SmartSchedule: Automated Student Course Planner

by

Samerah Al Daher
Yasmine Sati
Kawthar Abdallah
Rein Ghattas
Ali Yehya

Project Description

Submitted in Partial Fulfillment of the Requirements for the

Software Engineering Course

Department of Computer Engineering

Faculty of Engineering

Spring 2024-2025

جـامعة بيروت العربية
BEIRUT ARAB UNIVERSITY

Contents

List of Tables

Project Description

Project Overview

SmartSchedule is a web-based platform designed to help students manage their courses more smoothly and effectively. The system automates the process of selecting courses for registration based on the student's academic status, preferences, and university course offerings. By integrating Flask (Python) as the backend and MySQL as the database, the platform provides a RESTful API that enables students to access course offerings, track their enrolled classes, and generate optimized schedules effortlessly.

Objectives

1. **Simplify Course Selection:** Allow students to view available courses, including time slots, professors, and prerequisites, in one place, and to have their auto-generated schedule.
2. **Prevent Scheduling Conflicts:** Automatically check for overlapping courses and suggest the best available option.
3. **Accessibility:** Enable students to track their enrolled courses and adjust schedules easily.
4. **Smooth Database Integration:** Store student data, course offerings, and enrollment details in a structured database.
5. **Future Scalability:** Allow expansion for additional features such as a chatbot for advising.

Background

Selecting courses is a crucial issue in each student's study plan since good management of courses surely ensures a smooth and comfortable learning process. However, managing course registration each semester can be stressful, complicated and time-consuming for students. This issue is caused by the manual course searching, the complexity of different available sections, and the variety of choices for the plan. This project aims to simplify the student course selection process by providing a structured and automated scheduling system with modern web applications and API-driven automation. With this system, students can focus on their education rather than their administrative tasks.

Literature Review

The following studies and tools provide valuable insights into the design and implementation of course scheduling systems. Each has its strengths and limitations, which have informed the development of SmartSchedule.

Li and Womer (2009):

This study presents a hybrid MILP/CP Benders Decomposition algorithm for scheduling projects with multi-skilled personnel. The approach effectively separates temporal and logical feasibility, making it suitable for complex planning and scheduling problems. However, it requires expertise in Benders Decomposition and may not be easily adaptable to university course scheduling. [1]

Garrido and Onaindia (2010):
This work explores the application of AI planning techniques in e-learning environments. The authors propose an integrated planning and scheduling approach that accommodates temporal and resource constraints, making it applicable to real-world scenarios. While the approach is flexible and supports multi-criteria optimization, its practical implementation challenges are not extensively discussed. [2]

Ajanovski (2013):
This study develops a system for creating teacher and teaching schedules based on specific requirements. The system prioritizes teachers based on their teaching load and subject specifications, ensuring efficient scheduling. While the system is effective for teacher scheduling, it lacks detailed information on its functionalities and adaptability to student course scheduling. [3]

Yang and Xie (2017):
This research proposes a genetic algorithm-based approach for university course scheduling. The algorithm incorporates coevolution to improve performance and efficiently generates high-quality scheduling solutions. The study demonstrates the effectiveness of genetic algorithms in addressing complex scheduling problems, though it may struggle with local optima in some cases. [4]

Hossain et al. (2019):
This study employs a particle swarm optimization algorithm to tackle university course scheduling. The algorithm incorporates a forceful swap operation and a repair mechanism to handle constraints effectively. The experimental results demonstrate its efficiency, but further testing is needed to validate its performance in diverse scenarios. [5]

Tavakoli et al. (2020):
This research proposes a three-stage heuristic algorithm for university course timetabling. The algorithm is applied to the industrial engineering department of a university, achieving a 96% course presentation rate. The approach shows promise but requires further validation in other academic departments. [6]

Yu Chen et al. (2022):
This research proposes an integer model and a genetic algorithm for university class scheduling. The algorithm aligns satisfaction values with the preferences of professors and students, successfully generating classroom timelines. The study highlights the effectiveness of genetic algorithms in addressing NP-hard scheduling problems. [7]

Beirut Arab University FYP Project (2023):
This project, developed by Ahmad T. Shaaban, Ahmad S. Abdellatif, Mohammad S. Abu Khurj, and Ali W. Choker, introduces a web-based application for course scheduling at Beirut Arab University. The system uses Next.js, Puppeteer, and DialogFlow to automate course registration, generate optimized schedules, and provide personalized recommendations. The project prioritizes data security, user-friendliness, and scalability, offering a practical solution for BAU students. However, the system's scheduling algorithm may require further optimization to handle complex constraints. [8]

Table 1:Literature Review Summary

Ref	Authors	Description	Advantages	Disadvantages
1	Li and Womer (2009)	A hybrid MILP/CP Benders Decomposition algorithm for scheduling multi-skilled personnel.	Effective for complex planning and scheduling problems.	Requires expertise in Benders Decomposition
2	Garrido and Onaindia (2010)	AI planning techniques for e-learning, integrating temporal and resource constraints.	Supports multi-criteria optimization and time/resource constraints	Practical implementation challenges not extensively discussed
3	Ajanovski (2013)	A system for creating teacher and teaching schedules based on specific requirements.	Facilitates efficient teacher scheduling	Limited information on system functionalities
4	Yang and Xie (2017)	A genetic algorithm-based approach for university course scheduling with coevolution.	Efficiently generates high-quality scheduling solutions	May struggle with local optima.
5	Hossain et al. (2019)	A simulated annealing multi-objective algorithm for university course timetabling.	Outperforms traditional genetic algorithms	Limited information on algorithm implementation
6	Tavakoli et al. (2020)	A three-stage heuristic algorithm for university course timetabling.	Improves course presentation rate	Requires further validation in other academic departments

7	Yu Chen et al. (2022)	A genetic algorithm for university class scheduling, aligning satisfaction with preferences.	Aligns satisfaction values with preferences	Limited information on algorithm implementation
8	Shaaban et al. (BAU FYP)	A web-based scheduling system using Next.js, Puppeteer, and DialogFlow for BAU students.	Automates course registration, prioritizes data security and user-friendliness	Scheduling algorithm may require further optimization for complex constraints

Applications

The SmartSchedule platform has several practical applications:

1. **University Students**: The target audience of the platform will be university students, who are in need of a course schedule manager.
2. **Academic Advisors:** The platform can be used by advisors to assist students in choosing courses and meeting academic milestones.
3. **University Administrators**: The platform can also assist administrators in overseeing course listings and enrollment statistics more efficiently.
4. **Future Integration**: The platform can be developed further to include features including academic advising chatbots, integration with learning management systems (LMS), and support for several universities.

Alternative Designs

Several alternative designs were considered during the development of SmartSchedule:

1. **Mobile App:** A mobile application version of SmartSchedule was considered to provide students with on-the-go access to their schedules. However, this was deemed less feasible due to the complexity of integrating with university systems and the need for cross-platform compatibility.
2. **Desktop Application:** A standalone desktop application was also considered, but it was rejected in favor of a web-based platform to ensure broader accessibility and ease of updates.
3. **AI-Powered Scheduling:** An advanced AI-based scheduling system was explored, but it was decided to start with a rule-based algorithm to simplify the initial implementation and ensure reliability.
4. **Integration with Existing Systems:** The option to integrate SmartSchedule with existing university systems (e.g., student portals) was considered, but this was postponed for future development due to the complexity of such integrations.

5.

References

[1] Li, H., & Womer, K. (2009). Scheduling projects with multi-skilled personnel by a hybrid MILP/CP benders decomposition algorithm. *Journal of Scheduling*, *12*(3), 281-298. doi:10.1007/s10951-008-0079-3

[2] Garrido, A., & Onaindia, E. (2010). On the application of planning and scheduling techniques to e-learning. In *Proc. 23rd Int. Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA-AIE 2010) - LNAI 6096* (pp. 244-253). Springer.

[3] Reference: Ajanovski, V. V. (2013). Integration of a course enrolment and class timetable scheduling in a student information system. International Journal of Database Management Systems, 5(3), 85-95.

[4] Yang, L., & Xie, C. (2017). Research on model of course scheduling system for ideological and political teaching in colleges and universities based on particle swarm optimization. *Revista Facultad de Ingeniería*, *32*(17), 657-663.

[5] Hossain, S. I., Akhand, M. A. H., Shuvo, M. I. R., Siddique, N., & Adeli, H. (2019). Optimization of university course scheduling problem using particle swarm optimization with selective search. *Expert Systems with Applications*, *127*, 9-24. doi:10.1016/j.eswa.2019.02.026

[6] Tavakoli, M. M., Shirouyehzad, H., Lotfi, F. H., & Najafi, S. E. (2020). Proposing a novel heuristic algorithm for university course timetabling problem with the quality of courses rendered approach; a case study. *Alexandria Engineering Journal*, *59*(5), 3355-3367. doi:10.1016/j.aej.2020.05.004

[7] Chen, Y., Ghasemi, P., Bayanati, M., Ebrahimi, M., & Khalijian, S. (2022). A novel optimization approach for educational class scheduling with considering the students and teachers' preferences. *Discrete Dynamics in Nature and Society*, *2022*, 5505631. doi:10.1155/2022/5505631

[8] Beirut Arab University FYP Project (2023). Streamlining Course Scheduling: Revolutionizing Timetabling for BAU Students.