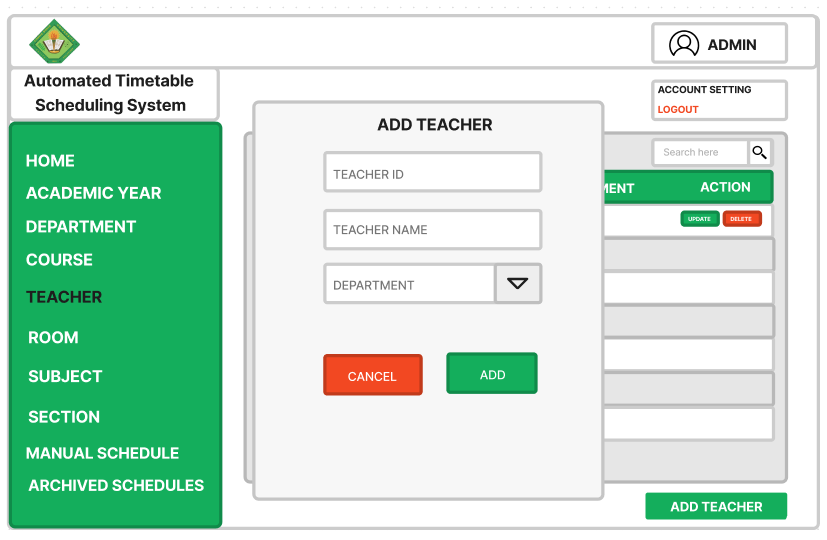
In this page the add teacher button is clicked, if there is a newly hire teacher the admin will just input the teacher id, teacher name and department then click the add button.

Figure 21. Adding teacher list page

Updating of teacher is also available, if there is something that need to be change in the data of teacher the admin will just simply update the information.

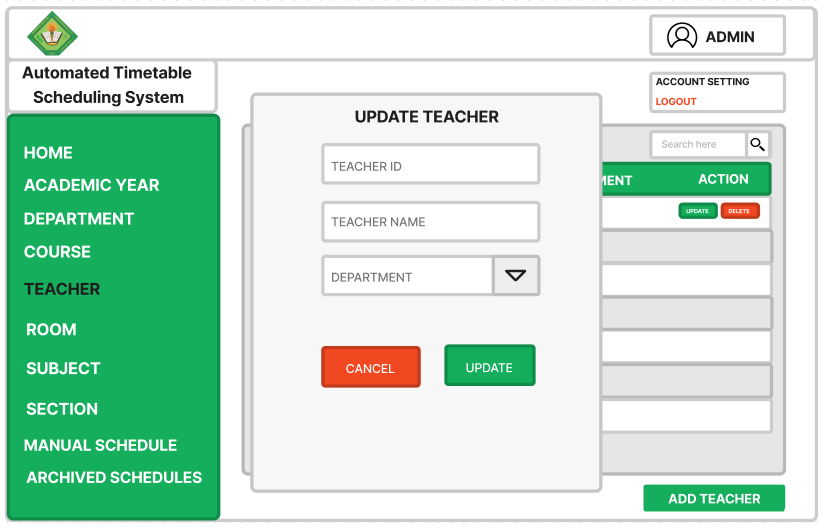
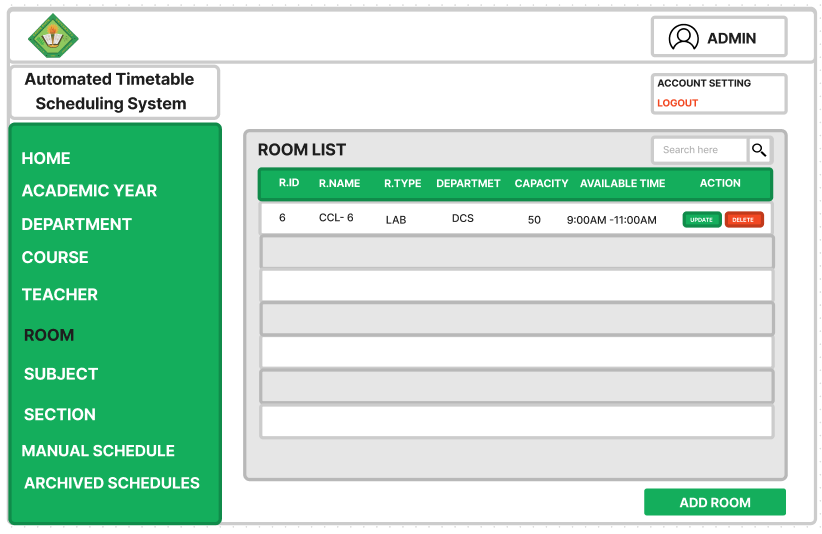


Figure 22. Updating teacher list page

In room list page the admin can view the following information about the room such as the room id, room name, department, capacity, the time of the availability of the room and adding of the user of the room. The admin is also allowed to add, update and delete room.

Figure 23. Room list page

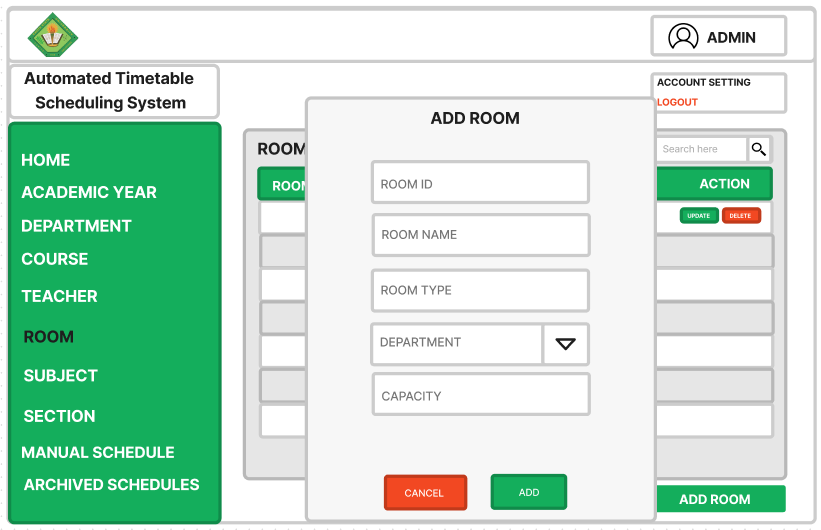
In this page the add room button is clicked, the admin needs to input necessary data in the form to update the room information.

Figure 24. Adding room list page

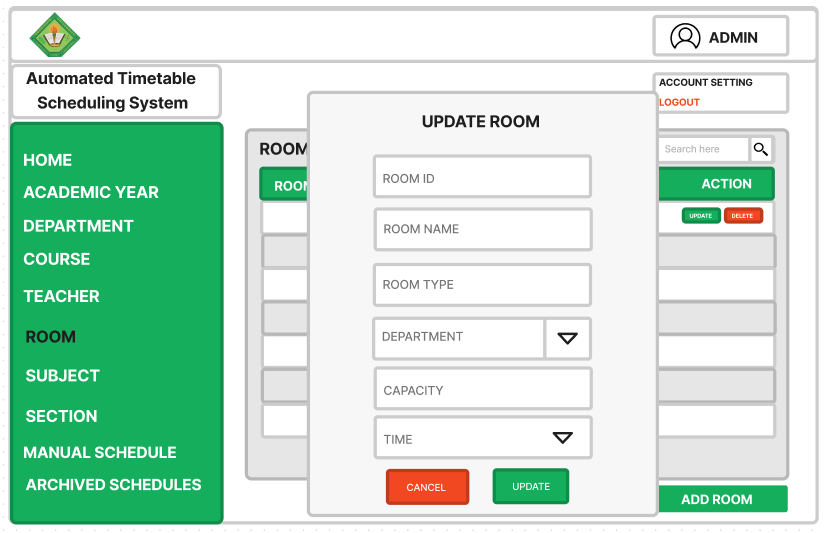
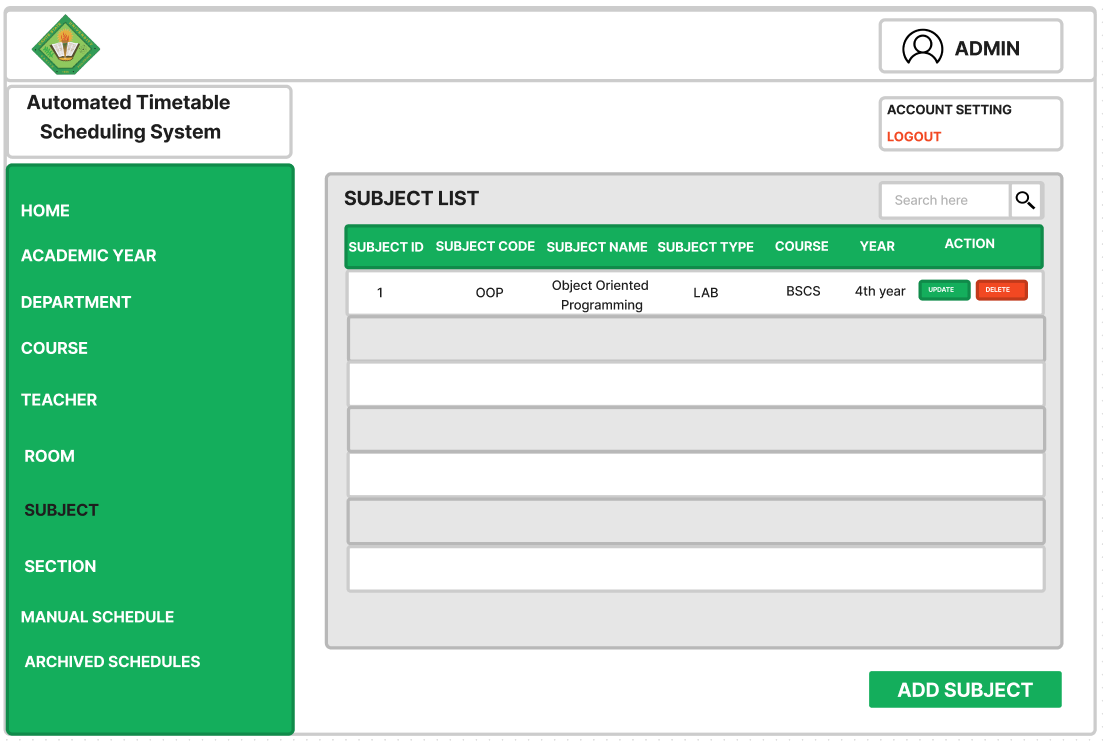
If the admin needs to update something in the room the admin needs to click the update button and a form will appear. The admin needs to input the necessary data to update the room information.

Figure 25. Updating room list page

In this page, there will be a list of subjects available for the semester. The admin is allowed to add, update, and delete the subject. The following data is needed to add subject such as subject id, subject name, subject type, course, and year.

Figure 26. Subject list page

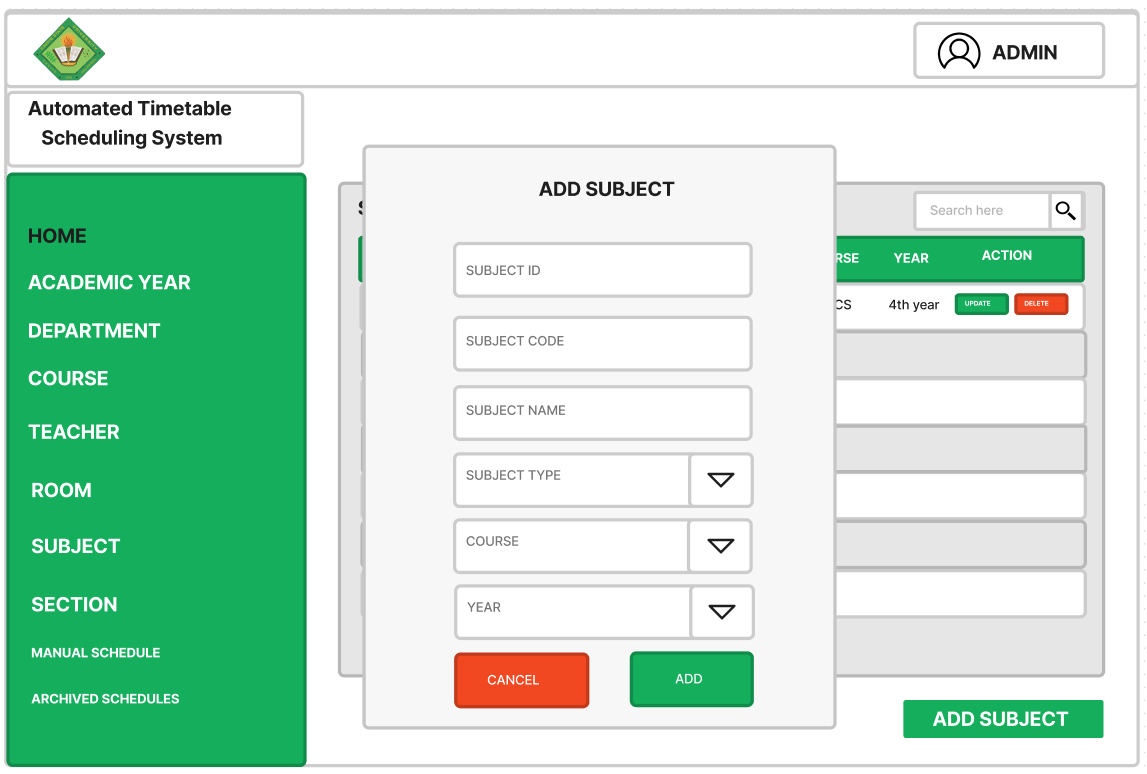
After the admin clicked the add button a form will appear, and the admin must fill out the information needed to add a new subject.

Figure 27. Adding subject list page

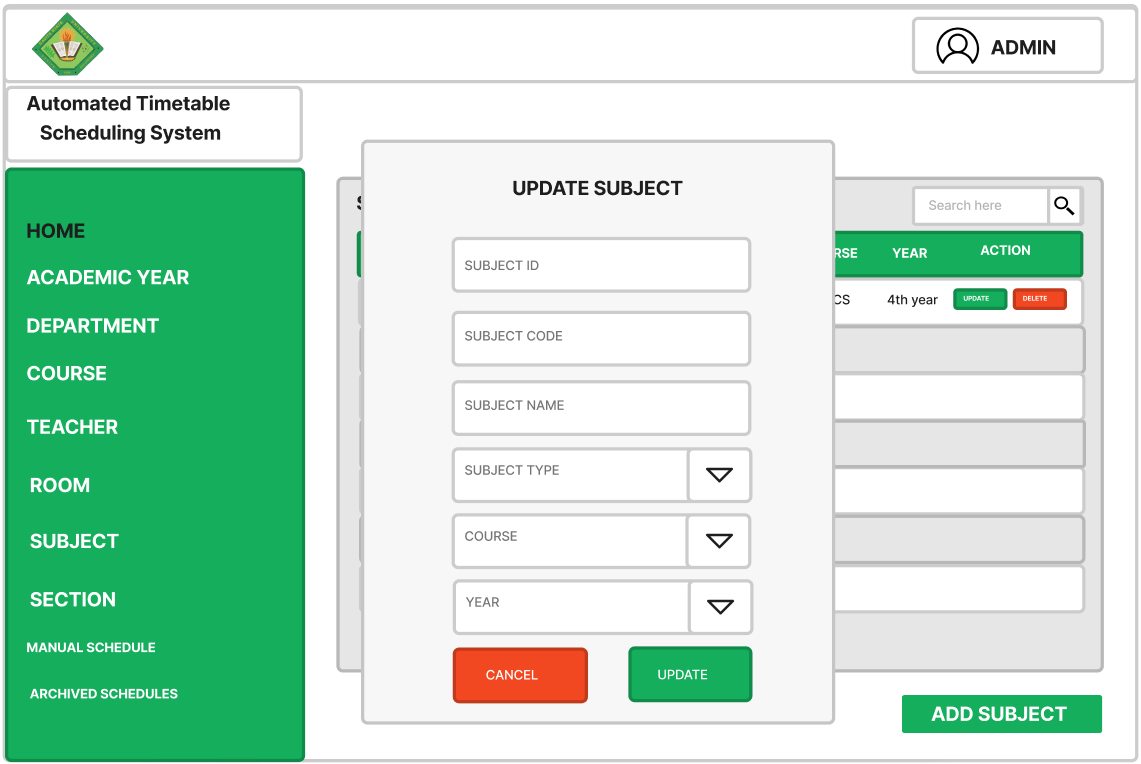
The admin can update the following information for the specific subject as shown in figure 28.

Figure 28. Updating subject list page

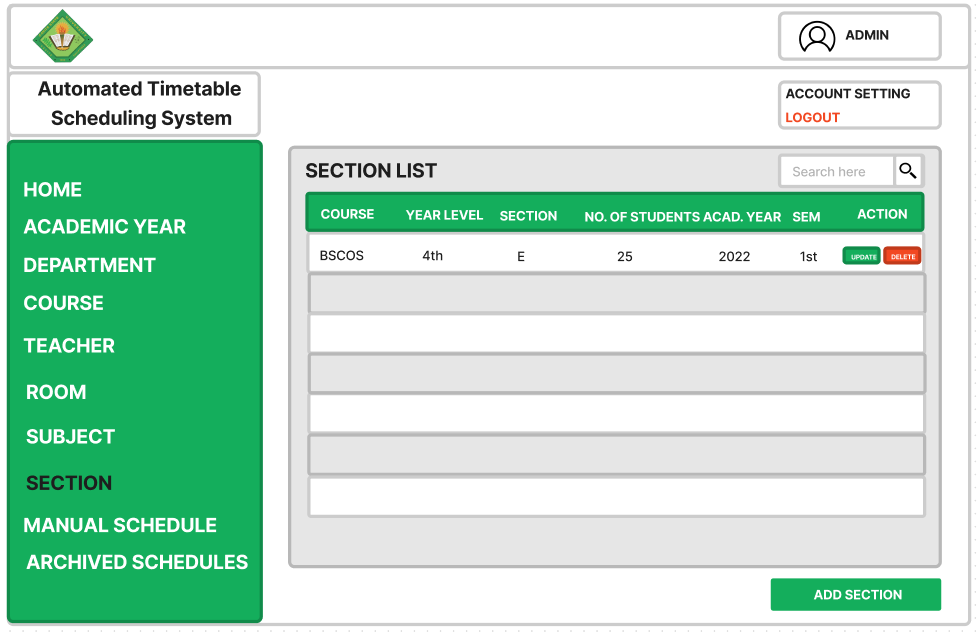
In the section list page, the course, year level, section, number of students, academic year and semester will be needed to add new section. In this page there is a list of section,

Figure 29. Section list page

If the add section button is clicked the admin must fill out the necessary information on the form to add new section.

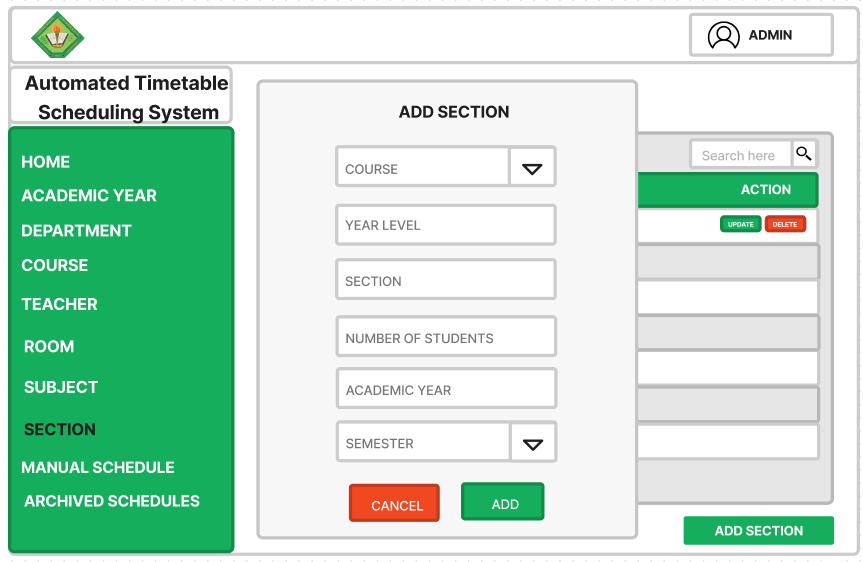


Figure 30. Adding section list page

To update the section, the admin must update the information as by entering new information on the form as shown in figure 31.

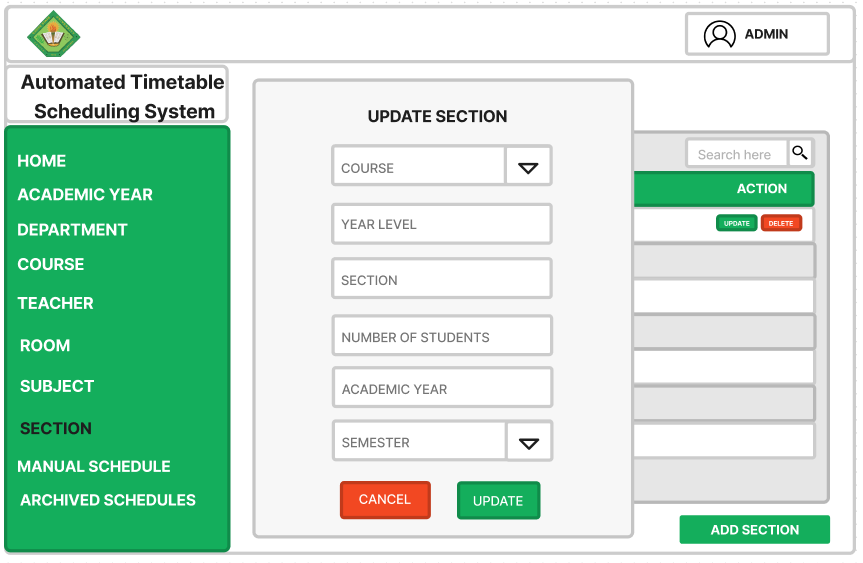
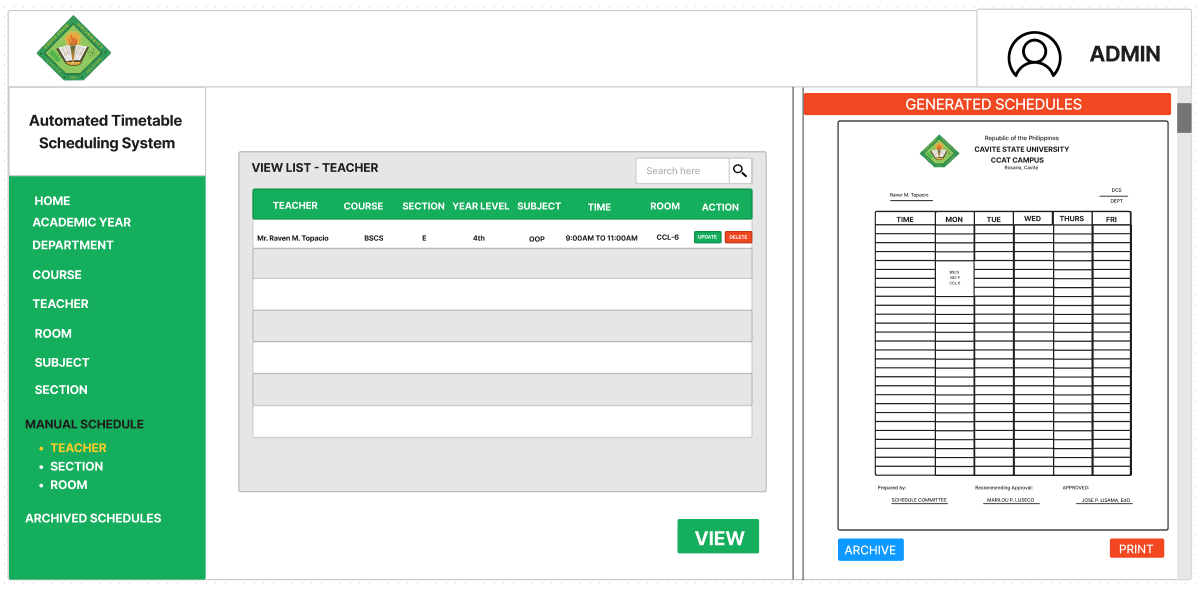


Figure 31. Updating section list page

In view list – teacher page, the admin is allowed to see the teacher’s name, section, course, year and the time of teacher schedule in specific class and also the admin can update and delete the schedule of the teacher

Figure 32. View list Teacher page

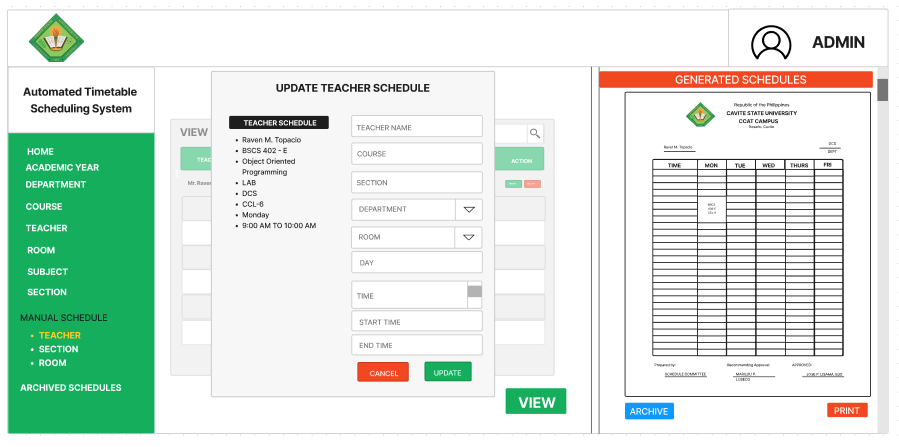
The admin can update the following information the teacher in this form

Figure 33. Updating view list Teacher page

When you click the print in view list teachers the system will automatically print a generated schedule for the specific teacher

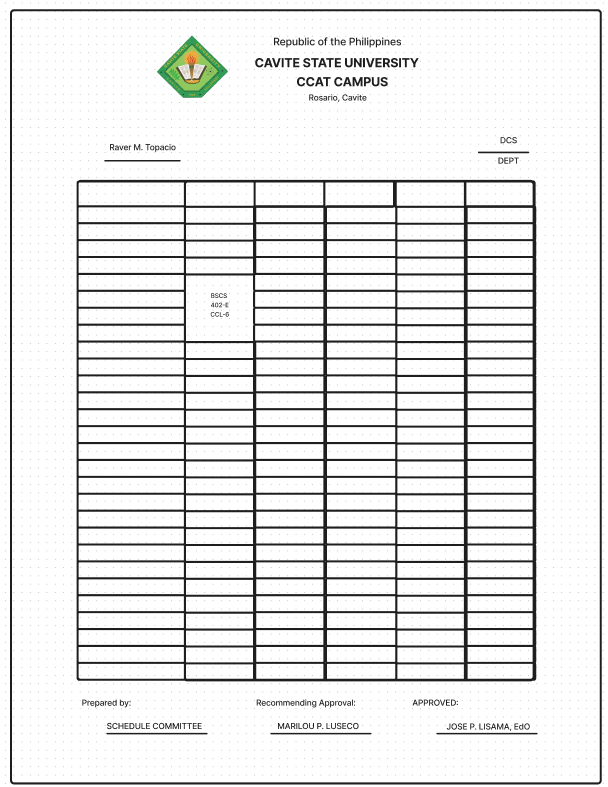


Figure 34. Generated Teacher Schedule

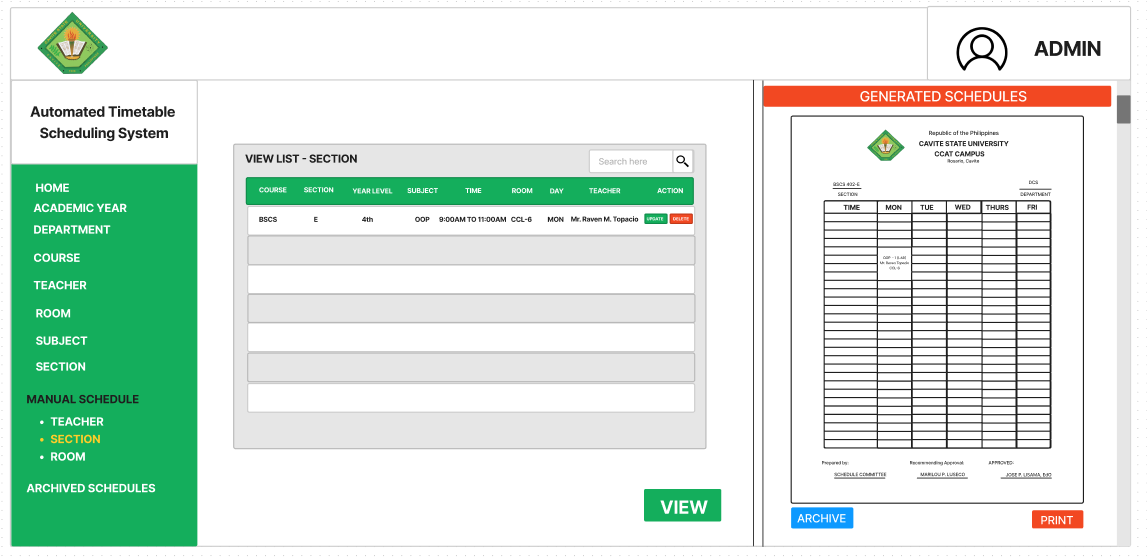
In this page, view list- section you can update and view the schedule of specific section at the same time.

Figure 35. View list section page

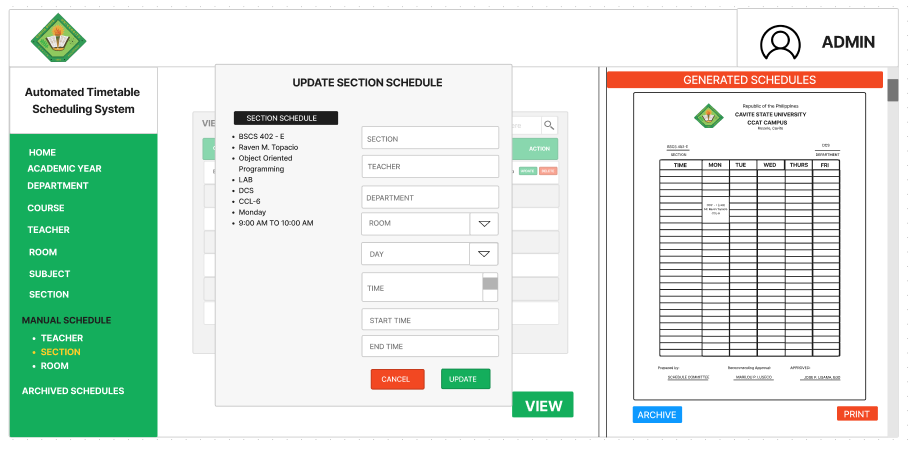
 In this page, the update button is clicked, and the admin is allowed to update the section schedule while viewing the generated schedules.

Figure 36. updating View list section page

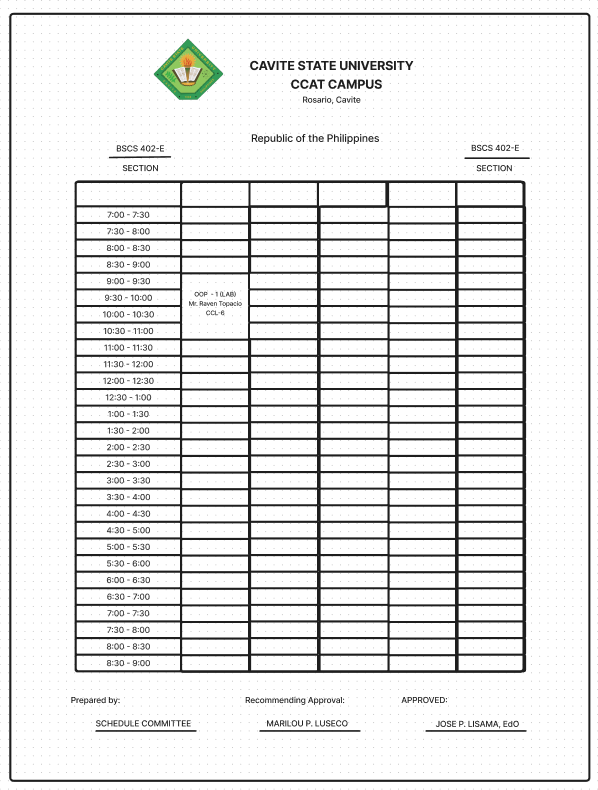
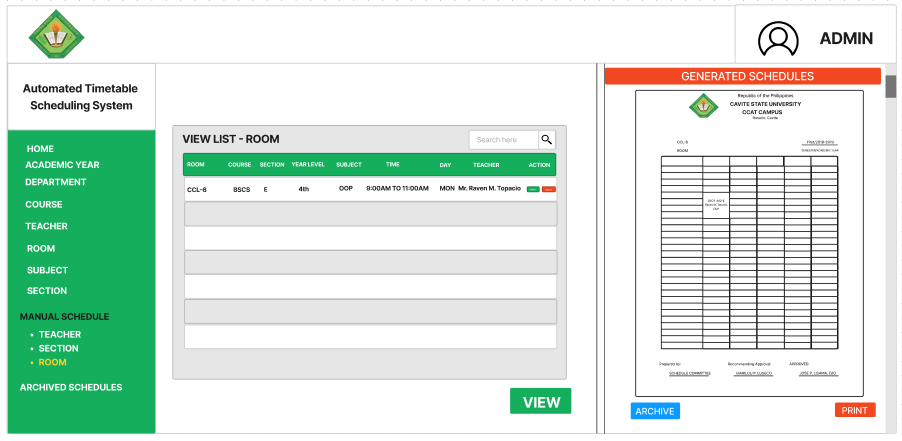
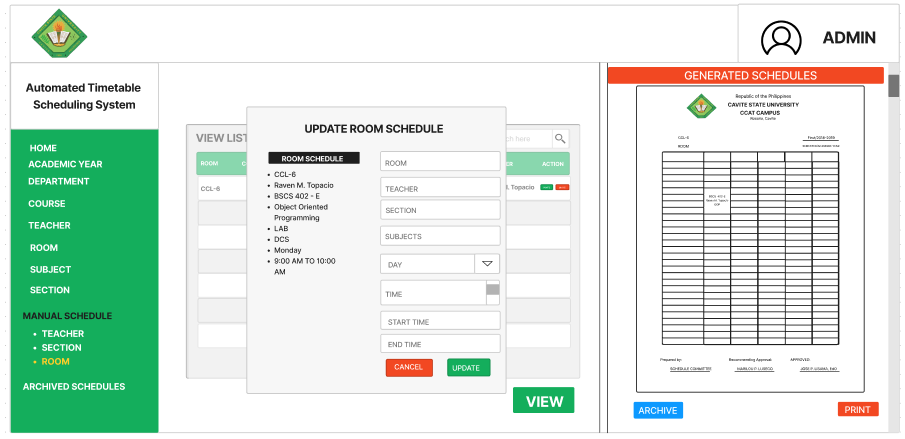
In this page, print button is clicked and the system will print the schedules of all section.

Figure 37. Generated Class Schedule

In this page of view list- room you can see the time and who will use the specific room. The admin is allowed to update and view the schedules at the same

Figure 38. View List-room page

In this page, the admin is allowed to update the schedule of specific section.

****Figure 39. updating View List-room page

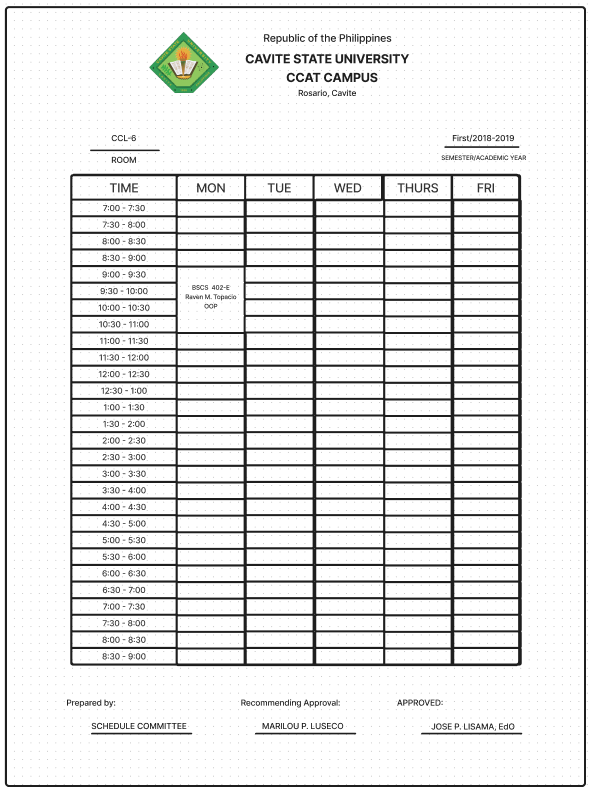
****When you click the print in view list room the system will automatically print a generated schedule for the specific room like the file below.

Figure 40. Generated Room page

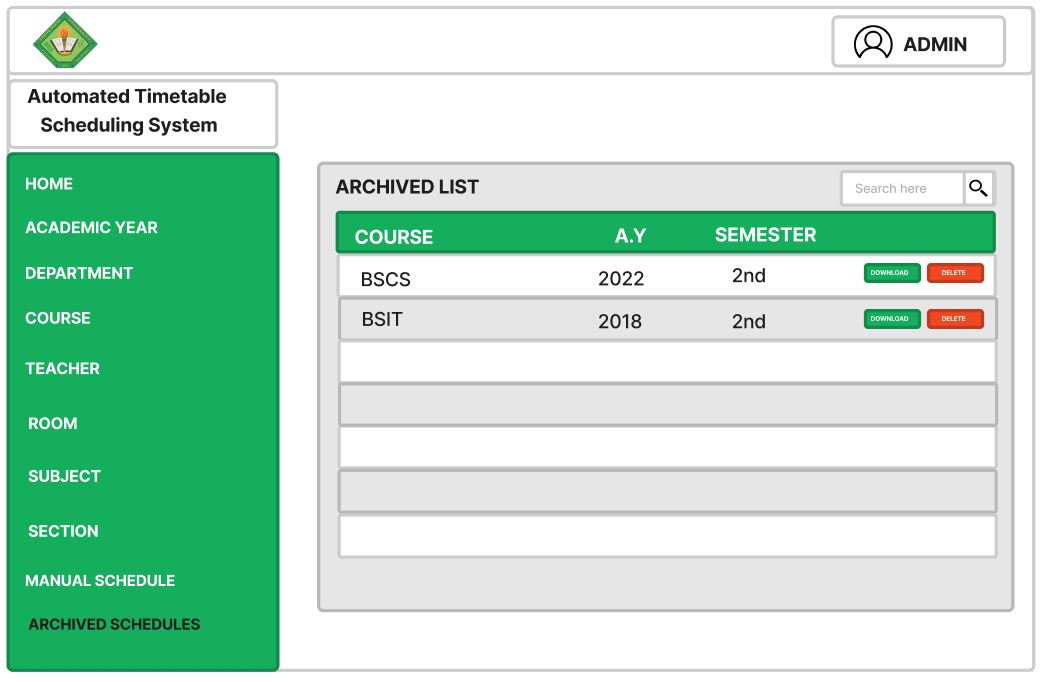
 In the archived list page, the admin can view the past generated schedule and download it as pdf file or delete it permanently

Figure 41. Archive list page

**Principles of Operation**

The Proponents will develop a scheduling system for Cavite State University CCAT Campus. This system will be administered by a person, it could be Teacher or University Staff. To use this system effectively, the administrator should login first then the administrator will be on the homepage where he/she can manage the system.  After login, the system will display the following features: Generate Schedule, View Schedule. Under the Generate Schedule Tab, the system will allow the administrator to choose between Automated Generation or Manual Generation. The Automated Generation method will allow the administrator to input data such as teacher information, rooms, course/programs etc. and the system will generate course schedules automatically. On the other hand, the Manual Generation will only generate schedules for specific subjects or courses. Lastly, in the View Tab the system will allow the administrator to view and print the generated schedules.

**Testing and Evaluation**

The developers will test the following:

* Reliability of the output – the proponents will test the generated output(timetables) based on the following criteria:
  + Teachers should not have same classes at the same time.
  + Subject should be taught two days per week.
  + Room should not have same schedule at the same time.
  + The room type should be corresponded to the subject type.
* Functionality – in this testing the proponents will perform alpha testing to test the following modules:
  + Login Module
  + Automated Generation Module

To test each module or tabs the proponents will use the test case table shown below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TEST CASE ID** | **OBJECTIVE** | **EXPECTED RESULTS** | **ACTUAL RESULTS** | **NUMBER OF DEBUGGED** |
| T-001 | Test the login module | 3 failed login attempts should be blocked by the system. | 3 failed login attempts were blocked by the system. | 1 |
|  |  |  |  |  |
|  |  |  |  |  |

* **Test Case ID –** this is the unique identification number and to identify what is being tested.
* **Objective –** this is the primary goal of the testing.
* **Expected Results –** this is the expected outcome of the test.
* **Actual Results –** it refers to the real outcome of the testing.
* **Number of Debugged -**it refers to the number of debug the tester made to achieve the objective.

**Data Analysis**

The proponents will evaluate 50 respondents as shown below:

Table 1:  Evaluators description

|  |  |
| --- | --- |
| **Evaluators** | **Number of Evaluators** |
| Web Developers | 10 |
| Students | 30 |

The proponents will compute the evaluation results by getting the total of each criterion then divide it by the total evaluators and interpret the total score using the table below:

Table 2:  Descriptive Interpretation of total score

|  |  |
| --- | --- |
| **SCORE** | **INTERPRETATION** |
| 4.51 -5.00  3.51 - 4.50  2.51 - 2.50  1.51 - 2.50  1.00 - 1.50 | Excellent  Very Satisfactory  Satisfactory  Unsatisfactory  Needs Improvement |

**Implementation Plan or Deployment**

The implementation of the system will follow the succeeding schedules:

1. Request for the campus approval of the system to be implemented in CVSU CCAT Campus.
2. Checking for the campus equipment capabilities to use the system.
3. Installation of the system prototype.
4. Testing of the system with large data for 2 weeks.
5. The results of the testing will be analyzed.
6. If the system passes the certain criteria, the system will be used, and the output of the system will be linked to the CVSU CCAT Campus Portal.

**REFERENCES**

Al-Jarrah, M. A., Al-Sawalqah, A. A., & Al-Hamdan, S. F. (2017). Developing a course timetable system for academic departments using genetic algorithm <https://www.researchgate.net/profile/Sami-Hamdan-4/publication/313487752_Developing_A_Course_Timetable_System_for_Academic_Departments_Using_Genetic_Algorithm/links/601b7a55299bf1cc26a00768/Developing-A-Course-Timetable-System-for-Academic-Departments-Using-Genetic-Algorithm.pdf>

“Simplelearn” (2022), What Is Data Collection: Methods, Types, Tools, and Techniques <https://www.simplilearn.com/what-is-data-collection-article>

Ben Moreland, (2015), Hard and Soft timetabling constraints. <https://educationspaceconsultancy.com/hard-and-soft-timetabling-constraints-not-only-a-working-timetable-but-a-great-timetable/>

“CQL” (2022), 12 Reasons Why Data is Important. [https://www.c-q-l.org/resources/guides/12-reasons-why-data-is important/#:~:text=Data%20allows%20organizations%20to%20measure,changed%20over%20the%20long%2Dterm](https://www.c-q-l.org/resources/guides/12-reasons-why-data-is%20important/#:~:text=Data%20allows%20organizations%20to%20measure,changed%20over%20the%20long%2Dterm).

Paymon Khamooshi (2019), The Benefits of using a web-based application. <https://www.geeks.ltd.uk/insights/the-benefits-of-using-web-based-applications>

Souvik - “ Apps and Software on RSWEBSOLS” (2022), Desktop vs Web Application<https://www.rswebsols.com/tutorials/software-tutorials/desktop-vs-web-application#:~:text=If%20the%20target%20audience%20is%20consumers%2C%20then%20a%20web%20application,easily%20than%20a%20desktop%20application>

Edmund Kieran Burke, Sanja Petrovic (2002), Recent research   directions in automated timetabling <https://doi.org/10.1016/S0377-2217(02)00069-3>

Zina Houhamdi, Belkacem Athamena, Rani Abuzaineddin, Mohammad Muhairat (2019), A Multi-Agent System for Course Timetable Generation. <https://www.researchgate.net/publication/331995150_A_multi-agent_system_for_course_timetable_generation>

Achini Kumari Herath (2017), Genetic Algorithm for University Course Timetabling Problem. <https://egrove.olemiss.edu/cgi/viewcontent.cgi?article=1442&context=etd>

Vijini Mallawaarachchi (2017), Introdcution to Genetic Algorithm <https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3>

Ahmed Gad (2018), Introduction to Optimization with Genetic Algorithm. <https://www.kdnuggets.com/2018/03/introduction-optimization-with-genetic-algorithm.html>

Manoj Kumar, Mohammad Husian , Naveen Upreti & Deepti Gupta (2010), Genetic Algorithm: Review and Application. <https://deliverypdf.ssrn.com/delivery.php?ID=636105110097094064001029120115073108005045006058036003100112031070126064024096098089061030040044008109118027108003096021077069020059035013080124072116027072001083096039050008024124117065120072103079096018006114119066080102069114126013020094068121120094&EXT=pdf&INDEX=TRUE>

Wonjae Lee, Hak-Young Kim (2005), Genetic Algorithm Implementation in Python.  <https://www.researchgate.net/publication/4178596_Genetic_algorithm_implementation_in_Python>

Ander Fernández Jauregui (2022), Genetic Algorithm in Python. <https://anderfernandez.com/en/blog/genetic-algorithm-in-python/>

“Django Software Foundation” (2022), Meet Django. <https://www.djangoproject.com/>

**Related System**

“Neon Projects” (2015), Automated College Timetable Generator. <https://nevonprojects.com/automated-college-timetable-generator/>

Gamale , Jeralyn M., Anuta, Edward C., Sayson, Zina D. (2011), Automated Scheduling System Capstone Project. <https://capstoneguide.com/automated-class-scheduling-system-capstone-project-document/>

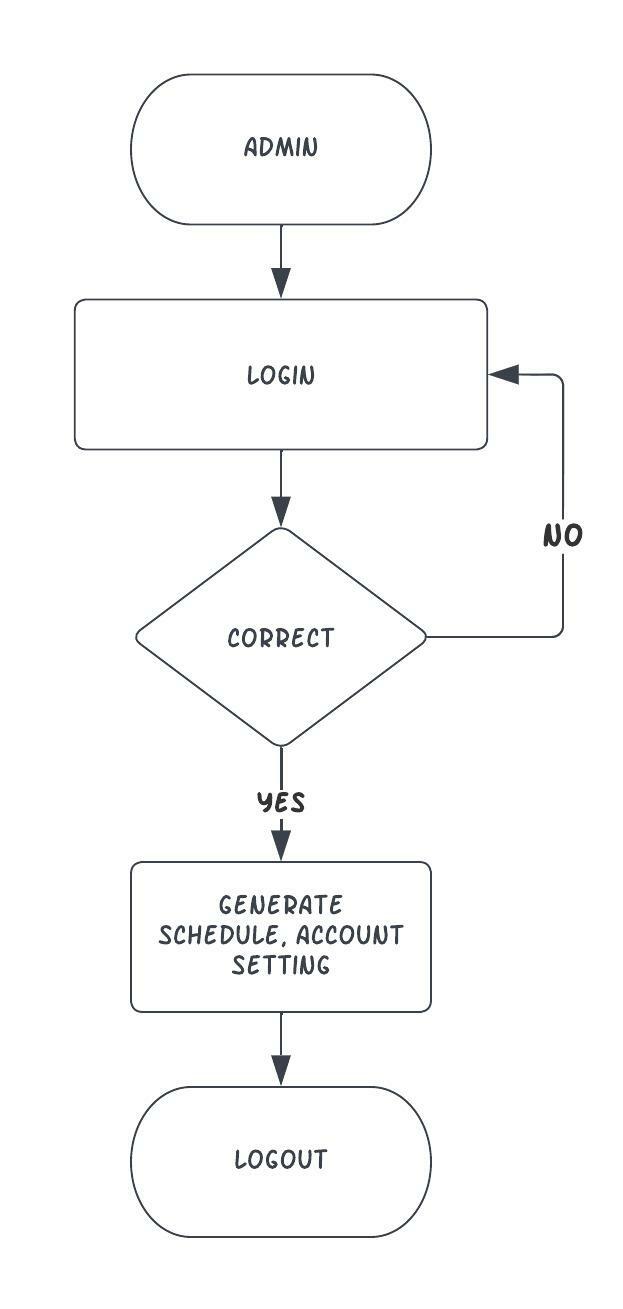
John Benedict C. Legaspi, Roman M. De Angel, Ace C. Lagman, John Heland Jasper C. Ortega (n.d), Web based Course Scheduling System using Greedy Algorithm. <https://ijssst.info/Vol-20/No-S2/paper14.pdf>

Abhik Mitra (2020), Intelligent Timetable Generator using Genetic Algorithm. <http://abhik.pythonanywhere.com/>

Loic Ngou (2019), Timetable Generator using Genetic Algoritmh. https://Timetable Generator

**APPENDIX TABLES**

**APPENDIX FIGURES**



Appendix Figure 1. Admin Flowchart

Diagram

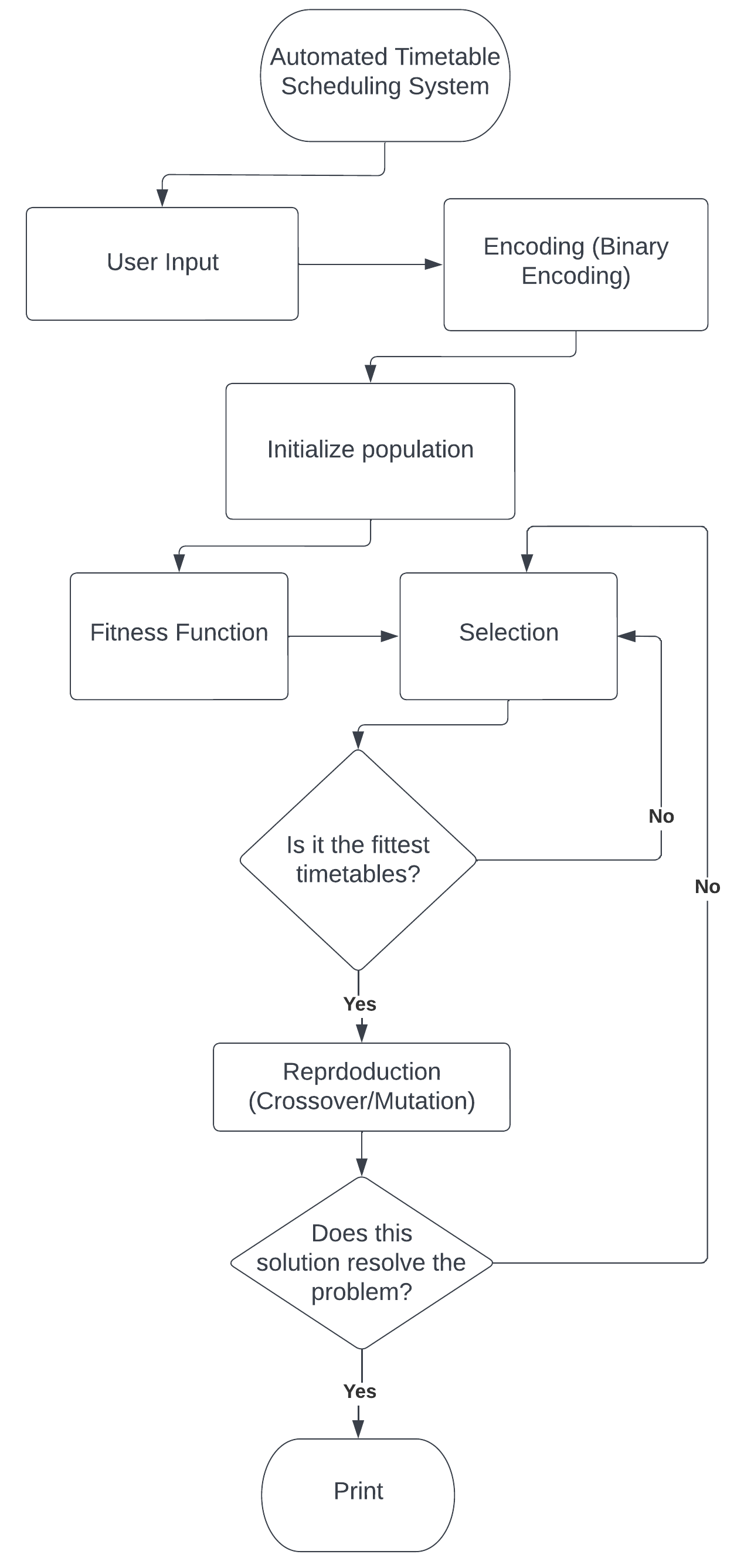
Description automatically generated

Appendix Figure 2. Automated Generation Flowchart

Diagram

Description automatically generated

Appendix Figure 3. Manual Update Flowchart



Appendix Figure 4. System Process flow chart

Diagram

Description automatically generated

Appendix Figure 5. Admin Use Case Diagram pt.1

**Diagram

Description automatically generated**

Appendix Figure 6. Admin Use Case Diagram pt.2

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **TASK** | **2022** | | | | | | | |
| **MAY** | **JUNE** | **JULY** | **AUG.** | **SEPT.** | **OCT.** | **NOV.** | **DEC.** |
| **1** | **Requirements Analysis and Design phase** |  | | |  |  |  |  |  |
| **2** | **Prototype Cycle Phase** |  |  |  |  |  |  |  |  |
| **3** | **Developing and Testing Phase** |  |  |  |  | | | | |
| **4** | **Deployment Phase** |  |  |  |  |  |  |  |  |

Appendix Figure 7. Gantt Chart

**Diagram

Description automatically generated**

Appendix Figure 8. Entity Relationship Diagram

**APPENDICES**