

AI-Driven Dynamic Scheduling and Real-Time Notification System for Staff Optimization and Conflict Resolution

Monesh S.¹, Sheninth JR.², Sriram S.³, Abirami⁴

¹⁻³Student, ⁴Assistant Professor, Department of Artificial Intelligence and Data Science, Velammal Institute of Technology, Chennai, India.

E-mail: ¹moneshshanmugam@gmail.com, ²sheninthjr@gmail.com, ³Sriramsri687@gmail.com, ⁴abirami@velammalitech.edu.in

Abstract

In institutions, developing a timetable is a challenging and problematic process, particularly when creating it manually. When managing multiple departments, some common challenges arise like overlapping schedules, unequal staff allocation for classes and insufficient classroom usage. This proposed work presents an AI-driven solution to automate and optimize timetable scheduling in colleges by using intelligent algorithms. The research involves proposed methodologies that include Constraint Satisfaction Problem (CSP) solutions and Genetic Algorithms, allowing the system to allocate courses, faculty members, classrooms and time slots without any conflicts. This model provides an equal and optimal distribution by considering several limitations like faculty availability, workload balance and institution regulations. The proposed research uses the Genetic Algorithm that represents timetables as “chromosomes” evaluated using a fitness function based on moderate or severe limitations producing solutions through crossover and mutation. The system increases the accuracy of scheduling by reducing human effort. It also handles faculty or time alterations in real-time. This smart automation method improves scalability across institutions and creates a path for a smart educational management system. The proposed system provides a significant shift from manual techniques to data-driven scheduling, increasing educational planning and utilizing resources effectively.

Keywords: Timetable Generation, Genetic Algorithm, CSP, Automation, Scheduling System, Conflict Resolution, Faculty Allocation.

1. Introduction

Academic timetable scheduling is an efficient approach for educational institutions with a major impact on teaching quality, resource utilization, and satisfaction among learners. Many educational institutions, particularly those with large student populations and multiple departments, generate class schedules manually or through semi-automated processes. These methods are time-consuming, inaccurate and difficult to manage due to several limitations such as staff availability, classroom accessibility, and course requirements. Traditional scheduling methods become less effective as institutions develop and the choice of courses for students expands. The issues involve avoiding schedule conflicts across different departments and divisions, dividing staff workloads equally and optimising the overall use of classroom resources. Additionally, sudden changes, such as staff absences and changes in classroom assignments, complicate the manual scheduling process.

The implementation of Artificial Intelligence (AI) and optimization algorithms for generating schedules serves as a solution to these increasing problems. AI-driven systems can handle complicated rule sets and provide conflict-free schedules that satisfy institutional regulations by using Genetic Algorithms (GA) and Constraint Satisfaction Problem (CSP) methods. This system reduces human interaction and scheduling errors by enhancing flexibility and adaptability in real-time. The proposed work mainly focuses on developing an efficient, automatic scheduling system that generates timetables without any conflicts for different departments and sections. The system ensures high accuracy, better utilisation of faculty and classrooms, and time savings through the use of optimization methods and constraint-handling algorithms. The main objective is to provide a smart, scalable solution that accommodates changes in requirements for modern educational institutions surpassing traditional techniques.

2. Literature Survey

Schedule management presents a major challenge for educational institutions, particularly regarding staff availability, class schedules, and the frequent changes within institutions. Various research studies have been conducted to address these issues with computational and AI-based solutions. This literature analysis will identify the gaps and resolve them through the proposed work, highlighting the necessary contributions.

The research work [1] creates an academic schedule without any conflicts by combining the Genetic Algorithm and constraint satisfaction strategies. Different kinds of moderate and severe limitations, such as class size, faculty availability, and space constraints, can be handled flexibly using this method. However, this research lacks the ability to reschedule or provide essential updates when unexpected problems arise in real-time.

Heuristic and graph-coloring methods are used in this research [2] to generate class schedules. This model is not effective in rapidly changing situations because it fails to handle sudden changes, such as faculty substitutions or combining sections in real-time, even though it performs well with stable datasets.

A smart system that improves over time by interacting with its environment was presented in this work [3]. It adjusts schedules based on faculty availability, student performance, and institutional goals. In real-time implementation, due to unpredictable benefits, it faces challenges without high computational support since the reinforcement learning models frequently experience accessibility issues and long training durations.

A comprehensive review of constraint satisfaction programming (CSP) for modeling the complicated rules and requirements in this model is provided [4]. This system effectively handles faculty requirements, course necessities, and classroom limitations. However, efficiency may become an issue when organizations or institutions use large datasets and require regular schedule modifications.

The study [5] evaluates the use of Support Vector Machine (SVM) and decision trees for predicting optimal periods for courses. This method shows better results in recognizing patterns in historical data, but it is limited when encountering new schedules that did not exist in the training data, which reduces its ability to generalize and adapt.

This research [6] developed a method that allows separate users, such as classroom handlers, faculty, and course assigners, to modify schedules independently. The communication costs and challenges of managing these user actions are unsustainable for large-scale real-time systems, leading to decentralized models and conflict resolution issues.

The notification alert technology included in this research work [7] enables immediate responses from faculty and students regarding schedule modifications. However, this system's user acceptance of the communication enhancement is limited. The drawback of this study is

that the scheduling software lacks AI features to predict or avoid upcoming conflicts in the future, remaining static and rigid.

This research [8] addresses how schedules and user preferences can be stored in cloud servers, allowing multiple device access and a centralized data tracking process. However, this model does not include predictive analysis, and optimized models are not employed to improve scheduling accuracy.

The proposed work addresses these gaps by developing a real-time flexible scheduling system enabled by an AI model. The proposed method includes a cloud-connected relational database for centralized data processing, a real-time notification system for end-user involvement, and machine learning models such as Random Forest, Naïve Bayes, and Decision Tree for conflict identification and resolution. The system provides accurate, effective, and flexible scheduling solutions that are suitable for modern educational institutions by combining a smart AI-based backend with an accessible frontend.

3. Design

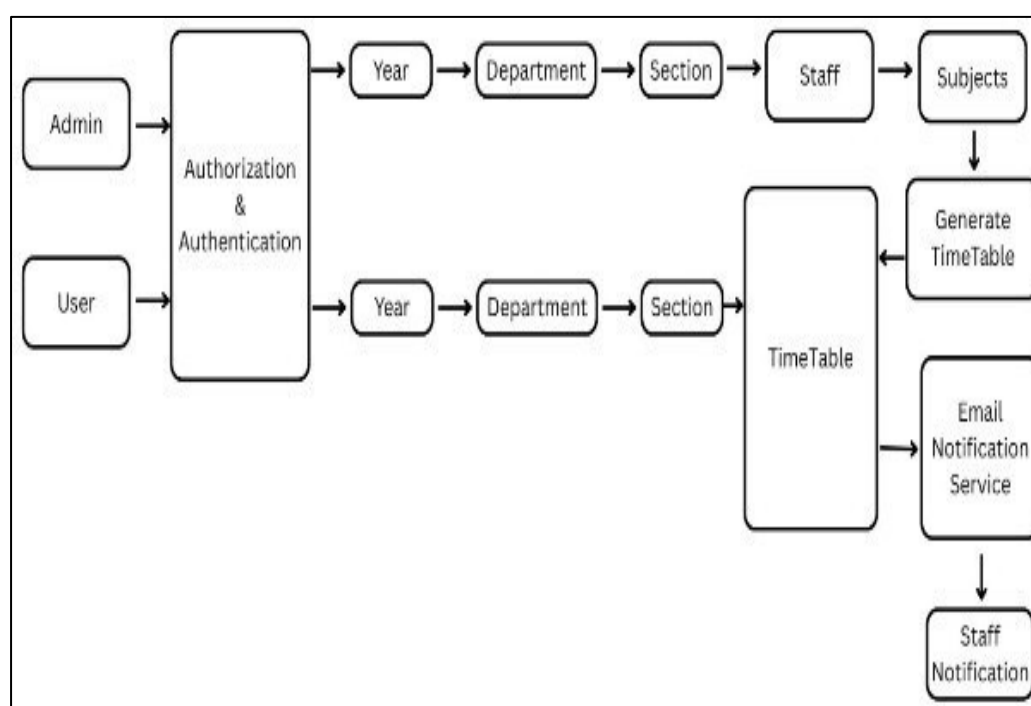


Figure 1. Block Diagram

The basic input structure (refer figure 1) of the AI-driven scheduling management system for educational institutions is a key component that controls the collection, preparation, validation and organization of all academic and faculty-based data. The first

scheduling inputs systematically collect this process by suggesting the teaching periods, course lists, availability of faculty, department-wise course allocation, grouping students based on sections, classroom availability and institutions limitation like increasing workloads for faculty and lab time regulations.

The GA is mainly used as an optimization method in the timetable scheduling system. Each users' schedule is associated with faculty, classroom and time slot assignments like a chromosome. This technique improves the results repeatedly by using a fitness function that evaluates hard restrictions like faculty availability, course periods, and classroom capacity without overlaps, as well as soft restrictions like faculty preferences and balancing workloads, which enhance the quality of the schedules. The crossover method creates new schedules by transferring sections of two existing ones. Mutation is used to maintain variety and prevent local optima; specific time slots for the staff are constantly distributed. In general, the sources of these inputs are the department coordinators, administrative records and the educational scheduling databases. The initial inputs require detailed preprocessing steps to standardize formats, rectify errors and remove duplicate schedules in order to ensure efficient processing. The course durations are converted into structured timeslot arrays, and the classroom facilities are cross-validated against the section size to avoid overloading the students. Validation requirements established within the system are used at this step to detect faculty entries, such as course allocation conflicts, lack of classroom assignments, and overlapping faculty positions.

The system will constantly modify to enhance scheduling robustness by modelling edge conditions like mass faculty unavailability or unexpected classroom cancellations during the input phase using augmentation-like techniques. The purpose of these AI-based test scenarios is to increase the scheduling system's adaptability and prepare it to handle unexpected situations in real-time. Some hard limitations like section groupings, course duration and faculty availability are examples that cannot be addressed but must be satisfied. Soft limitations, such as workload balance, faculty time preferences and student preference are optimized but can be reduced in case of conflict detection. This proposed work modifies the consequence levels to improve outcomes such as student comfort and faculty satisfaction without violating any regulations.

The data is organized and introduced into the scheduling model, followed by validation and preprocessing methods. An AI-based optimization algorithm uses rule-based

heuristics, constraint satisfaction methods, or evolutionary algorithms to dynamically assign periods directly communicated in the input layer. This model develops practical and optimal timetable solutions by considering both hard and soft limitations. Additionally, the input system has a continuous refining process that constantly updates and improves the schedule in response to real-time modifications such as the addition of a new topic or shifts in faculty availability. Throughout the academic year, the scheduling system will continue to be flexible and adaptable due to its dynamic adjustment mechanism.

The input system design ensures that all relevant data for the creation of the schedule is collected, extracted, verified and structured in an efficient way that allows the scheduling system to provide outputs without conflicts and with dependability. The method provides final academic schedules that are well-structures and conflict-resolved at the output design phase. Each section or staff member assigned faculty, class durations, subjects, and classroom locations are clearly identified in these timetables, which are displayed in a standard graphical format. The output is produced in several approaches, section-wise, faculty -wise, and classroom-wise to satisfy the needs of different users like students, faculty and administrators. Hard limitations such as avoiding duplicate assignments, classroom size, and overlapping classes are handled by CSP. Soft limitations such as equal workload, preferred timeslots and institutional regulations are optimized through the application of GA. Each output timetable has comprehensive metadata including the allocated faculty ID, room number, topic code, day, and period number. This ensures clarity and traceability in schedule interpretation. The system can also manage situations when several labs works or class groups need to be scheduled at the same time.

4. Methodology

4.1 Data Collection

Academic schedules, staff availability, course details and student data collected from the college's academic administrative system composed the dataset used in this study. The dataset's primary components consist of various kinds. The faculty availability data was collected based on the working hours, areas of expertise, and preferred timeslots of each faculty member.

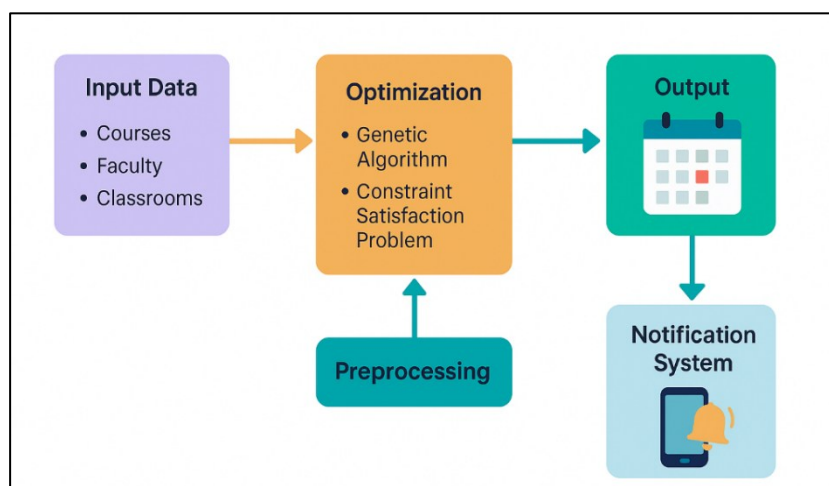


Figure 2. Architecture Diagram

The course details were gathered based on the available courses, their time duration, necessary faculty allocation, and related areas discussed. The section wise student groups data provides each section's enrollment of students and their individual course schedules. Student preferences data based on the needs of individual students, indicates preferred specific times or accommodations. The model was trained and evaluated on historical scheduling data that featured complicated conflicts as shown by the data splitting, evaluation and refinement phases. The stress testing process creates fake situations where many faculty members from several departments do not exist at the same time.

4.2 Preprocessing

In the preprocessing method, the data is cleaned, organized and prepared for use in the scheduling system. The key components of the preprocessing steps are:

- **Segmentation:** This process involves identifying the important elements of the schedule including student groups, faculty preferred times and available timeslots. The relevant data such as faculty availability and the number of faculty members, is used throughout the timetable development process.
- **Conflict_Detection:** The system searches for possible conflicts such as mismatched course enrollments, overlapping faculty schedules, and student schedule conflicts. In this scenario, if a faculty member or student has two requests, the system will detect and modify the schedule appropriately.

- **Expanding Timetable Values:** The schedule data including discussion hours, faculty availability, student group sizes and time slots is modified to ensure consistency across the data. Standardized time slots such as one-hour sections will make the processing easier.
- **Normalization for Scheduling Algorithms:** Normalization enables the scheduling model to manage limitations systematically and prevent conflicts by ensuring that each limitation such as student enrollments or faculty time preferences is handled similarly. The system's architecture indicates low-latency responses (less than five minutes) because of the preprocessing (normalization, segmentation) which reduces processing expenses and accurately provides resolution times that cannot be measured.

4.3 Model Structure

The process of structuring the model involves using machine learning techniques to handle the challenges of optimizing the timetable. The following steps are involved in the modeling structure:

- **Data Splitting:** The 25% of the data is used for testing and the remaining 75% of data is used for training process. The testing set is used to assess the model's capacity for generalization and the remaining set contains the historical data for scheduling (faculty availability, student groups and course enrollments).

The dataset details include the academic timetables, staff availability course details and student data from the college academic administration system.

- **Algorithm Selection and Training:** The system is trained on historical data, so that the model optimizes the schedule using an optimization method. The algorithm achieves the capacity to manage limitations like course duration, student schedules and faculty availability. To identify the optimal faculty to course ratio, it considers the preferences and limitations of the students.
- **Evaluation and Fine-Tuning:** Once trained, the model is tested using the testing set to evaluate how well it generates schedules that satisfy all constraints. The model is evaluated on factors like conflict-free scheduling, faculty working hours,

and student satisfaction. Fine-tuning may involve adjusting the penalties for constraint violations or optimizing the algorithm for better results.

- **Real-Time Adjustments and Optimization:** Once the model is trained, it can dynamically adjust timetables based on real-time inputs. For instance, if a faculty member is unavailable or a student's schedule changes, the system re- optimizes the timetable and provides real-time notifications to affected students and faculty.

This research focuses on optimization techniques such as Genetic Algorithm (GA) and Constraint Satisfaction Problem (CSP) instead of standard machine learning mode. The model testing depends on the 25% split which is referred to as validation. The optimization model is used to create schedules without conflicts after the data is “trained”. Schedules are regularly updated to reflect real-time changes such as faculty unavailability or new course assignments. The email notification service notifies the faculty when the schedule is established and when conflicts arise that will cause modifications. This technology will be adjusted in real time and is accessible across the departments.

5. Workflow of the Architecture

The workflow of the model is identified using the following steps:

- **Frontend/UI:** This layer is used for the timetable view and the admin dashboard.
- **Backend:** This layer offers an ML/GA module, optimization techniques and a scheduling system. GA is introduced and used in the ML/optimization module.
- **Database:** The database is used to verify the student groups, course details, faculty availability and institutional limitations.
- **Notification Subsystem:** This section is used to utilize email services and delivery management.

An AI-driven timetable management system efficiently combines faculty schedules, student data and course details in order to create an effective timetable that will reduce conflicts and handle real-time changes.

5.1 Reason for Developing this Model

The scheduling system continuously optimizes satisfaction and ensures feasibility under large-scale, dynamic and real-time limitations requiring the employment of a hybrid model. This model is more appropriate for utilizing real-world academic scheduling applications because it achieves a balance between the rigidity of CSP and the flexibility of GA.

6. Results

The AI-driven dynamic scheduling and conflict detection system is developed in this proposed work to solve the main problems associated with manually developing timetables in educational institutions. The system was implemented to simplify and automate the scheduling process by combining course and classroom allocation, faculty availability and institutional limitations into a single platform.

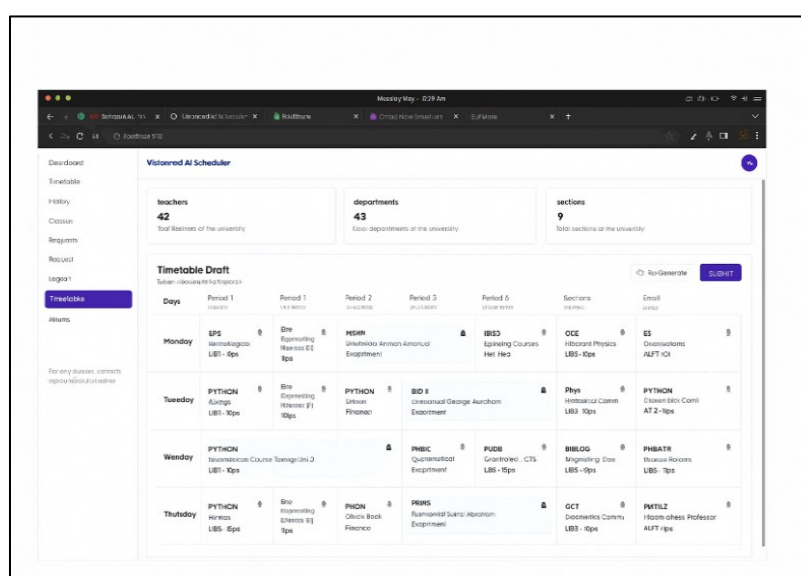


Figure 3. Time Table Scheduler

Figure 3 shows the timetable scheduler used to effectively organize class sessions and illustrates the system interface for managing and scheduling academic activities. The system's scheduling, user management, and data configuration are all under administrative control through the Admin Dashboard, which is depicted in Figure 4.

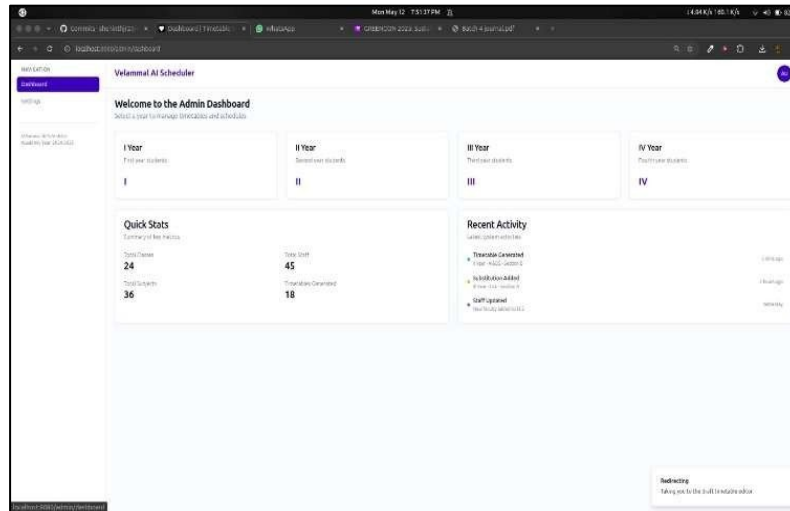


Figure 4. Admin Dashboard

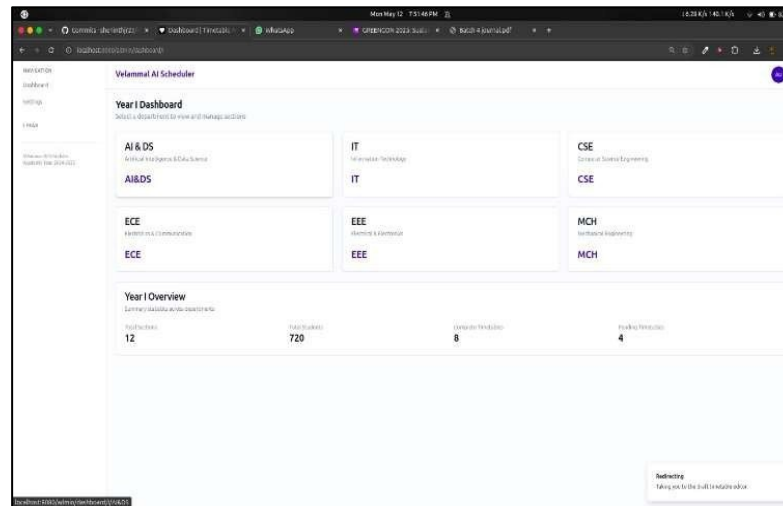


Figure 5. User Dashboard

The User Dashboard, shown in Figure 5, allows each user to view customized schedules, control preferences, and access associated features. The admin dashboard feature provides a centralized interface that easily handles faculty entries, course assignments, and limitations in educational settings. The dashboard will identify scheduling anomalies before the completion of the process by providing conflict alerts and notifications in real time. In the user dashboard, access to customized schedules for faculty and students is provided, particularly when the schedule is modified dynamically.

The proposed work is able to automatically identify and resolve issues. The system uses rule-based logical scheduling methods to manage overlapping, avoid duplicate assignments of faculty, and ensure equitable distribution of workloads. It also changes the

schedule as required to include faculty absence or unexpected situations without compromising the overall consistency of the schedule.

The proposed results illustrate the efficient implementation of different techniques developed in this work, demonstrating the importance of automation and AI-based educational management processes. The system's dynamic development of accurate schedules without conflicts indicates that AI-driven solutions will assist with academic scheduling, reduce manual effort, and improve institutions' efficiency in real time. The performance evaluation of this result identifies various qualitative advantages: schedules without conflicts, real-time flexibility in accessing the timetable, reduced manual work, and improved planning efficiency.

7. Future Enhancement

The AI-driven dynamic scheduling and real-time notification system can be optimized to enhance the performance and adaptability in the future. Initially, the system's ability to generalize the different types of education will improve by adding data to enhance functions such as including academic events, holiday lists, and institutional working calendars. Additionally, errors will be reduced by combining various scheduling algorithms; the model's performance will be enhanced by using ensemble learning methods like boosting and bagging. The model's efficiency and processing times will be increased, creating a more accurate solution by adjusting the hyperparameters.

8. Conclusion

This proposed work effectively solves the complicated method of creating an academic schedule and simplifies the model by using the AI-driven flexible Scheduling and Real-time Notification system. This method ensures conflict-free and effective scheduling for all students in the institution by carefully combining faculty availability, course allocation, and classroom access for sub-dividing student groups. There will be an immediate response to unexpected issues, such as faculty unavailability or sudden alterations in periods, which can be addressed using the dynamic conflict identification and resolution system. This system utilizes different dashboards for its functions. The user dashboard provides real-time access to customized schedules, while the admin dashboard allows complete authority over data inputs along with scheduling requirements. Users will receive notification alerts that enhance

collaboration and communication in the future. This method will improve academic processes by reducing manual labor and increasing accuracy and transparency. Overall, the proposed work illustrates the effective use of AI in educational institutions. Future improvements, such as predictive scheduling and connecting the leave and attendance system, will increase scalability across departments and institutions.

References

- [1] Sonawane, Ms Premrata A., and Leena Ragha. "Hybrid genetic algorithm and TABU search algorithm to solve class time table scheduling problem." *International Journal of Research Studies in Computer Science and Engineering* 1, no. 4 (2014): 19-26.
- [2] Devi, M. Uma, J. Jayapradha, Bhaavya Naharas, and Saniya Sharma. "Automated timetable generation for academic institutions." In *AIP Conference Proceedings*, vol. 3075, no. 1, p. 020094. AIP Publishing LLC, 2024.
- [3] Renggli, Fabienne Josefine, Maisa Gerlach, Jannic Stefan Bieri, Christoph Golz, and Murat Sariyar. "Integrating Nurse Preferences Into AI-Based Scheduling Systems: Qualitative Study." *JMIR Formative Research* 9, no. 1 (2025): e67747.
- [4] Deris, Safaai, Sigeru Omatu, and Hiroshi Ohta. "Timetable planning using the constraint-based reasoning." *Computers & Operations Research* 27, no. 9 (2000): 819-840.
- [5] Li, Yuanyuan, Stefano Carabelli, Edoardo Fadda, Daniele Manerba, Roberto Tadei, and Olivier Terzo. "Machine learning and optimization for production rescheduling in Industry 4.0." *The International Journal of Advanced Manufacturing Technology* 110, no. 9 (2020): 2445-2463.
- [6] Skobelev, Petr. "Multi-agent systems for real-time adaptive resource management." In *Industrial Agents*, pp. 207-229. Morgan Kaufmann, 2015.
- [7] Perera, K. M. P. B. N. "LECTURE HALL SCHEDULING AND TIMETABLE MANAGEMENT SYSTEM." PhD diss., 2021.

- [8] Jeong, Ji-Seong, Mihye Kim, and Kwan-Hee Yoo. "A content oriented smart education system based on cloud computing." *International Journal of Multimedia and Ubiquitous Engineering* 8, no. 6 (2013): 313-328.
- [9] Rane, Nitin. "Integrating building information modelling (BIM) and artificial intelligence (AI) for smart construction schedule, cost, quality, and safety management: challenges and opportunities." *Cost, Quality, and Safety Management: Challenges and Opportunities* (September 16, 2023) (2023).
- [10] Gupta, Poonam, Omkar Prasad Ijardar, Ashish Jadhav, and Vansh Saheb. "AI-Based Solution To Enable Ease of Grievance Lodging and Tracking for Citizens Across Multiple Departments." In *International Conference on Advances and Applications in Artificial Intelligence (ICAAAI 2025)*, Atlantis Press, 2025, 1002-1022.
- [11] Ajayi, Rhoda. "Integrating IoT and cloud computing for continuous process optimization in real-time systems." *Int J Res Publ Rev* 6, no. 1 (2025): 2540-2558.
- [12] Heggond, Shreeshail. "Artificial Intelligence and Machine Learning for Smart Construction: Enhancing Real-Time Monitoring and Decision Making." Available at SSRN 5233045 (2025).
- [13] Taye, Gemechu, Sonal Sharma, Pratishtha Shah, and Yohanes Getinet Nuriye. "Exploring the role of artificial intelligence in class scheduling and management: a comprehensive survey and review." In *2023 International Conference on Computer Science and Emerging Technologies (CSET)*, IEEE, 2023, 1-11.