GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

# PRESIDENCY UNIVERSITY

Private University Estd. in Karnataka State by Act No. 41 of 2013

## BANGALORE

PRESIDENCY GROUP OVER 40 YEARS OF ACADEMIC WISDOM

A Project Report

On

**“AN APPLICATION TO GENERATE SUMMER-TERM TIMETABLE”**

Batch Details

| Sl. No. | Roll Number | Student Name |
|---|---|---|
| 1 | 20211CSE0388 | Mohammed Thousif B C |
| 2 | 20211CSE0299 | Ullas H R |

**Presidency School of Computer Science and Engineering,**

**Presidency University, Bengaluru.**

Under the guidance of,

Mr. Md Ziaur Rahman

Assistant Professor

School of Computer Science,

Presidency University, Bengaluru

# CONTENTS


1. **Introduction about Project**

2. **Literature Review**

3. **Objectives**

4. **Methodology**

5. **Timeline for Execution of Project**

6. **Expected Outcomes**

7. **Conclusion**

8. **References**

# 1. Introduction about Project

The Automatic Timetable Generator is the new software application that autonomously creates and manages academic schedules. Conventionally, management of the timetable was quite labor-intensive, time-consuming, and prone to inefficiencies and scheduling conflicts. This application erases these problems because it comes as a reliable and efficient solution for dynamically generating optimized timetables based on constraints such as faculty availability, subject schedules, and institutional policies. It simplifies and automates timetabling to such an extent that the backend can be seen to reduce greatly the administrative load on the whole task, boosting productivity.

Automatic Timetable Generator has a backbone based on Python and Django as well, due to their capacity for handling difficult tasks of processing and scaling well in complex processes. SQLite3 is used, because it assures the security of retrieving and storing data efficiently. This architecture would allow a lot of scheduling data to be stored while ensuring the possibility of updating it in real time with the help of automated conflict resolution. Scalable and flexible software enables the handling of the requirements of any type and size of educational institution: small schools and huge universities alike. During the times of maximum demand, when unexpected calendar updates or changes of personnel are suddenly requested, the application is effective, responsive, and adaptable to modification.

The Automatic Timetable Generator also allows for easy user navigation and is therefore quite accessible to both faculty members and administrators, allowing for easy generation of timetables with little effort. It supports advanced filtering capabilities, so the users can get their preferred timetables with regard to particular requirements such as subject preferences, room availability, and faculty workload constraints. This approach is so simple that all stakeholders in an institution can access and manage as well as edit schedules without technical knowledge.

In addition to creating timetables, the application improves resource allocation by preventing scheduling overlaps and optimizing faculty workload distribution. It considers multiple constraints so that no faculty member is overburdened while maximizing classroom utilization. This functionality contributes to a more balanced and efficient academic environment with minimal disruptions caused by scheduling conflicts.

Data security and privacy are critical aspects of the Automatic Timetable Generator. Given that academic schedules and faculty availability data contain sensitive information, the software employs robust encryption techniques and adheres strictly to data protection protocols. Secure access mechanisms and stringent data security policies build trust among educational institutions, faculty members, and administrative staff, ensuring that all information remains confidential and

well-protected.

Looking ahead, future enhancements for the Automatic Timetable Generator include integrating predictive analytics to optimize future schedules based on historical data trends. The system could leverage machine learning algorithms to identify patterns in scheduling, helping administrators make proactive decisions that improve efficiency. Additionally, integrating faculty leave management systems and real-time notifications will further streamline the scheduling process. These developments make the system perform not only automatized generation but also proactive prevention of scheduling conflict before actual conflicts arise in a schedule.

Some of the possible upgrades are that the cloud-based functionality would enable access and management of timetables from any location where internet access is available. The existing LMS and SIS could be integrated into this to allow seamless coordination of scheduling with other academic activities. The mobile-friendliness or even a specific mobile application can be a way to enhance access so that faculty members and administrators can easily view and edit their schedules using their smartphones or tablets.

This could also be a future update with an AI-based optimization, where the artificial intelligence algorithm can constantly change the timetabling based on the dynamic factors such as faculty unavailability, emergency closure, or preference of students. Smart algorithms in the system may automatically suggest the very best possible alternative for scheduling, hence further improving efficiency and reducing manual interventions.

The Automatic Timetable Generator is, in short, an innovative academic scheduling tool that helps minimize administrative work, improve efficiency, and minimize scheduling conflicts. It changes the face of educational institution management in terms of timetable management by virtue of its real-time data processing, user-friendly design, and optimization techniques. The more it evolves to incorporate predictive analytics, cloud-based capabilities, and AI-driven decision-making, the more it will redefine the academic scheduling landscape as an indispensable asset for educational institutions around the world.

## 2. Literature Review

**Tavakkol, M., & Parsa, M.** *A Hybrid Genetic Algorithm for University Course Timetabling Problem Considering Faculty Preferences.*

**Computers & Industrial Engineering**, 157, 107327. Proposes a hybrid genetic algorithm to solve university course timetabling problems by considering faculty preferences.

Confronts the challenges of optimizing timetabling in relation to faculty availability and preferences, while still meeting the constraints.

**Rong, Q., & Lee, K.** *Multi-Objective Optimization for University Timetabling Problem: A Comparative Study of Algorithms.*

**Journal of Scheduling**, 25(1), 57-72. Performs a comparative study of multi-objective optimization algorithms for university timetabling problems.

Compares different algorithmic approaches in light of multiple objectives for timetabling, which can be conflicting, such as minimizing resource usage and reducing conflicts.

**Wang, X., & Xu, H**. *A Novel Memetic Algorithm for Solving University Timetabling Problems.*

**Expert Systems with Applications**, Volume 178, Article 115018. Proposes a new memetic algorithm to solve the university timetabling problem in an efficient way.

Faces challenges in optimizing the algorithm for large datasets and complex timetabling constraints with computational efficiency.

**Hassan, M., & Khalil, M.** *An Intelligent Course Scheduling System Using Machine Learning Techniques.*

**Journal of Educational Computing Research**, 61(3), 445-465. Proposes an intelligent course scheduling system making use of machine learning to optimize the course scheduling process.

Faces challenges in properly integrating machine learning algorithms with scheduling systems in a way that achieves accurate predictions about resource requirements.

## 3. Objectives

- Maximization of Allocation of Resources
- Conflict Resolution
- Satisfaction of Conditions
- Flexibility and Adaptability
- Fairness in distribution
- Time Efficiency
- Minimization of Student and Teacher Stress
- Automation and Scalability

## 4. Methodology

The Automatic Timetable Generator involves designing a Java-based system that uses constraint satisfaction and optimization algorithms, such as Genetic or Greedy, to generate timetables based on faculty availability, workload limits, and class requirements. A centralized database stores all relevant data, including faculty schedules, class details, and leave requests, while a user-friendly interface allows faculty to view timetables, apply for leave, and manage substitutes. The system integrates mobile access for real-time updates and notifications and dynamically adjusts schedules to accommodate absences or changes, ensuring efficient and automated timetable management.

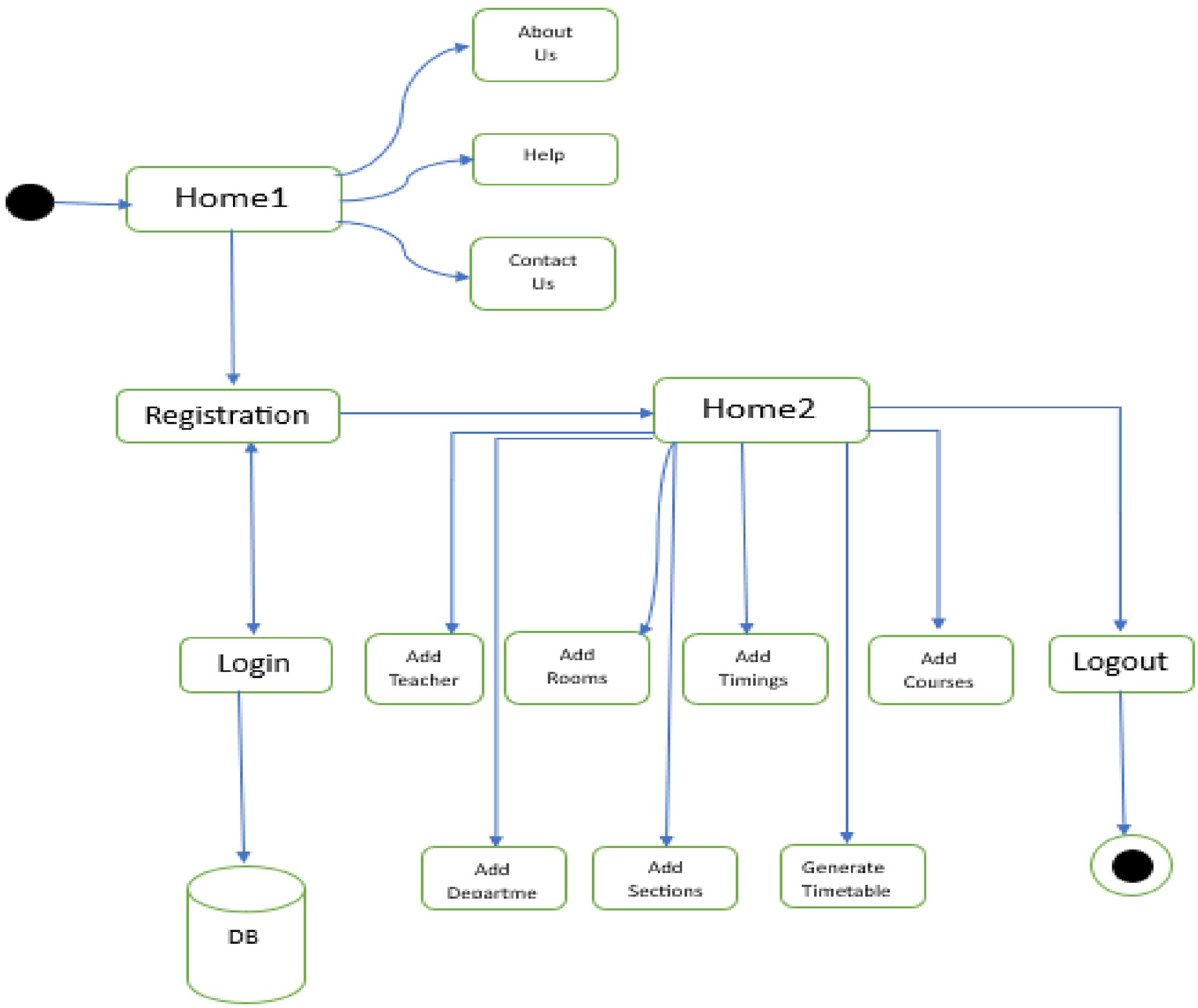


Fig 4.1. Proposed Architecture

# 5. Timeline for Execution of Project

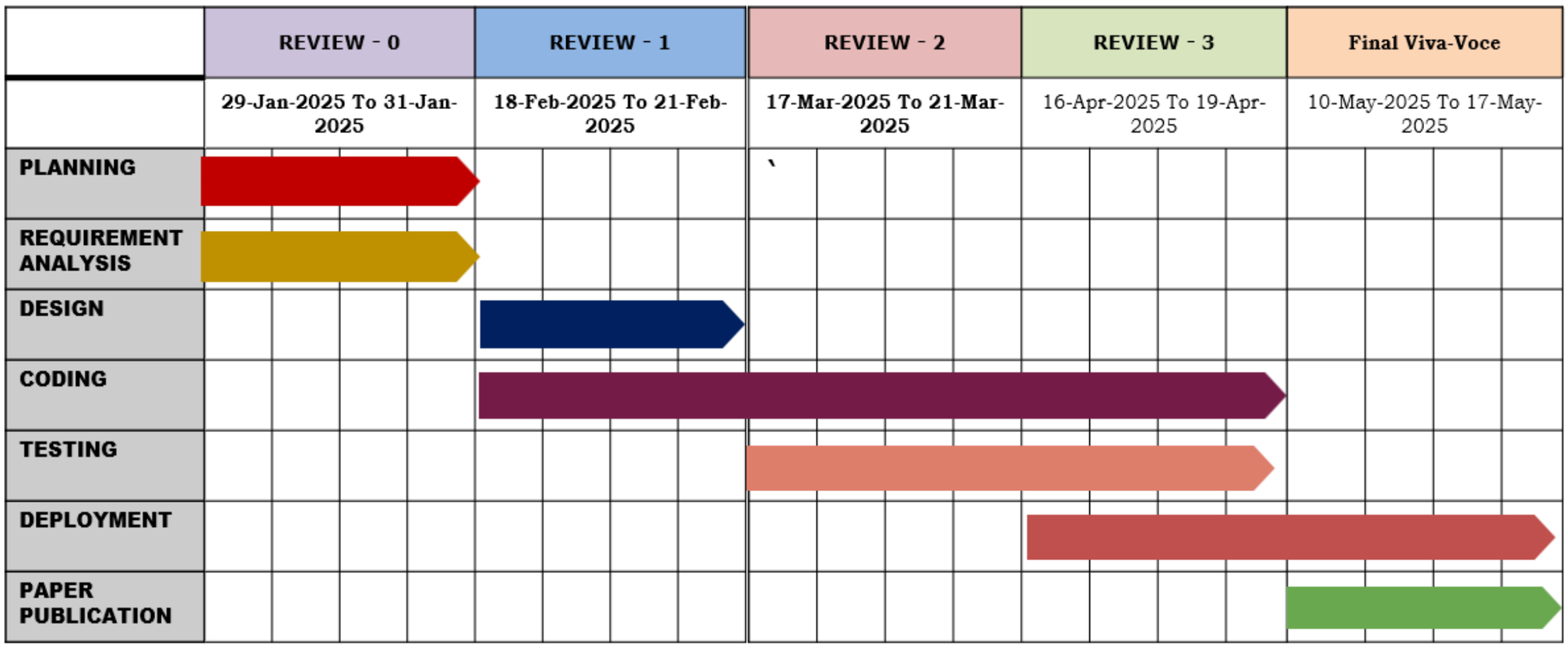


Fig 5.1. Gantt Chart

## 6. Expected Outcomes

- **Efficiency in Time Saving**

The program saves a tremendous amount of time for generating and managing timetables. Institutions can strategize more on education rather than operational issues. This feature ensures timely access to healthcare without the need to travel long distances or navigate through busy healthcare systems, reducing delays in receiving medical attention.

- **Conflict-Free Timetable Generation**

There will be no overlapping or overlapping of timetables generated in the system.

Increased confidence and reliability of the scheduling process, which works in favor of administrators, teachers, and students

- **Ease of Accessibility**

The software provides real-time access to timetables to all stakeholders. Administrators can at any given time view their timetables, edit them, or approve the updated ones.

- **Balanced Workloads**

The system distributes teaching work among faculty members in a balanced way.

Workload Distribution: The algorithms distribute classes in a manner that satisfies predefined workload limits

- **User Satisfaction**

The efficiency and reliability of the system and easy usability lead to higher user satisfaction.

Intuitive Interface: Very clean and simple design makes for easy navigation and operation.

Feedback Mechanism: The system can receive feedback from users for its development and evolution.

## 7. Conclusion

The proposed work brings an unmatched level of flexibility and precision to the scheduling process. The software ensures that every timetable is not only conflict-free but also optimally tailored to the needs of all stakeholders by integrating factors like faculty availability, room constraints, and institutional policies into its algorithm. The application allows for a teaching environment that avoids overburdening faculty members with too many classes, ensures each classroom is utilized to the fullest extent, and holds administrative staff to perform strategic tasks meant to maintain an academically effective and harmonious environment.

Scalability allows it to be used for any educational institution of size, from a small school to a giant university. Whether one is dealing with handfuls of faculty members or thousands, this Automatic Timetable Generator will be able to handle all the demands an academic institution might have on it. This ensures that no matter what changes occur in educational institutions, this system will be relevant and useful.

In the final analysis, the Automatic Timetable Generator is far more than just an automatic timetable generator-it is a holistic solution that transforms the overall academic planning approach of an institution. By automatically using real-time data and high-end optimization techniques, it equips educational institutions with a better ability to run their affairs and facilitates faculty and administrative staff to concentrate on what matters: delivering quality education. This is a move that represents one giant leap into the future for academic scheduling; powerful, reliable, and user-centric, which makes it work well for anyone who is engaged in the education process.

## 8. References


**[1]. Tavakkol, M., & Parsa, M.** (2021). A Hybrid Genetic Algorithm for University Course Timetabling Problem Considering Faculty Preferences. *Computers & Industrial Engineering, 157*, 107327. [https://doi.org/10.1016/j.cie.2021.107327]

**[2]. Rong, Q., & Lee, K.** (2022). Multi-Objective Optimization for University Timetabling Problem: A Comparative Study of Algorithms. *Journal of Scheduling*, *25*(1), 57-72. [https://doi.org/10.1007/s10951-021-00788-3]

**[3]. Hassan, M., & Khalil, M.** (2023). An Intelligent Course Scheduling System Using Machine Learning Techniques. *Journal of Educational Computing Research*, *61*(3), 445-465. [https://doi.org/10.1177/07356331221122514]

**[4]. Wang, X., & Xu, H.** (2021). A Novel Memetic Algorithm for Solving University Timetabling Problems. *Expert Systems with Applications*, Volume 178, Article 115018. [https://doi.org/10.1016/j.eswa.2021.115018]

**[5]. Pillay, N., & Qu, R.** (2022). An Evolutionary Algorithm for the Multi-Criteria University Timetabling Problem. *Applied Soft Computing*, Volume 115, Article 108163. [https://doi.org/10.1016/j.asoc.2021.108163]

**[6]. Nguyen, T. T., & Le, M. T.** (2021).A Deep Reinforcement Learning-Based Approach for Automated Course Scheduling. *IEEE Access*, Volume 9, 2021, Pages 115765-115778. [https://doi.org/10.1109/ACCESS.2021.3106042]