

Timetable Creator

Project Overview

1. Technology Stack

Frontend:

- JSP (JavaServer Pages): For creating dynamic web pages.
- HTML/CSS/JavaScript: For form creation and basic client-side interactivity.

Backend:

- Java (Servlets & JSP): For core business logic and server-side operations.
- JDBC (Java Database Connectivity): To establish a connection between Java applications and MySQL.

Database:

- MySQL: To store data related to courses, teachers, classrooms, and schedules.

2. Key Features

- Admin panel for managing courses, subjects, and teachers.
- Automatic timetable generation based on predefined constraints.
- Conflict-free scheduling (no teacher or room overlaps).
- View, download, and print timetables.
- Update or delete existing schedules.
- User authentication and secure access.

3. System Architecture

Frontend (React.js):

- Developed with React and React Router
- Responsive UI using Bootstrap/Tailwind CSS
- Axios used for API communication

Backend (Node.js + Express.js):

- RESTful API development
- Handles authentication, CRUD operations, and timetable logic
- JWT-based authentication

Database (MongoDB):

- NoSQL database for flexible data storage
- Collections: Users, Timetables, Courses, Classrooms, Subjects

4. Modules

I. User Module

- Admin: Manage overall system
- Teacher: View and manage their schedule
- Student: View personal timetable

II. Timetable Module

- Define days, periods, and constraints
- Auto-generate or manually edit timetables
- Save and version control

III. Course Management Module

- Add/edit/delete courses and subjects
- Assign faculty to subjects

IV. Availability Module

- Mark faculty availability
- Prevent scheduling conflicts

5. Future Enhancements

- Integration with academic calendars
- AI-powered automatic timetable optimization
- Mobile app support (React Native)
- Role for parents to view their child's schedule
- Real-time chat support for admin and faculty

6. Conclusion

The Timetable Creator project simplifies academic scheduling using modern web technologies. It improves institutional efficiency, minimizes human error, and ensures a smooth scheduling experience for all stakeholders. Leveraging the power of the MERN stack makes it highly scalable and maintainable for future needs.