TimeTable Generator Backend - Simple Summary

What Does It Do?

This backend automatically creates perfect college timetables. You give it your teachers, subjects, rooms, student groups, and time slots - and it generates a schedule where nobody has conflicts.

The Problem It Solves

Creating timetables manually is incredibly difficult because you have to avoid:

- Teachers being scheduled to teach two classes at the same time
- Rooms being double-booked for multiple classes
- Students being required to attend two classes simultaneously
- Lab sessions being split across non-consecutive time periods

Our backend solves all these problems automatically using smart algorithms.

How It Works (3 Simple Steps)

Step 1 - You Provide the Data:

You enter all your college information including teacher names and availability, subject details, room capacities and types, student groups, and available time slots throughout the week.

Step 2 - Smart Processing:

The backend uses two powerful algorithms (CSP Solver and Genetic Algorithm) to analyze all possibilities and create optimal schedules while checking every constraint to ensure zero conflicts.

Step 3 - You Get Results:

The system returns a perfect timetable with no conflicts, balanced teacher workloads, consecutive lab sessions, and efficient room utilization.

The Two Smart Algorithms

Algorithm 1: CSP Solver (Constraint Satisfaction Problem)

Think of this like solving a Sudoku puzzle with strict rules. The CSP solver guarantees a 100% valid timetable with absolutely zero conflicts. It's very fast for small to medium-sized colleges. It works by treating timetable creation as a mathematical puzzle where every rule must be satisfied.

Algorithm 2: Genetic Algorithm

This works like evolution in nature. The algorithm creates 50 random timetables, rates each one on quality, keeps the best ones, combines them to create new versions, makes random mutations for variety, and repeats this process for 100 generations. By the end, you get a really high-quality timetable that not only works but is optimized for teacher satisfaction.

What We Built - The Five Layers

Layer 1 - API Layer:

This is the front door where requests come in. It provides REST endpoints for managing teachers, subjects, rooms, and timetables.

Layer 2 - Services Layer:

This is the business logic coordinator that handles all the operations and coordinates between different parts of the system.

Layer 3 - Solver Layer:

This is the brain of the system where the CSP and Genetic algorithms live and do the actual timetable generation.

Layer 4 - Domain Layer:

This contains all the rules and constraints that define what makes a valid timetable.

Layer 5 - Database Layer:

This is where all data is stored using SQLite, including teachers, subjects, rooms, and generated timetables.

Key Features

The backend includes a complete REST API with search functionality, filtering options, and pagination for easy data browsing. It has automatic conflict detection to ensure no scheduling errors. Lab continuity ensures that 2-hour or 3-hour labs stay in consecutive time slots. Teacher workload optimization balances classes fairly across all teachers. Gap minimization reduces idle periods in teacher schedules. Duplicate validation prevents creating the same teacher, room, or subject twice. Detailed error messages help you understand and fix any issues quickly.

Smart Constraint System

Hard Constraints - These MUST Be Followed:

A teacher cannot teach two classes simultaneously. A room cannot host two classes at the same time. A student group cannot attend two classes at once. Lab sessions must be scheduled in consecutive time blocks. Rooms must have sufficient capacity for the student group size.

If any hard constraint is violated, the timetable is invalid and gets a penalty of -1000 points.

Soft Constraints - Nice To Have:

Minimize gaps between classes in a teacher's daily schedule. Balance workload evenly across all teachers. Avoid giving teachers four or more consecutive classes without breaks. Try to schedule classes in preferred time slots when possible.

Soft constraint violations get smaller penalties of -10 points each. The goal is to maximize the total score to get the best possible timetable.

Recent Improvements We Made

Lab Scheduling: Previously, labs could be split across random time slots which made no sense. Now they're properly scheduled in consecutive blocks.

Quality Optimization: Before, the system only checked for conflicts. Now it actively optimizes for teacher satisfaction by minimizing gaps and balancing workloads.

API Features: We upgraded from basic create and read operations to full CRUD (Create, Read, Update, Delete) with advanced search and filtering.

Error Handling: Generic error messages were replaced with detailed, specific feedback that tells you exactly what went wrong and how to fix it.

Technology Stack

We built this using FastAPI, a modern Python web framework that's fast and easy to use. SQLAlchemy handles database operations with clean, Pythonic code. Google OR-Tools provides the powerful CSP solver. SQLite serves as a lightweight, file-based database perfect for this application. The entire backend follows Clean Architecture principles, making it maintainable and scalable.

Current Status

The backend is 100% complete and production-ready!

All constraints are fully implemented and tested. Both solving algorithms are working perfectly. The complete API includes validation and error handling. The system is currently running on http://localhost:8000 and ready to use.

Next Step: Build a frontend web interface so users can interact with this powerful backend through a beautiful, easy-to-use interface.

The Bottom Line

This backend is an intelligent timetable generator that takes your college data, uses advanced AI algorithms to create perfect schedules, guarantees zero conflicts while optimizing for quality, and provides complete REST APIs to manage everything.

No more spending hours manually creating timetables and fixing conflicts. Let the algorithms do the heavy lifting while you focus on what matters!

Ready to revolutionize your timetable creation process!