RESEARCH ARTICLE OPEN ACCESS

AUTOSCHED:AI-POWERED EXAM TIMETABLE OTIMPIZER

[1]R. Arunadevi [2]S. Bhuvaneshwari [3]M. Priyadharshini

[1] Assistant Professor, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science,

Thanjavur, Tamil Nadu – 613006, India

[2] UG Student , Information Technology, Parisutham Institute of Technology and Science,

Thanjavur,TamilNadu-613006,India

sweetybuvana226@gmail.com

[3] UG Student , Information Technology, Parisutham Institute of Technology and Science,

Thanjavur,TamilNadu-613006,India

priyadharshinimuthukrishnan07@gmail.com

Abstract

The manual generation of examination timetables in academic institutions is a complex and time-consuming task that is prone to human errors such as subject clashes, inefficient hall allocation, and improper scheduling.

This paper proposes Autosched, an automated exam timetable generation system based on constraint-based scheduling techniques. The system accepts inputs such as subject details, examination schedules, student strength, and hall capacity, and generates an optimized, conflict-free timetable.

The system incorporates modules for clash detection, intelligent hall allocation, and centralized data management. By automating the scheduling process, the proposed system significantly reduces manual effort. improves accuracy, and enhances overall efficiency in timetable management.

1. Introduction

Timetable scheduling is a critical administrative function in educational institutions. Traditionally, it is performed manually using spreadsheets or paper-based methods, which becomes increasingly inefficient as the number of students, courses, and examination sessions grows.

Manual scheduling often results in issues such as overlapping examinations, incorrect hall assignments, and difficulties in updating schedules dynamically. These challenges highlight the need for an automated and intelligent scheduling system.

The proposed system, Autosched, aims to streamline the timetable generation process using constraint-based techniques, ensuring

efficient resource utilization and error-free scheduling.

2. Existing System

In the current system, examination timetables are prepared manually or using basic tools such as Microsoft Excel. The administrator is responsible for scheduling exams, detecting conflicts, and assigning examination halls.

Limitations of Existing System

- ➢ High probability of human errors
- ➢ Time-consuming and labor-intensive process
- ➢ Lack of automatic clash detection
- ➢ Difficulty in modifying schedules
- ➢ Absence of centralized data storage

These limitations reduce efficiency and increase the chances of scheduling conflicts.

3. Proposed System

The proposed system, Autosched, automates the entire timetable generation process using constraint-based scheduling algorithms.

Input Parameters

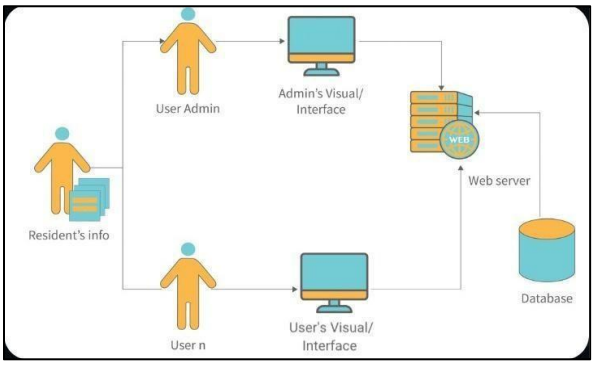
- ➢ Subject details
- ➢ Exam dates and time slots
- ➢ Student strength per subject
- ➢ Hall capacity and availability

Key Features

- ➢ Automatic detection of subject and arrear clashes
- ➢ Intelligent hall allocation based on capacity constraints
- ➢ Centralized database for data storage
- ➢ Instant generation of conflict-free timetables

The system ensures optimal resource utilization while minimizing manual intervention.

4. System Architecture



The architecture of the proposed system consists of multiple interconnected modules that ensure smooth data processing and efficient timetable generation.

5. Architecture Flow

1. Admin inputs data such as subjects, exam schedules, and hall details
2. The system processes the input data
3. Clash detection module identifies scheduling conflicts
4. Hall allocation module assigns halls based on constraints
5. Data is stored in a centralized database

6. Final timetable is generated and displayed

6. Module Description

1 Login Module

Provides secure authentication for administrators to access the system.

2 Subject and Exam Entry Module

Allows the admin to input subject details, exam dates, and time slots.

3 Hall Allocation Module

Allocates examination halls based on student strength and hall capacity using optimization logic.

4 Clash Detection Module

Identifies and prevents conflicts such as overlapping exams and duplicate scheduling.

5 Timetable Generation Module

Generates the final timetable after validating all constraints.

6 Database Management Module

Maintains all system data securely using a centralized database (e.g., MySQL).

7 Display and Output Module

Displays the generated timetable and supports export in formats such as PDF and Excel.

7. Methodology

The system uses a constraint-based scheduling algorithm to generate the timetable.

Constraints Considered

This modular architecture ensures scalability, flexibility, and efficient execution.

- ➔ No student should have two exams at the same time
- ➔ Hall capacity must not be exceeded
- ➔ Exams must be scheduled within available time slots
- ➔ Arrear and regular subjects should not clash

8. Algorithm Approach

- ➔ Input data collection
- ➔ Constraint validation
- ➔ Conflict detection
- ➔ Iterative schedule generation
- ➔ Optimization for hall allocation

This approach ensures that the generated timetable is both feasible and optimized.

9. Future Enhancements

- ➢ Integration of Artificial Intelligence for predictive scheduling
- ➢ Email and SMS notifications for students and staff
- ➢ Role-based access control for multiple users
- ➢ Cloud-based deployment for remote access
- ➢ Advanced optimization techniques for large-scale scheduling

10. Conclusion

The Autosched system provides an effective solution for automating exam timetable generation in academic institutions. By reducing manual effort and minimizing errors, the system ensures accurate and conflict-free scheduling.

The implementation of constraint-based techniques enhances efficiency, scalability, and reliability, making the system highly suitable for modern educational environments.

11. References

[1] E. K. Burke and S. Petrovic, "Recent research directions in automated timetabling," European Journal of Operational Research, vol. 140, no. 2, pp. 266–280, 2002.

[2] M. W. Carter and G. Laporte, "Recent developments in practical examination timetabling," Proceedings of the International Conference on the Practice and Theory of Automated Timetabling, 1996.

[3] A. Schaerf, "A survey of automated timetabling," Artificial Intelligence Review, vol. 13, no. 2, pp. 87–127, 1999.

[4] E. Burke, J. Kingston, and D. de Werra, "Applications to timetabling," in Handbook of Graph Theory, CRC Press, 2004.

[5] S. Abdullah, E. Burke, and B. McCollum, "A hybrid evolutionary approach to the university course timetabling problem," IEEE Transactions on Evolutionary Computation, vol. 16, no. 3, pp. 368–386, 2012.

[6] K. Deb, "Optimization for Engineering Design: Algorithms and Examples," Prentice Hall, 2012.

[7] F. Glover and M. Laguna, "Tabu Search," Kluwer Academic Publishers, 1997.

[8] H. Babaei, J. Karimpour, and A. Hadidi, "A survey of approaches for university course timetabling problem," Computers & Industrial Engineering, vol. 86, pp. 43–59, 2015.