iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

AUTOMATED TIMETABLE GENERATOR SYSTEM

DR. K. SANTHI SREE
Professor, Department of Information Technology,
Jawaharlal Nehru Technological University Hyderabad,
drksanthisree@gmail.com

CHIRRA RAKESH REDDY
Post Graduate Student, M. Tech (SE) Department of Information Technology,
Jawaharlal Nehru Technological University, Hyderabad,
chrakeshreddy56@gmail.com

ABSTRACT
Designing academic timetables manually is a complex and time-consuming process, often leading to scheduling conflicts and inefficient use of resources. To address these issues, this project introduces an Automated Timetable Generator System, a web-based solution aimed at simplifying and optimizing the scheduling process for educational institutions. The system leverages constraint-based scheduling algorithms to ensure that faculty, classrooms, and course timings are efficiently allocated without overlaps. With a user-friendly interface, it allows administrators to define rules and constraints, then automatically generate accurate and conflict-free schedules. The application is developed using Java Servlets, JSP**,** and MySQL, offering both scalability and performance. Key features include dynamic timetable adjustments**,** conflict resolution**,** and export capabilities in PDF and Excel formats. The solution improves efficiency, reduces administrative burden, and enhances scheduling accuracy.

Keywords:
Automated Timetabling, Web-Based Scheduling, Constraint-Based Algorithms, Educational Systems, Conflict-Free Scheduling, Java Servlets, Timetable Optimization

INTRODUCTION

In educational institutions, the task of generating class schedules involves numerous constraints. It requires careful coordination of multiple variables such as course timings, faculty availability, classroom capacity, and institutional constraints. As institutions grow in size and complexity, manual scheduling becomes increasingly inefficient, leading to overlapping classes, double-booked faculty, and underutilized resources. These challenges not only increase administrative workload but can also negatively impact academic performance and faculty satisfaction.

To address these issues, this project presents an **Automated Timetable Generator System** designed to simplify and optimize the scheduling process. The system uses a **constraint-based algorithm** to allocate resources intelligently, ensuring that all predefined rules are met and conflicts are avoided. Built using **Java Servlets, JSP**, and **MySQL**, the system offers a web-based interface for ease of access and usability. It supports dynamic input of data and constraints, generates conflict-free schedules, and provides export options in PDF and Excel formats. By automating a traditionally manual task, the system not only improves scheduling accuracy but also enhances operational efficiency across the institution. This research aims to demonstrate the system's effectiveness in real-world academic environments and its potential for broader adoption.

RELATED WORK

The various techniques have been proposed to improve the process of academic scheduling:

- **"System of automated timetabling."**[1] implements a scheduling model based on graph coloring techniques to detect and resolve conflicts in course assignments.
- **"Automating Timetable Generation with Conflict Resolution Algorithms."**[2] introduced a web-based timetable system that supports real-time adjustments and utilizes conflict resolution algorithms to maintain schedule consistency.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

- **"Automating Time Table Planning Process."**[3] used participatory research methods to involve stakeholders in designing scheduling constraints, improving the practical relevance of the system.
- **"Automated Time-Tabling System for Tertiary Institutions."**[4] developed a rule-based scheduling engine that applies backtracking methods to reassign courses when conflicts occur.
- **"Timetable organizer for screenings."**[5] designed a semi-automated planner originally for research screening that can also be adapted for educational scheduling.
- **"Automated timetable generation."**[6] applied genetic algorithms to generate optimized academic schedules, reducing conflict rates significantly.
- **"Automated Timetable Generator."**[7] created a scheduling tool that prioritizes faculty availability and course importance when generating timetables.
- **"Automated lecture timetabling system."**[8] implemented heuristic methods in a scheduling system tailored to university lecture planning.
- **"Smart Timetable Scheduler."**[9] proposed an intelligent scheduling system integrated with institutional platforms for real-time updates.
- **"Automated timetabling using object-oriented scheduler."**[10] explored object-oriented scheduling frameworks that allow modular development and easy scalability.
- **"Automated system for university timetabling."**[11] conducted a case study in Algeria, demonstrating how automated systems can be applied at scale in university settings.

These studies highlight the diversity of approaches to academic scheduling and underscore the growing importance of automation in educational administration.

SYSTEM ARCHITECTURE

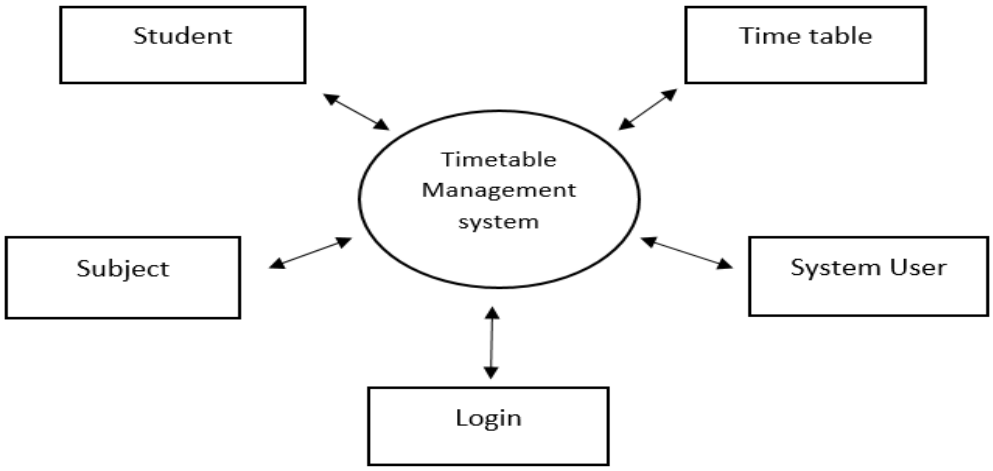


Fig 1: System Architecture

OBJECTIVES

The primary aim of this project is to develop an efficient and intelligent system that can automate the creation of academic timetables while minimizing errors and reducing manual workload. The system is designed to meet the diverse scheduling needs of educational institutions by considering various constraints such as faculty availability, room capacity, and course requirements. The following objectives guide the development and implementation of the system:

1. **Automate the creation of class schedules:** The system is designed to replace traditional, manual scheduling with an automated solution. It processes input data such as course lists, faculty availability, and classroom details to generate complete, organized timetables without human intervention.
2. **Resolve scheduling conflicts:** the key goals is to detect and rectify the conflicts automatically. This includes preventing overlapping lectures, avoiding double-booked instructors, and ensuring classrooms are not scheduled for multiple sessions at the same time.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

3. **Optimize the allocation of institutional resources:** The system ensures that all available resources, such as classrooms, laboratories, and faculty, are used efficiently. It prioritizes the effective distribution of these resources based on institutional requirements and constraints.
4. **Enable timetable export:** To support sharing and distribution, the generated timetables can be exported in commonly used formats like PDF and Excel. This makes it easy for administrators, faculty, and students to access and view schedules.
5. **Provide a user-friendly interface:** The web-based system includes an intuitive dashboard that allows administrators to input data, define constraints, and generate schedules easily, without needing technical expertise.

METHODOLOGY

To develop an effective and reliable timetable generator, a structured approach was followed that combines system design, algorithm development, and implementation using modern web technologies. The methodology focuses on collecting relevant academic data, applying constraint-based scheduling techniques, and building a user-friendly platform to generate and manage timetables efficiently. The process is divided into:

1. **System Architecture Design:** The application is built using a three-tier architecture comprising a web-based front end (HTML, CSS, JSP), a backend (Java Servlets), and a relational database (MySQL). This ensures scalability, modularity, and maintainability.
2. **Data Collection and Input:** Information related to faculty members, available classrooms, course details, and institutional constraints (like lecture durations, room capacities, etc.) is gathered and entered into the system database through the admin interface.
3. **Constraint Definition:** Constraints are defined to ensure realistic and feasible scheduling. Examples include no overlapping lectures for a faculty member, courses requiring specific room types (labs, lecture halls), and limited teaching hours per day.
4. **Scheduling Algorithm Implementation:** A constraint-based scheduling algorithm is employed. It uses graph-based conflict detection and heuristic optimization to prioritize important constraints. Backtracking is applied to resolve conflicts during the scheduling process.
5. **Timetable Generation:** The system automatically generates a timetable by processing all constraints and input data. It ensures that every session is assigned to an appropriate slot without conflict.
6. **Conflict Detection and Resolution:** If any conflicts are detected during or after timetable generation, the system automatically reassigns sessions using backtracking and constraint relaxation where necessary.
7. **Export and Distribution:** Once the final timetable is generated, it can be exported in PDF and Excel formats. This feature allows for easy sharing and printing by faculty, administrators, and students.

RESULTS

The system was tested using realistic academic data to evaluate its effectiveness in automating the timetable generation process. Testing focused on execution speed, conflict resolution accuracy, resource utilization, and user satisfaction. The results show that the system meets its objectives, handles complex scheduling scenarios, and delivers practical benefits for educational institutions.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

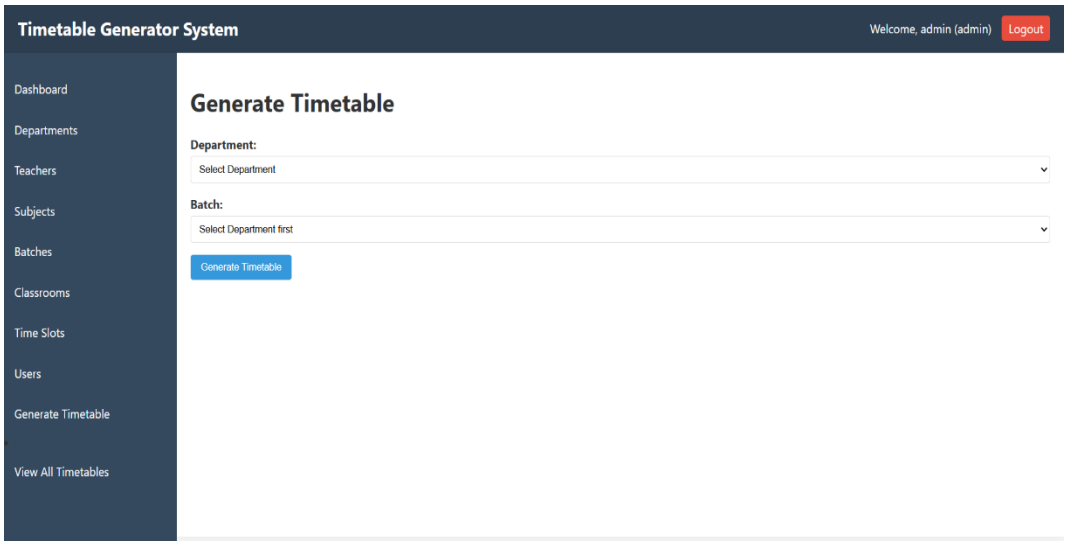
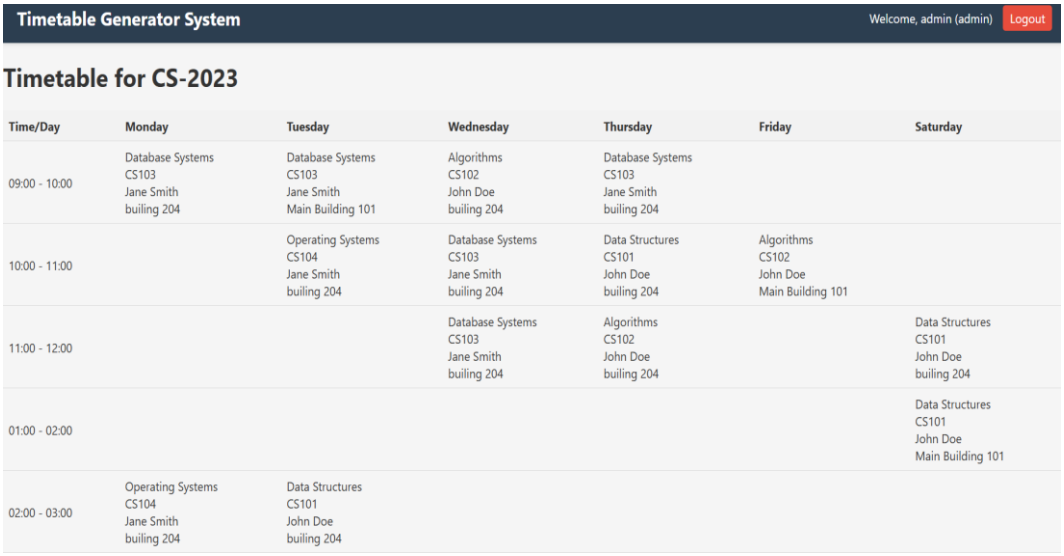


Fig 2: Details to be given for generating timetable



Time/Day	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
09:00 - 10:00	Database Systems CS103 Jane Smith builing 204	Database Systems CS103 Jane Smith Main Building 101	Algorithms CS102 John Doe builing 204	Database Systems CS103 Jane Smith builing 204		
10:00 - 11:00		Operating Systems CS104 Jane Smith builing 204	Database Systems CS103 Jane Smith builing 204	Data Structures CS101 John Doe builing 204	Algorithms CS102 John Doe Main Building 101	
11:00 - 12:00			Database Systems CS103 Jane Smith builing 204	Algorithms CS102 John Doe builing 204		Data Structures CS101 John Doe builing 204
01:00 - 02:00						Data Structures CS101 John Doe Main Building 101
02:00 - 03:00	Operating Systems CS104 Jane Smith builing 204	Data Structures CS101 John Doe builing 204				

Fig 2: Generated Timetable for the batch

CONCLUSION

The Automated Timetable Generator System developed in this project offers a practical and efficient solution to one of the most complex administrative challenges faced by educational institutions—class schedule creation. Traditional manual methods are not only time-consuming but also prone to frequent errors such as resource conflicts and poor utilization of faculty and classrooms. This system addresses these issues by leveraging a constraint-based scheduling algorithm integrated into a user-friendly, web-based platform.

Through comprehensive testing, the system demonstrated high accuracy, scalability, and speed in generating conflict-free timetables. It significantly reduced scheduling time from hours to minutes and eliminated common issues like overlapping classes and double-booked faculty. Additional features such as PDF and Excel exports, real-time adjustments, and constraint flexibility make it highly adaptable to varying institutional needs.

In conclusion, the proposed system contributes a reliable, scalable, and intelligent approach to academic scheduling, offering clear improvements in efficiency, accuracy, and overall management within educational environments.

REFERENCES

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

[1] Legierski, W., & Widawski, R. (2003). *System of automated timetabling*. In ITI 2003: Proceedings of the 25th International Conference on Information Technology Interfaces. IEEE.
[2] Lalitha, V., et al. (2023). *Automating Timetable Generation with Conflict Resolution Algorithms in Web-Based Systems*. IEEE ICCEBS.
[3] Joy, J., & Nambirajan, T. (2018). *Automating Time Table Planning Process: Participatory Action Research*. SCMS Journal of Indian Management, 15(4).
[4] Shomoye, M. A., & Mosaku, A. O. *Development and Implementation of an Automated Time-Tabling System for Tertiary Institution*.
[5] Goktug, A. N., Chai, S. C., & Chen, T. (2013). *A timetable organizer for screenings*. Journal of Biomolecular Screening, 18(8), 938–942.
[6] Devi, M. U., et al. (2024). *Automated timetable generation for academic institutions*. AIP Conference Proceedings, 3075(1).
[7] Sayed, E. S., et al. (2015). *Automated Timetable Generator*. International Journal for Innovative Research in Science & Technology, 1.
[8] Soyemi, J., Akinode, J. L., & Oloruntoba, S. A. (2017). *Automated lecture time-tabling system for tertiary institutions*. International Journal of Applied Information Systems, 12, 20–27.
[9] MB, A., et al. (2024). *Smart Timetable Scheduler and Management System for Campus*. Frontiers in Health Informatics, 13(3).
[10] Gunadhi, H., Anand, V. J., & Yong, Y. W. (1996). *Automated timetabling using an object-oriented scheduler*. Expert Systems with Applications, 10(2), 243–256.
[11] Betaouaf, T. H., Boukli-Hacene, R., & Cherier, M. A. (2021). *Application of automated system for university course timetable scheduling: An Algerian case study*. International Journal of Knowledge Engineering and Data Mining, 7(1–2), 86–112.