

# PROJECT TIME-TABLE

Good afternoon, we will be presenting our Project synopsis on Web-based Automatic Timetable Scheduler for Schools and Colleges.

## Problem Identification:

Timetable scheduling is a crucial yet challenging task for educational institutions. Manual timetable creation is “**time-consuming, error-prone, and inefficient**”. Key challenges faced during the process are:

- Scheduling conflicts between teachers, students, and classrooms.
- Ensuring proper distribution of subjects across available time slots.
- Balancing workloads while considering faculty preferences.
- Accommodating last minute changes.

These challenges make manual scheduling inefficient, leading to delays, increased workload for the coordinator, and potential dissatisfaction among the faculty.

## Existing Solution:

Traditionally, timetable scheduling is handled manually by a coordinator or a faculty member. This process involves:

Listing all courses, faculty members, and available classrooms.

- Manually assigning time slots while considering availability constraints.
- Adjusting schedules to resolve conflicts.
- Handling last-minute changes, which often lead to disruptions.

Major disadvantages in this process is., it is time consuming and prone to errors. Also, it is difficult to adjust the schedules dynamically when conflicts arise.

## Proposed Solution:

To overcome these challenges, we propose a Web based automatic Timetable scheduler using Evolutionary algorithms like Genetic Algorithm. It is a metaheuristic inspired by natural selection, designed to optimize complex problems like scheduling. This method ensures:

- Minimized timetable conflicts.
- Optimal utilization of resources (classrooms, faculty, and time slots).
- Reduction in human intervention and errors.

## User Requirements:

These are the system requirements: i.e., the system should have 4GB of RAM, windows 7 and upwards, Python version atleast 3.8, SQLite for default database, internet access.

Project components include,

- identifying the time constraints of the institution,
- proper details about the course, faculty- schedule, labs, classes, classrooms etc
- Allow manual adjustments, as per requirements.

## FlowChart: (Mention this content in your words looking at the diagram)

This flowchart represents the **step-by-step process** of timetable generation:

- **Start** – Initiates the scheduling process.
- **Input Details** – Admin provides details like subjects, faculty, and classrooms.
- **Processing Inputs using Genetic Algorithm** – The system applies Genetic Algorithm to optimize the schedule.
- **Generating Timetable** – The optimized timetable is created.
- **View Generated Timetable** – Users can review the timetable.
- **Stop** – The process ends.

## Activity Diagram: (Mention this content in your words looking at the diagram)

This activity diagram provides a **detailed workflow** of the timetable scheduling system:

- **Admin Login:** If login is **valid**, access is granted.  
If login is **invalid**, the process terminates.
- **Home Page:** The admin adds **course details** and **faculty details**.
- **Details:** The system collects constraints such as faculty availability, time constraints etc.,
- **Generate Timetable using Algorithm:** The **Genetic Algorithm** processes the **input data** and creates the timetable.
- **View Timetable:** The admin can view the generated timetable.
- **Logout & Terminate:** After viewing, the admin logs out, ending the session.

### Methodology:

The development of the **Web-based Automatic Timetable Scheduler** follows a structured approach to ensure efficiency and usability.

1. **Design and Development:** The system is designed to provide a **user-friendly interface** that allows administrators to easily input data and generate optimized timetables.
  - a. **HTML and CSS** for front-end development.
  - b. **Django**, for back-end development and along with PyPi libraries for data handling.
  - c. **email system** to facilitate **user account creation, notifications, and communication**.

2. **Deployment:**

**Deployment Options:** The system can be hosted on **AWS, Google Cloud, or a dedicated server** for optimal performance.

**Version Control & Management:** **GitHub** is used for **collaborative development and version control**.

### Conclusion:

As discussed, an evolutionary algorithm, genetics algorithm for time tabling has been proposed. This solution significantly improves efficiency, accuracy, and adaptability. It minimizes manual effort, reduces scheduling conflicts, and optimizes resource utilization. By implementing this solution, educational institutions can streamline the scheduling process, enhance faculty satisfaction, and ensure a smooth academic workflow.

---

GA is a metaheuristic inspired by natural selection, designed to optimize complex problems like scheduling

### **Key Features of the Genetic Algorithm Approach:**

- **Population Initialization:** Generates multiple potential timetables randomly.
- **Selection:** Identifies the best timetables based on predefined constraints.
- **Crossover:** Combines elements from multiple schedules to create optimized ones.
- **Mutation:** Introduces small changes to explore better scheduling solutions.
- **Fitness Function:** Evaluates schedules based on factors like minimal conflicts, optimal resource usage, and balanced faculty workload