International Journal of Electrical, Electronics and Computer Systems
ISSN: 2347-2820
Volume 15 Issue 01 (2026) | Pages: 168-176

Result Paper on AI-Driven Conflict-Free Academic Timetabling System

P. U. Chavan[1], Swapnil A. Vyawahare[2], Prathamesh R. Pawar[3], Rohan M. Tingare[4], Harshal N. Mhasawade[5]

[1,2,3,4,5]Dept. of Computer Engineering, SBPCOE,Indapur
[1]*poojabhoite17@gmail.com*, [2]*swapnilvyawahare55@gmail.com* [3]*prathmeshp744@gmail.com*, [4]*rohantingare2@gmail.com*, [5]*harshalmhasawade@gmail.com*[5]

Peer Review Information

***Type**: Article*
***Received**: 24 March 2026*
***Revised**: 09 April 2026*
***Accepted**: 27 May 2026*
***Published**: 06 June 2026*

Abstract

Academic timetabling is the problem of assigning courses, instructors, rooms, and student groups to limited time slots while respecting hard constraints (no clashes, room capacity, teacher availability) and optimizing soft preferences (time spreads, load balancing). Manual methods are tedious, error-prone, and slow to react to change.

This project implements a web-based, role-secured timetable system that automates schedule generation through an Al solver service. Core capabilities include authenticated data entry and governance, building a solver input from authoritative master data, computing feasible schedules, presenting conflicts for review, and publishing official outputs (including PDF exports). The system emphasizes accuracy, scalability, security, and operational transparency through auditability and clear state transitions.

The paper concludes by discussing future research directions, including real-time rescheduling, user in-the-loop systems, and explainable AI for transparent decision-making. By consolidating recent advancements and open challenges, this survey provides a strong foundation for future studies and assists educational institutions in adopting intelligent, conflict-free, and scalable timetabling solutions.

Keywords: Timetabling; Genetic Algorithm; Reinforcement Learning; Optimization; Scheduling; Educational Technology; Automated Timetable Generation; AI-Based Scheduling.

How to Cite This Article

Chavan, P. U., Vyawahare, S. A., Pawar, P. R., Tingare, R. M., & Mhasawade, H. N. (2026). Result paper on AI-driven conflict-free academic timetabling system. *International Journal of Electrical, Electronics and Computer Systems*, 15(1), 168–176.

Introduction

Scheduling is one of the most critical and complex administrative tasks in educational institutions. Universities and colleges manage multiple departments, semesters, faculty members, classrooms, laboratories, and academic subjects. The process of assigning lectures to appropriate time slots while satisfying various constraints is highly challenging and time-consuming. Traditionally, timetables are prepared manually or using spreadsheet-based tools. However, as the number of constraints increases, manual scheduling becomes inefficient and error-prone. Conflicts such as overlapping faculty assignments, classroom clashes, and uneven subject distribution frequently occur, requiring repeated modifications. The AI-Driven Conflict-Free Academic Timetabling System is designed to automate and optimize the scheduling process using a constraint-based optimization approach. The system integrates modern web technologies such as React.js for frontend development, Node.js and Express.js for backend services, MySQL for database management, and a Python-based FastAPI optimization engine for intelligent timetable generation. The system provides role-based access to different stakeholders, including Administrator, Head of Department (HOD), Faculty Members, and Students. By integrating automated conflict detection and intelligent scheduling, the system ensures efficient resource allocation, reduced manual effort, and improved institutional productivity.

Literature Survey

This section presents a comparative literature survey of key research works in academic timetabling, focusing on Artificial Intelligence (AI) and metaheuristic approaches. The table summarizes methodologies, contributions, and limitations of more than fifteen significant studies.

Table 1. Comparative Analysis of Existing AI-Based Timetable Scheduling Approaches

Ref	Year	Technique(s) / Approach	Key Contribution	Limitations / Comments
Burke& Petrovic	2002	Survey / Meta-analysis	Broad categorization of timetabling problems, constraints, heuristics.	Focuses more on theory than implementation.
Ghaffar et al.	2025	Hybrid AI+ heuristics	Systematic review of hybrid AI techniques for exam timetabling.	Focuses on exams rather than course timetabling.
Farinola& Assogba	2025	GA + Explicit constraint model	Conflict-free timetables tested with real institutional data.	Requires customization per institution.
Khokale et al.	2025	Decision Tree, K-Means, Random Forest	Comparative study of modern AI methods for scheduling.	Conceptual, less experimental validation.
Saw et al.	2025	GA + ML (Adaptive Scheduler)	Introduced feedback-based adaptive scheduling.	Limited to small institutions.
Alonge et al.	2025	NSGA-II	Multi-objective optimization balancing conflicts and fairness.	Scalability to large datasets needs further study.
IJARCCE Survey	2025	AI+ Heuristic Scheduler	Proposed web-based timetable generator with verification.	Focuses on design more than optimization novelty.
Aghicha et al.	2024	Dual-Method AI + GA	AI for rule extraction combined with GA refinement.	Limited real-world experiments.

Sharma & Joshi	2021	Reinforcement Learning	Adaptivescheduling using RL models.	Convergence and complexity issues.
Lee & Das	2022	Deep Learning Optimization	Neural networks applied for schedule optimization.	Requires large datasets and compute.
Patel & Mehta	2020	Genetic Algorithm	Demonstrated GA efficiency in academic timetabling.	Limited support for soft constraints.

Limitations Of Existing Work

Despite progress, prior research has limitations:

- Difficulty handling multi-format institutional data.
- Limited predictive analytics for adapting to policy or structural changes.
- Weak support for collaborative, role-based scheduling.
- Minimal integration with cloud-based environments for remote learning.
- Scalability issues when applied to large datasets.
- Lack of strong data privacy and security mechanisms.

Problem Statement

The problem addressed in this project is the design and development of a web-based automated system capable of generating optimized academic timetables while satisfying institutional constraints. The system must handle multiple parameters such as courses, faculty availability, classroom capacity, and time slots, and generate a valid schedule without conflicts.

The system should ensure that no faculty member is assigned multiple lectures at the same time and no classroom is allocated to more than one lecture simultaneously. It must also ensure that each subject satisfies the required number of weekly lectures and that the timetable is generated efficiently within acceptable time.

Additionally, the system should provide secure and role-based access to users such as Admin, HOD, Faculty, and Students. The scheduling process must consider multiple constraints simultaneously and produce an optimized, conflict-free timetable as output.

Proposed System

The proposed system is an AI-Driven Conflict-Free Academic Timetabling System designed to automate and optimize the timetable generation process using intelligent algorithms and a centralized architecture.

The system begins with input data, which includes essential information such as courses, faculty details, and classroom availability. This data is provided by the administrator and stored in a centralized database for further processing.

The core of the system is the AI Optimization Engine, which uses techniques such as Genetic Algorithms and constraint-solving methods to generate an optimized timetable. The engine processes all constraints, including faculty availability, classroom allocation, and subject requirements, to ensure a feasible and efficient schedule.

A Conflict Detection and Validation module is integrated into the system to identify and resolve scheduling conflicts. This ensures that no faculty member or classroom is assigned to multiple lectures at the same time, maintaining a conflict-free timetable.

The system supports role-based access, allowing different users such as Admin, HOD, Faculty, and Students to interact with the system based on their roles. Each user has specific functionalities, such as managing data, approving schedules, or viewing timetables.

The Timetable Generation module produces an optimized schedule based on the processed data and constraints. The generated timetable is then stored in a centralized database (MySQL) for secure and efficient data management.

The system also provides authentication and authorization mechanisms to ensure secure access and data protection. Only authorized users can access or modify system data.

Finally, the system generates output and reports, allowing users to view timetables and export them in formats such as PDF and CSV for easy access and sharing.

Overall, the proposed system improves efficiency, reduces manual errors, ensures conflict-free scheduling, and provides a scalable and secure solution for academic timetable management.

Architecture

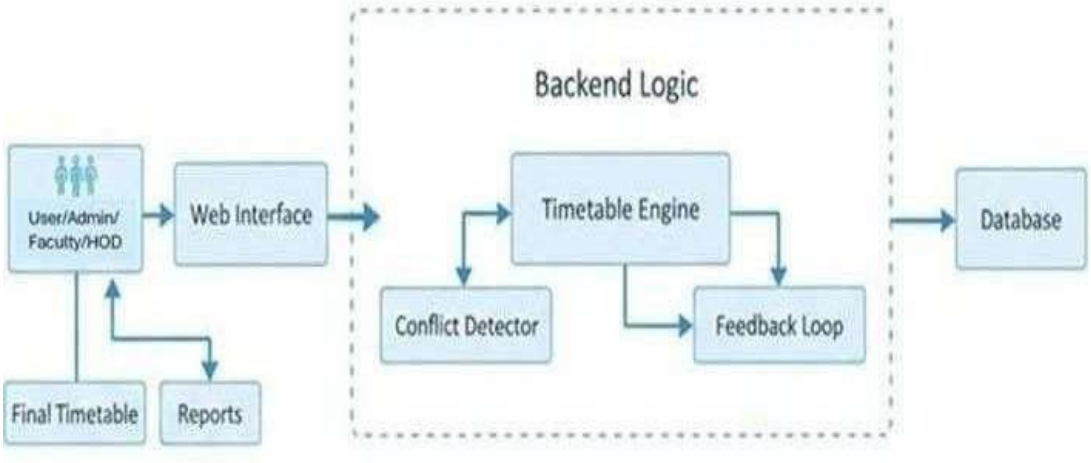

Fig. 1. *System Architecture*

Architecture Description

1. User Module

This module represents the users who interact with the system. The main users include:

- Administrator
- Head of Department (HOD)
- Faculty
- Students

The Admin manages the system by entering and updating data such as courses, faculty details, classrooms, and time slots. Faculty and students use the system to view their assigned timetable. This module ensures proper interaction between users and the system.

2. Web Interface Module

The Web Interface acts as the communication layer between users and the system. It provides a user-friendly interface where users can:

- Login securely
- Enter scheduling data
- Manage timetable information
- View generated timetables and reports

This interface sends user requests to the backend logic for processing and displays the final results to users.

3. Timetable Engine Module

The Timetable Engine is the core component of the system. It is responsible for generating the academic timetable based on the given input data and constraints.

Functions of this module include:

- Processing course, faculty, and classroom data
- Applying scheduling algorithms
- Generating optimized timetable schedules
- Ensuring that constraints are satisfied

The engine uses optimization techniques such as Genetic Algorithms and constraint-based scheduling methods to produce an efficient timetable.

4. Conflict Detection Module

This module checks the generated timetable for any scheduling conflicts. It ensures that:

- No faculty member is assigned multiple lectures at the same time
- No classroom is allocated to more than one lecture simultaneously
- Subjects are scheduled according to required lecture hours

If conflicts are detected, the system identifies them and sends the information to the timetable engine for correction.

5. Feedback Loop Module

The Feedback Loop module continuously improves the timetable generation process. When conflicts or inefficiencies are detected, this module sends feedback to the timetable engine so that it can adjust the schedule.

This iterative process continues until the system generates a valid and conflict-free timetable.

6. Database Module

The Database module stores all system data in a centralized repository. It maintains information such as:

- User data
- Faculty details
- Course information
- Classroom details
- Generated timetables

The database ensures secure storage, efficient data retrieval, and proper management of scheduling information.

7. Output & Report Module

After successful timetable generation, the system produces the Final Timetable and Reports. Users can view their schedules and generate reports. The timetable can also be exported for official use.

This module provides the final output of the system to users.

Objective

The main objective of the AI-Driven Conflict-Free Academic Timetabling System is to develop an automated system for generating optimized and conflict-free academic timetables while satisfying all institutional constraints.

The specific objectives of the system are as follows:

- To automate the process of academic timetable generation
- To implement constraint-based optimization techniques for scheduling
- To prevent faculty and classroom conflicts
- To ensure that each subject meets the required number of lecture hours
- To provide secure authentication using role-based access (Admin, HOD, Faculty, Student)
- To reduce manual effort and time required for timetable creation
- To provide export functionality in formats such as PDF and CSV
- To ensure scalability and flexibility for future enhancements

Hardware And Software Requirements

Hardware Requirements

The system requires the following minimum hardware configuration for efficient performance:

- Processor: Intel i3 or higher
- RAM: Minimum 8 GB
- Storage: At least 10 GB free space
- System Type: Desktop or Laptop
- Internet Connectivity: Required for web-based access

Software Requirements :

The system is developed using modern technologies and requires the following software components:

- Frontend: React.js, TypeScript
- Backend: Node.js, Express.js
- Database: MySQL
- Optimization Engine: Python FastAPI
- Development Tools: Visual Studio Code
- Operating System: Windows or Linux
- Web Browser: Chrome, Edge, or any modern browser

Algorithm

The AI-Driven Conflict-Free Academic Timetabling System uses Artificial Intelligence-based optimization techniques to generate an efficient and conflict-free timetable. The following algorithms are used in the system:

1. Genetic Algorithm (GA)

Genetic Algorithm is used as the primary optimization technique for timetable generation. It works based on the concept of natural selection and evolution.

- Generates multiple timetable solutions (population)
- Evaluates each solution using a fitness function
- Applies operations like selection, crossover, and mutation
- Iteratively improves solutions to minimize conflicts
- This algorithm helps in finding an optimal or near-optimal timetable efficiently.

2. Constraint Satisfaction Technique (CSP)

Constraint Satisfaction is used to ensure that all hard constraints are satisfied during timetable generation.

- No faculty overlap
- No classroom conflict
- Proper subject allocation
- Valid time slot assignment
- This ensures that the generated timetable is feasible and valid.

3. Reinforcement Learning (Concept Used)

Reinforcement Learning concepts are used to improve decision-making in scheduling.

- Learns from previous timetable generation results
- Improves optimization over iterations
- Enhances performance for complex scheduling scenarios

4. Optimization Techniques

Additional optimization techniques are applied to:

- Balance faculty workload
- Distribute lectures efficiently
- Reduce idle time and gaps

Output

The AI-Driven Conflict-Free Academic Timetabling System generates outputs in the form of interactive and role-based timetable dashboards. The system provides different views for Faculty, Head of Department (HOD), and Students, ensuring that each user can access relevant scheduling information efficiently.

The Faculty Dashboard displays individual teaching schedules, including subject allocation, time slots, classroom details, and weekly distribution of lectures. It also provides summary statistics such as total classes, subjects, departments, and divisions.

The HOD Dashboard presents a department-wise timetable, allowing administrators to view and manage schedules across multiple semesters and divisions. It includes features such as timetable generation, grid view, and export options for better control and monitoring.

The Student Dashboard provides a clear and structured class schedule, enabling students to view their daily and weekly lectures along with subject names, room numbers, and timings.

All outputs are designed to be user-friendly, visually organized, and free from scheduling conflicts. The system also supports export functionality in formats such as PDF, CSV, and iCal, making the timetable easy to share and use in academic environments.

Overview

The AI-Driven Conflict-Free Academic Timetabling System is a web-based application developed to automate and optimize the process of timetable generation in educational institutions. The system is designed to handle complex scheduling requirements by considering multiple constraints such as faculty availability, classroom allocation, subject requirements, and time slots.

The system integrates modern technologies including React.js for frontend development, Node.js and Express.js for backend processing, MySQL for database management, and a Python-based FastAPI optimization engine for intelligent timetable generation. This combination

of technologies ensures efficient processing, scalability, and user-friendly interaction.

The application provides role-based access to different users such as Administrator, Head of Department (HOD), Faculty, and Students. Each user can perform specific tasks such as managing data, approving timetables, and viewing schedules. The system also includes features like conflict detection, real-time timetable access, and export functionality.

Overall, the system provides a centralized and efficient solution for academic timetable management, reducing manual effort, minimizing errors, and improving overall productivity in educational institutions.

Key Features

The AI-Driven Conflict-Free Academic Timetabling System provides several important features that improve efficiency, accuracy, and usability in timetable management.

- Automated Timetable Generation: The system automatically generates academic timetables using AI-based optimization techniques, reducing manual effort.
- Conflict-Free Scheduling: Ensures no faculty or classroom conflicts by applying constraint-based scheduling methods.
- Role-Based Access Control: Provides secure access for different users such as Admin, HOD, Faculty, and Students with specific functionalities.
- AI Optimization Engine: Uses techniques such as Genetic Algorithm and Reinforcement Learning to generate optimized timetables.
- Real-Time Timetable Access: Allows users to view updated timetables instantly through the web interface.
- Centralized Data Management: Stores all data such as faculty, subjects, classrooms, and schedules in a MySQL database.
- Conflict Detection and Validation: Identifies and resolves scheduling conflicts during timetable generation.
- Export Functionality: Enables export of timetables in formats such as PDF and CSV.
- Scalability and Flexibility: The system can handle multiple departments and can be extended for larger institutions.
- User-Friendly Interface: Provides an intuitive web interface for easy interaction and management.

Example Output

The example outputs of the system demonstrate the practical implementation of the AI-Driven Conflict-Free Academic Timetabling System across different user roles. Each interface is designed to provide clear, structured, and conflict-free scheduling information.

Example 1: Faculty Timetable

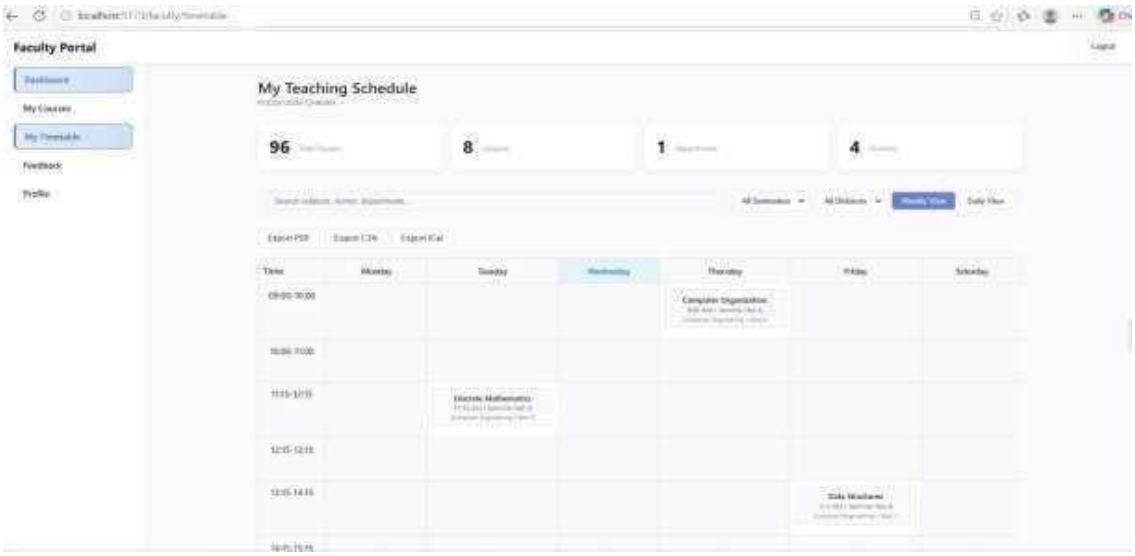

Fig. 2. *Faculty Timetable Dashboard*

The faculty timetable interface displays the weekly teaching schedule assigned to a specific faculty member. It includes detailed information such as subject names, lecture timings, classroom locations, and semester details. The dashboard also provides summary statistics like total number of classes, subjects, departments, and divisions, helping faculty members analyze their workload efficiently. Additional features such as search, filtering, and export options (PDF, CSV, iCal) enhance usability and accessibility.

Example 2: HOD Department Timetable

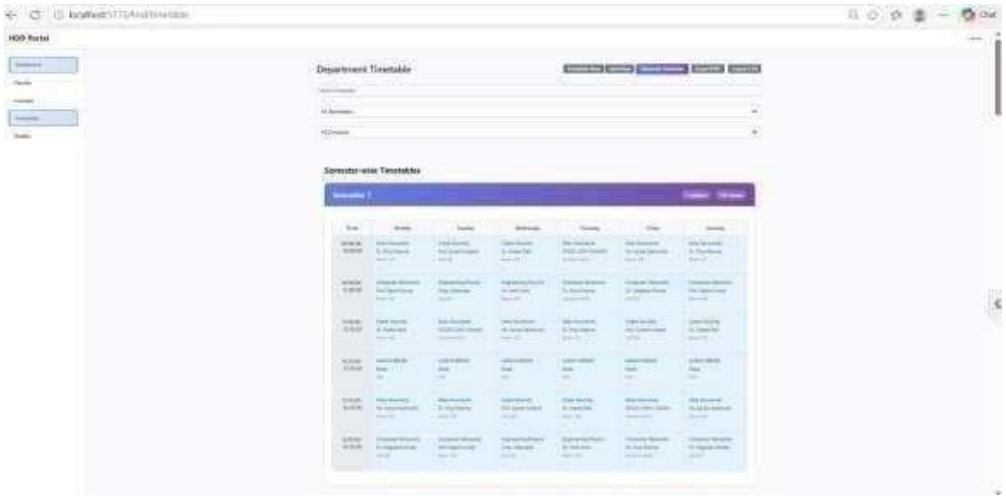

Fig. 3. *Department Timetable View for HOD*

The HOD interface presents a comprehensive department-wise timetable covering multiple subjects, faculty members, and classrooms. It allows the Head of Department to view schedules for different semesters and divisions in a single dashboard. The interface includes options such as timetable generation, grid view, and export functionality, enabling efficient management and monitoring of academic schedules. This view ensures that all constraints are satisfied and helps in identifying and resolving scheduling conflicts at the departmental level.

Example 3: Student Timetable

The student timetable interface provides a clear and structured view of the class schedule for students. It includes subject names, lecture timings, room numbers, and break periods such as lunch. The timetable is displayed in a weekly format, making it easy for students to track their daily academic activities. The system ensures that there are no overlapping lectures, providing a smooth and organized learning schedule. Filtering and export features further improve accessibility and usability for students.

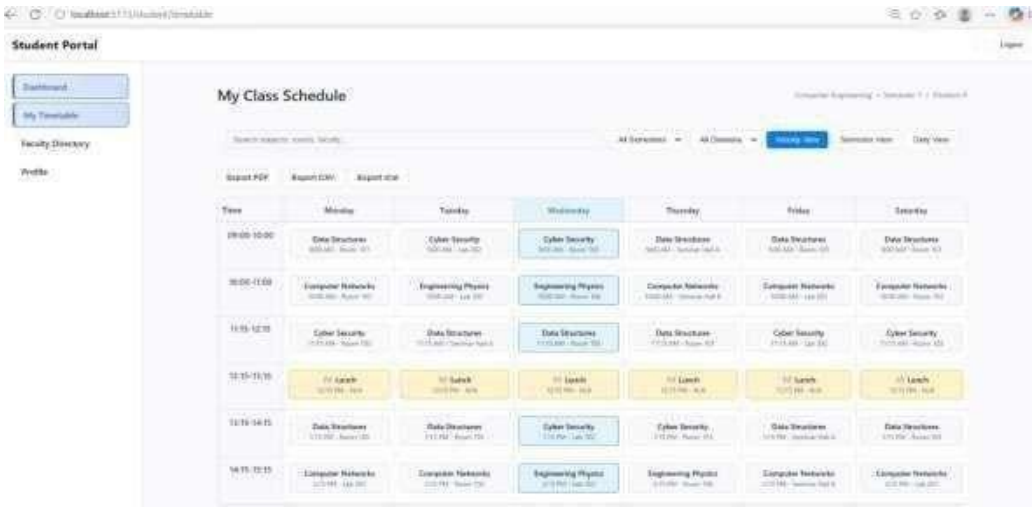

Fig. 4. *Student Weekly Timetable View*

These example outputs clearly demonstrate the system's ability to generate conflict-free, well-organized, and role-based timetables, improving efficiency and usability for faculty, administrators, and students

Result Discussion

The AI-Driven Conflict-Free Academic Timetabling System was successfully designed, implemented, and tested to evaluate its effectiveness in solving the academic scheduling problem. The system was tested using real-time input data such as courses, faculty availability, classroom details, and time slots. The results demonstrate that the system is capable of generating an optimized and conflict-free timetable efficiently. The system effectively eliminates common problems associated with manual timetable generation, such as

faculty conflicts, classroom clashes, and overlapping lectures. By applying AI-based optimization techniques such as Genetic Algorithm and constraint satisfaction, the system ensures that all hard constraints are strictly satisfied. Additionally, soft constraints such as balanced workload distribution and proper time allocation are also optimized.

The timetable generation process is completed within an acceptable response time, even when multiple constraints are applied. The use of a feedback loop mechanism further enhances the optimization process by iteratively improving the generated timetable until a feasible and efficient solution is obtained. The system provides role-based access for different users including Admin, HOD, Faculty, and Students. The Admin can manage all system data and generate timetables, while the HOD can review and approve them. Faculty members can view their assigned lectures, and students can access their schedules in real-time. This ensures transparency and usability of the system.

The system also supports features such as conflict detection, validation, and report generation. Users can view the final timetable and export it in formats such as PDF and CSV. This improves accessibility and makes the system suitable for practical use in educational institutions Overall, the results indicate that the proposed system significantly reduces manual effort, minimizes errors, improves scheduling efficiency, and provides a scalable and reliable solution for academic timetable management. The system can be further enhanced to handle larger datasets and additional constraints in future developments.

Conclusion

The AI-Driven Conflict-Free Academic Timetabling System provides an intelligent and automated solution for generating academic schedules without conflicts. The system efficiently manages core academic entities such as courses, faculty, rooms, time slots, and constraints.

By integrating Artificial Intelligence techniques such as Genetic Algorithm (GA) and optimization strategies, the system ensures that hard constraints (e.g., room clashes, faculty overlap, and student conflicts) are strictly satisfied while also considering soft constraints such as preferred time slots and workload balance.

The system reduces manual effort, minimizes scheduling errors, and saves adminis- trative time. Secure authentication, role-based access control, and audit tracking improve system reliability and accountability. Overall, the project enhances institutional efficiency, ensures fair workload distribu- tion, and provides a scalable and adaptable solution for modern academic scheduling needs.

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. stern.nyu.edu
2. A. Ghaffar et al., “Hybridization and Artificial Intelligence in Optimizing University Exam Timetabling,” *Review*, 2025. bera-journals.onlinelibrary.wiley.com
3. L. A. Farinola and M. B. M. Assogba, “Explicit Artificial Intelligence Timetable Generator for Colleges and Universities,” *Open Journal of Applied Sciences*, vol. 15, 2025. scirp.org
4. S. R. Khokale, A. Jadhav, R. Chavan, S. Wani, P. Iwanate, “A Survey Paper on Timetable Generator Using AI Methods,” *IRJAEH*, vol. 3, no. 03, Mar. 2025. irjaeh.com
5. S. K. Saw, S. Lawrence, S. Karunakara, N. Hrudhika Komal, V. Kumari G, “Adaptive
6. Scheduler: AI Optimization of Academic Timetable,” IJSDR, vol. 10, no. 4, 2025. ijsdr.org
7. O. Alonge, W. Sakpere, E. Adediran, “An Automated Timetable Scheduler Using NSGA-II,”
8. Advance Journal of Science, Engineering and Technology, vol. 10, no. 4, 2025. aspjournals.org
9. “Automatic Time Table Generator,” *IJARCCE*, vol. 14, no. 5, May 2025. Peer-reviewed Journal
10. N. J. Aghicha, S. Shaikh, A. R. Deshmane, K. Y. Suralkar, A. R. Koshti, “Review on Dual-
11. Method Timetable Generator Using AI and Genetic Algorithm,” *IRJMETS*, 2024. irjmets.com
12. “A Survey on Schedule Academic Time Table using AI and ML,” *IJRPR*, 2022. ijrpr.com
13. “AI Powered Automatic Timetable Generator,” *IJCRT*, 2025. ijcrt.org
14. A. Wren, “Scheduling, Timetabling and Rostering – A Special Relationship?,” in *Lecture Notes in Computer Science*, vol. 1153, 1996.
15. A. Patel and D. Mehta, “Genetic Algorithm based approach for timetable generation in academic institutions,” *International Journal of Computer Applications*, vol. 176, no. 30, 2020.
16. V. Sharma and R. Joshi, “Reinforcement Learning Techniques for Adaptive Academic Scheduling,” *International Journal of Artificial Intelligence in Education*, 2021.
17. J. Lee and A. Das, “Deep Learning-based Optimization for Class Schedule Generation,” *Procedia Computer Science*, vol. 198, 2022.
18. (Classic) “Constraint Satisfaction Problems and their Applications in Scheduling,” *Journal of Computational Intelligence*, 2019.