

A Project Report

On

**“Examination Timetable generation”**

Batch Details

|  |  |  |
| --- | --- | --- |
| Sl. No. | Roll Number | Student Name |
| 1 | 20211CSE0361 | Trishanya |
| 2 | 20211CSE0394 | Bhavith Mourya R |
| 3 | 20211CSE0377 | Dinson Sam Thomas |
| 4 | 20211CSE0384 | Ajay R |

**School of Computer Science,**

**Presidency University, Bengaluru.**

Under the guidance of,

Ms. Rakheeba Taseen

School of Computer Science,

Presidency University, Bengaluru

**CONTENTS**

1. Introduction
2. Literature Review
3. Objectives

## Methodology

## Architecture diagram

1. Timeline for Execution of Project
2. Expected Outcomes
3. Conclusion
4. References
5. **INTRODUCTION**

Generating a timetable is crucial for effectively managing educational institutions and has a significant impact on the academic experience of students and faculty. It requires balancing various factors, including course accessibility, exam scheduling, and resource allocation, while also addressing conflicting demands such as student registrations, faculty availability, and room capacities.

Conventional approaches often involve manual processes, which can result in errors and inefficiencies, such as scheduling overlapping exams for students and faculty unavailability. As educational institutions grow and diversify, there is an increasing demand for automated and optimized solutions to ensure a seamless and efficient academic experience.

The main focus of this project is to develop a system that automatically generates examination timetables tailored specifically for Presidency University. The main objective is to create an efficient scheduling solution that takes into account various important factors, including:

* Course Registration Information: It is crucial to understand which students are enrolled in which courses in order to avoid exam conflicts. The system needs to analyze the registration data to identify potential clashes for students taking multiple exams on the same day.
* Elective Information: Scheduling can become complex due to varying student elective selections. The system must be able to accommodate different elective combinations and ensure that all students have access to their chosen subjects without conflicts.
* Faculty Availability: Faculty members play a vital role in the examination process, often serving as invigilators. Their availability must be considered in the scheduling algorithm to avoid situations where faculty members are double-booked or unavailable for assigned exams.
* Constraints: The scheduling process needs to incorporate several defined constraints, which include:
* Hard Constraints: These are mandatory requirements that must be fulfilled, such as ensuring that there are no overlapping exams for students enrolled in multiple courses.
* Soft Constraints: These are preferences for minimizing student fatigue, for example, ensuring that no student has more than three hours of examinations or invigilation in a single day.

By utilizing optimization techniques and algorithms, the proposed system aims to automate the timetable generation process, ensuring that it meets the specified constraints while maximizing efficiency and fairness in scheduling. This project not only tackles the immediate challenges of timetable generation but also aims to provide a scalable solution that can adapt to future changes in course offerings, student enrollment, and faculty assignments.

1. **LITERATURE REVIEW**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.No** | **TITLE OF THE PAPER/AUTHOR/PUBLISHER/YEAR** | **ADVANTAGES** | **LIMITATIONS** |
| 1. | Automatic Timetable Generator  Authors: Prof. Jyothi Patil, Shambhavi V, Sneha N T, Sweta Jadhav, Tahura Sadaf  IJRASET,2023 | The Automatic Timetable Generator efficiently automates the scheduling process, significantly reducing manual effort and minimizing conflicts in examination and class schedules. | potential scalability issues with very large datasets, sensitivity to initial conditions in genetic algorithms |
| 2. | AUTOMATIC TIMETABLE GENERATION SYSTEM  Authors:Rajshri Firke, Pratiksha Bhabad, Omkar Gangarde, Abhimannyu Magar, Prof. Anuja Tawlare  IJCRT,2023 | The Automatic Timetable Generation System streamlines the scheduling process by automating timetable creation, ensuring optimal use of resources and reducing scheduling conflicts. | handling large-scale user demands, complexities in ensuring data privacy and security, and the need for ongoing updates to address user feedback |
| 3. | Design and Implementation of An Automatic Examination Timetable Generation and Invigilation Scheduling System Using Genetic Algorithm  Authors:Abdulaziz Aminu;WahyuCaesarendra;UmarSHaruna;Abubakar Sani;MansurSa'id;Daniel S Pamungkas;Sumantri R Kurniawan;  Endang Kurniawan  IEEE,2019 | The system utilizes a genetic algorithm to efficiently generate optimal examination timetables and invigilation schedules, minimizing conflicts and enhancing resource allocation. | limited accessibility as it's currently desktop-based, and challenges in handling complex timetable conflicts |
| 4. | An integer programming approach to curriculum-based examination timetabling  Authors:Cataldo, A., Ferrer, J.-C., Miranda, J., Rey, P. A., & Saure, A.  Annals of Operations Research-2017 | The study employs an integer programming approach to effectively address curriculum-based examination timetabling, providing optimal scheduling solutions that adhere to various constraints. | computationally expensive and may struggle with larger datasets due to the complexity of the integer programming formulation, leading to longer solution times. |
| 5. | Multi-objective optimization for exam scheduling to enhance the educational service performance  Authors:Abdallah, K. S  The Journal of Management and Engineering Integration-2016 | The study introduces a multi-objective optimization approach that improves exam scheduling efficiency, thereby enhancing overall educational service performance. | The reliance on heuristic methods like genetic algorithms can result in long computational times and difficulty in ensuring consistent quality of solutions across diverse scheduling scenarios. |
| 6. | A great deluge algorithm for a real-world examination timetabling problem  Authors:Mohmad Kahar, M. N., & Kendall, G.  Journal of the Operational Research Society-2015 | The study demonstrates the effectiveness of a great deluge algorithm in solving real-world examination timetabling problems, providing a robust method for optimizing scheduling while handling various constraints. | tuning of parameters, and its performance can vary significantly depending on the specific characteristics of the timetabling problem being addressed. |
| 7. | Real-life curriculum-based timetabling  Authors: Müller, T., & Rudová, H  Proceedings of the 9th International Conference on the Practice and Theory of Automated Timetabling—PATAT-2012 | The research explores real-life curriculum-based timetabling challenges and solutions, emphasizing practical applications of automated scheduling techniques to enhance efficiency and effectiveness. | Constraint programming can be limited by scalability issues, as it may struggle with performance in larger, more complex scheduling scenarios due to the increased number of constraints and variables involved. |
| 8. | A survey of search methodologies and automated system development for examination timetabling  Authors:Qu, R., Burke, E., McCollum, B., Merlot, L., & Lee, S.  Journal of Scheduling-2009 | This survey analyzes diverse search methodologies for automated examination timetabling, offering insights into their effectiveness and guiding future developments in scheduling systems. | The potential inefficiencies in finding optimal solutions are a limitation of these methodologies, as many of them depend on shortcuts that could result in local optima instead of globally optimal schedules. |
| 9. | A survey and case study of practical examination timetabling problems  Authors: Cowling, P., Kendall, G., & Hussin, N. M.  Journal of Scheduling-2001 | The survey presents a detailed analysis of practical examination timetabling challenges, offering valuable insights and solutions based on real-world case studies | The use of metaheuristic techniques can result in unreliable outcomes, as their performance is greatly influenced by the configuration of parameters and the intricacy of the scheduling scenarios they are applied to. |
| 10. | Examination timetabling: Algorithmic strategies and applications.  Authors:Carter, M. W., Laporte, G., & Lee, S. Y.  Journal of the Operational Research Society-1996 | The study provides a comprehensive overview of various algorithmic strategies for examination timetabling, highlighting their practical applications and effectiveness in optimizing scheduling solutions. | The techniques can struggle with scalability and efficiency, as greedy algorithms may yield suboptimal solutions and integer programming can be computationally intensive for large datasets. |

1. **OBJECTIVES**

* Automating the timetable generation process to minimize manual effort and errors.
* Ensuring conflict-free scheduling by taking into account all pertinent constraints, such as faculty and room availability.
* Minimizing student and faculty exhaustion by restricting the number of exams in a day.
* Creating a user-friendly interface for simple data input and timetable adjustments.

**EXPERIMENTAL DETAILS/METHDOLOGY**

**Hardware:**

* Personal Computer with at least 8GB RAM, Intel i5 processor.

**Software:**

* Java Development Kit (JDK)
* Spring Boot Framework
* MySQL Database
* Eclipse IDE
* HTML/CSS/JavaScript for the front end

**Modules used:**

1. User Interface Module

* Purpose: To create forms for user input and display the generated timetable.
* Technologies: HTML, CSS, JavaScript

2. Authentication Module

* Purpose: To manage user login and access control.
* Technologies: Spring Security

3. Timetable Generation Module

* Purpose: To generate the timetable based on input data and constraints.
* Technologies: Java

4. Data Access Module

* Purpose: To interact with the database.
* Technologies: Spring Data JPA, MySQL

5. Constraint Management Module

* Purpose: To define and apply constraints for timetable generation.
* Technologies: Java

**4. METHODOLOGY**

Step 1: Data Collection and Storage

* Input Data:
  + Course registration and elective information will be gathered from the university’s records and stored in a MySQL database.
  + Faculty availability for invigilation duties and room capacities will also be stored.
* Database Setup:
  + Design the database schema using MySQL to store details such as courses, students, exam durations, rooms, and faculty schedules.
  + Each student’s registered courses and electives will be linked to avoid scheduling conflicts.

Step 2: Define Constraints

* Hard Constraints:
  + A student cannot have two exams at the same time.
  + Faculty members cannot invigilate multiple exams simultaneously.
  + Room capacities must be respected during exam assignments.
  + No exams should exceed the room’s available hours.
* Soft Constraints:
  + Minimize the number of exams scheduled per day for students and faculty to avoid exhaustion.
  + Distribute exams evenly over the examination period to prevent student and faculty burnout.

Step 3: Backend Development (Java and Spring Boot)

* Java will be used to implement the core logic that handles the scheduling algorithm.
* Spring Boot will serve as the application framework, facilitating communication between the frontend, backend, and database.
* Scheduling Algorithm: Use optimization techniques to slot exams in the timetable while respecting the constraints.
  + The timetable generation logic will pull data from the MySQL database and process it to generate valid, conflict-free exam slots.

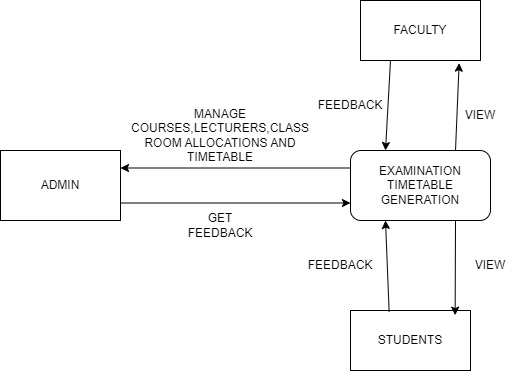
Step 4: Frontend Development (HTML/CSS/JavaScript)

* A user-friendly web interface will be developed using HTML/CSS/JavaScript.
  + The interface will allow administrators to:
    - Input course and faculty details.
    - View and adjust the generated timetable.
  + The interface will display timetables in a structured, easy-to-read format.

Step 5: Timetable Generation

Spring Boot will act as the middle layer, receiving input from the frontend, processing it with the scheduling algorithm, and storing the results in the database.

**5.ARCHITECTURE DIAGRAM**

****

**6.OUTCOMES**

The Presidency University's automated examination timetable generation project aims to achieve the following outcomes:

* Creation of a well-structured timetable that prevents exam overlaps for students taking multiple courses, ensuring fair access to exams.
* Efficient allocation of faculty and examination rooms to minimize conflicts and maximize resource utilization.
* Reduction of administrative workload through automation, enabling staff to focus on other important tasks while producing accurate and timely schedules.
* Improvement of the academic experience for students by providing a fair and manageable examination schedule, thus reducing stress and fatigue.
* Development of a flexible system that can easily adapt to changes in course offerings, student enrollments, and faculty assignments, ensuring long-term usability.

**7. TIMELINE OF THE PROJECT/ PROJECT EXECUTION PLAN**

* First Week of September: Project title selection, Requirement analysis, challenges, problem identification
* Mid-September: Completion of the 0th review, covering the project proposal, objectives, and basic implementation strategy.
* 18th October: First review, focusing on progress, early implementation of the timetable generation logic, and addressing initial constraints.
* End of October: Midway implementation review, completion of most functionality (input processing, conflict detection, resource allocation).
* Mid-November: Testing phase begins, reviewing the timetable generation results, fine-tuning the optimization process, and handling complex constraints like invigilation scheduling.
* End of November: Final review of the project, system testing, and bug fixes.
* Before December: Project completion, documentation, and submission.

**8. CONCLUSION**

The project developed for Presidency University to automate the creation of examination timetables represents a significant advancement in academic scheduling management. By utilizing advanced algorithms and optimization techniques, the project effectively addresses the complexities associated with generating a fair and efficient timetable that caters to the diverse needs of both students and faculty. The implementation of this system not only simplifies the scheduling process but also reduces conflicts, ensuring that students can take their exams without overlap, and faculty members are available for supervision. By automating a process that was previously manual and prone to errors, the project lightens the administrative workload, enabling staff to concentrate on more critical tasks within the educational institution. Additionally, the proposed solution's adaptability and scalability ensure that it can accommodate future changes in course offerings, student enrollments, and faculty assignments. This flexibility is essential as educational institutions continue to evolve and expand in response to changing academic landscapes. The successful implementation of this automated timetable generation system will improve the overall academic experience at Presidency University, cultivating a more organized and equitable examination process. This project not only aims to enhance operational efficiency but also contributes to the broader objective of supporting student success and satisfaction in their educational journey.

**9. REFERENCES**

* Lindahl, M., Sørensen, M., & Stidsen, T. R. (2018). A fix-and-optimize matheuristic for university timetabling. Journal of Heuristics, 24(4), 645–665. (<https://doi.org/10.1007/s10732-018-9371-3>)
* Cataldo, A., Ferrer, J.-C., Miranda, J., Rey, P. A., & Saure, A. (2017).An integer programming approach to curriculum-based examination timetabling. Annals of Operations Research, 258(2), 369–393. (<https://doi.org/10.1007/s10479-016-2321-2>)
* Abdallah, K. S. (2016). Multi-objective optimization for exam scheduling to enhance the educational service performance. The Journal of Management and Engineering Integration, 9(1), 14–23.
* Carter, M. W., Laporte, G., & Lee, S. Y. (1996). Examination timetabling: Algorithmic strategies and applications. Journal of the Operational Research Society, 47(3), 373–383. (<https://doi.org/10.1057/jors.1996.37>)
* <https://programmer2programmer.net/live_projects/Synopsis/download_project_synopsis.aspx?JAVA-Time-Table-Generation-System-JAVA-MySQL&id=145>
* <https://stackoverflow.com/questions/12209268/scheduling-timetable-algorithm>