

PROJECT SYNOPSIS

Department of Computer Engineering

MIT Academy of Engineering, Pune

Project Title: SamaySetu

Group Members: 202301040200 : Suryank Chavan

202301040205 : Swarup Daphal

202301040228 : Abhijeet Rane

202301040268 : Rohan Wani

Guide: Mr. Vijaykumar Mantri

1. Title of the Project

SamaySetu: A Smart Automated Timetable Generator for Educational Institutions

2. Introduction

Academic institutions face growing challenges in preparing accurate, fair, and clash-free timetables. Manual scheduling methods are not only time-consuming but also prone to human error, resulting in conflicts, uneven workload distribution, and inefficient utilization of resources.

SamaySetu aims to overcome these challenges by providing an automated, AI-assisted, and user-friendly timetable generator. By combining constraint-based optimization techniques, structured databases, and a dynamic web interface, the system will automate the scheduling process, reduce manual effort, and improve academic management efficiency.

3. Objectives

- To develop a robust and scalable system for dynamic timetable generation.

- To accommodate the addition and modification of teachers, courses, divisions, and classrooms.
- To ensure conflict-free schedules by managing both hard and soft constraints.
- To build a user-friendly GUI for administrators and faculty members.
- To allow dynamic rescheduling for emergency cases like faculty absence or room unavailability.
- To minimize manual effort, reduce errors, and improve institutional efficiency.
- To provide export features for timetables in PDF, Excel, and image formats.

4. Problem Statement

Manual timetable preparation is tedious, error-prone, and difficult to scale with institutional growth. Overlaps between teachers, classrooms, and courses are common. Last-minute changes often require complete rework, further increasing workload. Therefore, a dynamic and intelligent timetable generator is required to efficiently create, manage, and update conflict-free schedules.

5. Scope of the Project

- **Colleges and Universities:** Large-scale scheduling for multiple years, branches, and divisions.
- **Schools and Coaching Institutes:** Simplified and error-free timetable management.
- **Corporate Training Programs:** Scheduling training sessions, workshops, and events.
- **Event Scheduling:** Can be extended for conferences, seminars, and workshops.
- **Administrative Use:** Centralized data storage and updates for better decision-making.

6. Literature Review / Existing System

1. Opeyemi Alonge et al. (2025): Evolutionary algorithms like NSGA-II reduce conflicts and balance fairness.

2. Francis P. Diallo et al. (2024): Constraint-based optimization (Timefold Solver) improves large-scale scheduling.
3. Rutuja Kavade et al. (2023): AI and Genetic Algorithms generate flexible and conflict-free timetables.
4. Noorrezam Yusop (2022): Highlighted importance of human factors like work-life balance in timetable optimization.
5. Abhishek M. B. et al. (2024): Centralized campus-focused timetable system reduces manual workload.

7. Proposed System / Methodology

- **Database:** MySQL for storing structured timetable data (teachers, classrooms, subjects, divisions).
- **Frontend:** React.js + TypeScript for dynamic data input and visualization.
- **Backend:** Java Spring Boot with REST APIs for business logic and database communication.
- **Scheduling Engine:** OptaPlanner to generate optimized conflict-free timetables.
- **Visualization:** Division-wise, teacher-wise, and classroom-wise grid-based timetable views.
- **Export:** iText (PDF), Apache POI (Excel) for record keeping and distribution.

8. Modules of the Project

The project is divided into six major modules, each responsible for handling a critical part of the system.

1. **User Management Module:** Provides secure authentication and role-based access for administrators, faculty, and coordinators. It ensures data security and prevents unauthorized changes in the timetable system.
2. **Teacher and Course Management Module:** Handles details of teachers, their subjects, availability, and workload preferences. It ensures fair distribution of classes and maintains flexibility for updates in teacher assignments.

3. **Division and Classroom Management Module:** Manages division structures, classrooms, and laboratory resources. It prevents conflicts in infrastructure usage and ensures efficient allocation of physical resources.
4. **Timetable Generation Module:** The core scheduling engine that uses OptaPlanner to generate conflict-free timetables. It balances strict constraints (no overlaps) with flexible ones (teacher preferences, fairness) to optimize schedules.
5. **Visualization Module:** Displays timetables in grid-based views. Users can see division-wise, teacher-wise, and classroom-wise timetables. The module provides an interactive interface for quick analysis and modifications.
6. **Export and Reporting Module:** Supports exporting timetables to PDF, Excel, and image formats for distribution and record-keeping. It allows administrators to share finalized schedules with faculty and students efficiently.

9. Project Files Structure

The backend implementation of **SamaySetu** follows a modular and well-organized structure to ensure maintainability, scalability, and clarity in development. The files are organized as follows:

```

Backend/
  src/main/java/com/College/timetable/
    Controller/
      DepartmentController.java
      TeacherController.java
      CourseController.java
      RoomController.java
      DivisionController.java
      AcademicController.java

    Entity/
      DepartmentEntity.java
      TeacherEntity.java
      CourseEntity.java
      ClassRoom.java
      Division.java
      AcademicYear.java
      TimeSlot.java          (NEW)
      TimetableEntry.java    (NEW)

```

```
TeacherAvailability.java (NEW)
Student.java (NEW)
DayOfWeek.java (NEW)
CourseType.java
RoomType.java
Semester.java
```

```
Repository/
Dep_repo.java
Teacher_Repo.java
Course_repo.java
Room_repo.java
Division_repo.java
Academic_repo.java
TimeSlot_repo.java (NEW)
TimetableEntry_repo.java (NEW)
TeacherAvailability_repo.java (NEW)
Student_repo.java (NEW)
```

```
Service/
DepartmentService.java
TeacherService.java
CourseService.java
RoomService.java
DivisionService.java
AcademicService.java
```

```
Review1Application.java
```

```
src/main/resources/
application.properties
```

```
Documentation/
ENTITY_IMPLEMENTATION_SUMMARY.md
API_ENDPOINTS_REFERENCE.md
IMPLEMENTATION_COMPLETE.md
QUICK_START_GUIDE.md
```

This structured approach ensures separation of concerns:

- **Controllers:** Handle incoming API requests and responses.
- **Entities:** Represent core objects such as teachers, courses, classrooms, and timetables.
- **Repositories:** Provide data persistence and database operations using JPA.
- **Services:** Contain business logic for timetable generation and management.
- **Documentation:** Provides references for APIs, entity descriptions, and setup guides.

10. Expected Outcomes / Deliverables

- Automated timetable generation with no clashes.
- Scalable solution for institutions of all sizes.
- Fair distribution of faculty workload.
- Centralized data storage and easy retrieval.
- Exportable timetables in multiple formats.
- Enhanced efficiency and reduced manual workload.

11. Challenges and Risks

Despite the advantages, the project may face the following challenges and risks:

- **Data Accuracy and Completeness:** Incomplete or incorrect data (e.g., wrong faculty availability) can result in invalid timetables.
- **Constraint Modeling Complexity:** Balancing strict constraints (no clashes) with softer ones (preferences, fairness) is difficult and requires careful tuning.
- **Performance and Scalability:** Large institutions with hundreds of divisions and courses may increase computation time.
- **Dynamic Rescheduling:** Handling last-minute changes such as faculty leave or room unavailability without disrupting the entire schedule.
- **Solver Stability:** Poor parameter tuning in OptaPlanner may lead to unstable or suboptimal schedules.

- **Concurrency Issues:** Multiple admins making changes simultaneously could cause data conflicts; mechanisms like locking and transactions are needed.
- **Security and Role-based Access:** Protecting sensitive faculty data and ensuring proper access control.
- **System Maintainability:** Ensuring long-term maintainability of APIs, database schema, and scheduling algorithms.

12. Hardware & Software Requirements

Hardware:

- Processor: Intel i5 or higher
- RAM: 8 GB or higher
- Storage: 512 GB SSD

Software:

- Backend: Java (Spring Boot)
- Frontend: React.js, TypeScript, HTML, CSS
- Database: MySQL
- Scheduling Engine: OptaPlanner
- Export Libraries: iText (PDF), Apache POI (Excel)

13. Project Plan / Timeline

1. Week 1-2: Requirement gathering, literature survey, and database schema design.
2. Week 3-4: Development of frontend and backend modules.
3. Week 5-6: Implementation of scheduling algorithm (OptaPlanner).
4. Week 7: Integration of frontend, backend, and database.
5. Week 8: Testing and debugging with real-world datasets.
6. Week 9: Deployment, final documentation, and presentation.

14. Conclusion

The proposed project, **SamaySetu**, offers an advanced solution to the long-standing challenges of timetable generation. By integrating constraint-based optimization with modern web technologies, it ensures accuracy, fairness, and adaptability in academic scheduling. The system reduces human effort, eliminates conflicts, and enhances resource utilization. With scalability and modularity, SamaySetu can be extended beyond academic institutions to corporate training and event management, proving its relevance across diverse domains.

15. References

1. Opeyemi Alonge and Wilson Sakpere, “An Automated Timetable Scheduler Using NSGA-II for Optimized Scheduling in Educational Institutions,” SSRN, Jan 2025.
2. Francis Patrick Diallo, Cătălin Tudose, “Optimizing the Scheduling of Teaching Activities in a Faculty,” Applied Sciences, vol. 14, no. 20, 2024.
3. Rutuja Kavade, Sohail Qureshi, Nikita Veer, Vaishnavi Ugale, Priyanka Agrawal, “Smart Timetable System Using AI and ML,” IJCRT, May 2023.
4. Noorrezam Yusop, “A Systematic Literature Review: Optimization Timetable in Education to Support Work-Life Balance,” JCRINN, 2022.
5. Abhishek M. B., Vibha T. G., Bindu H. M., Lavanya K., “Smart Timetable Scheduler and Management System for Campus,” Campus Research Paper, 2024.

Student's Signature:

Suryank Chavan
Swarup Daphal
Abhijeet Rane
Rohan Wani

Guide's Signature:

Mr. Vijaykumar Mantri