

Project Title: ScheDool

Team Members

- Nunthatinn Veerapaiboon (Poon)
- Atchariyapat Sirijirakarnjaroen (Beam)
- Nachayada Pattaratichakonkul (May)
- Petch Suwapun (Diamond)
- Thanawin Pattanaphol (Win)

Advisor: Dr. Charnon Pattiyanon

1. Problem Statement

In the present day, several schools across Thailand face problems with timetable scheduling, especially public schools under the Office of the Basic Education Commission (OBEC) known in Thai as สำนักงานคณะกรรมการการศึกษาขั้นพื้นฐาน (สพฐ.), such as Bodindecha (Singha-Singhaseni) School. These schools require a team of no fewer than eight faculty members and up to two weeks to prepare the entire school's academic timetable of the semester.

For the case of most major Thai public schools, since there are two semesters in an academic year and more than 300 teachers to schedule, the task of creating a school-wide class schedule is an extremely complex task. Apart from being a time consuming and tedious process for the faculty, in order to create a large-scale school-wide timetable, many other factors are needed to be considered, such as, assigning teachers' teaching duties, surveying teachers' constraints, and creating a timetable that satisfies many other future constraints that may arise as the academic year goes on.

From our field surveys and interviews with the school faculty members, we found that they do have systems that assist with scheduling, however, said systems are limited in functionality and faculty members still need to spend a significant amount of their time in manually planning and rearranging the schedule. [See Figures 1, 2, and 3] This displays how current scheduling solutions that are used in some schools are not mitigating manual tasks as much as they might have been intended to be.

Realizing these problems, we decided to choose the topic of “class timetabling / scheduling” for our URD project. We hope this project will help make timetable creation easier, faster and more efficient by using AI technology in assisting with scheduling, reducing preparation time from two weeks to just several hours and reducing manual work by at least 70%. This way, faculty members can spend less time on logistics and more time on their core role as the nation's educators.

2. Beneficiaries and Stakeholders

Our project's main stakeholder is a large public high school located in Bangkok named “บดินทรเดชา (สิงห์ สิงหเสนี)” / “Bodindecha (Sing Singhaseni)”. Due to them being a large public school that still employs a semi-manual way of class timetable scheduling, we believe that, with the large set of data that our stakeholder can provide, our project will be able to help in speeding up said process along with any other administrative tasks that are included in such processes.

Additionally, as our project's goal is not to only produce a class timetable scheduling system that works with one specific school, but one that can be applied to any school in Thailand; we made a decision to collaborate with another large public high school located in Nakhon Ratchasima named “โรงเรียนสุรวิวัฒน์ มหาวิทยาลัยเทคโนโลยีสุรนารี” / “Surawiwat School, Suranaree University of Technology”, this is to make sure that the information that we receive from our stakeholders to be more diverse and balanced, leading to a more general-usage product.

With our collaboration with our stakeholders, we hope to provide our main beneficiaries, i.e. faculty team responsible for planning the class timetable for the entire school, with a product that alleviates the team's obstacles in planning through our fast, efficient and reliable timetabling system. We also hope that this project will also provide our additional beneficiaries, i.e. students, teachers, and administrators, with a clearer, more accessible and accurate timetable. Both combine to create a better learning environment for students, teaching environment for teachers, and overall management environment for school faculty and administrators.

3. Project Deliverables

Our team will deliver a **web-based scheduling system** that automates the generation and adjustment of school timetables. The system won't need to be installed on the school server, will be accessible via desktop and tablet browsers, with role-based access for administrators, teachers, and students. Administrators will be able to generate draft schedules, make adjustments, and finalize them for school-wide use. Teachers will be able to view their own schedules, as well as other teachers' and classroom schedules, and manage substitutions or class exchanges when necessary. Students will also have access to clear weekly timetables and teacher schedules to stay informed about their learning activities.

In addition to handling essential rules such as avoiding teacher or classroom conflicts, the system will also incorporate **soft constraints** that improve schedule quality and fairness. These include balancing teacher workloads across the week rather than concentrating lessons on just a few days, limiting long blocks of consecutive teaching, ensuring that complex subjects are spread evenly across the week for students, and respecting teacher preferences for certain times of day. The system will also aim to minimize unnecessary travel between buildings by grouping lessons logically whenever possible.

A key goal of the system is its ability to adapt to the **diverse constraints of different schools and curricula**. Schools may have unique requirements, such as fixed periods for special activities, team-teaching sessions, subject-specific room allocations, or visiting lecturers with restricted availability. The system is designed to accommodate such variations so it can serve not just a single school, but a wide range of educational institutions with differing curricular structures.

The platform will include **export** options to standard formats (CSV, Excel) so schools can easily integrate the timetable with their existing management systems. This ensures that the system can enhance operations without requiring major changes to existing workflows.

We intend to develop this system as a **Software-as-a-Service (SaaS)** platform, and in the long term, we plan to start scaling this system through the Thai Government High School, then expand to other types of schools and later establish a new standard for Thai educational system

4. Project Approach

This project will follow a structured four-step approach. First, we will **understand the problem** by conducting field visits and interviews with teachers and staff involved in scheduling, to gain a detailed view of current practices, challenges, and pain points.

Next, we will synthesize these insights and **explore potential technologies**. Candidate methods under consideration include Reinforcement Learning, Genetic Algorithms, Constraint Satisfaction Problems, and Heuristic Search. Each method will be evaluated for its ability to meet the project's criteria, including flexibility, scalability, and efficiency in handling complex, dynamic timetables.

Based on the selected approach, we will **design the system architecture** and **develop a prototype** to demonstrate core scheduling functionality. The prototype will be tested with real users to validate usability and effectiveness, and adjustments will be made as needed before progressing to a full system implementation.

Finally, after refining the prototype based on user feedback and testing results, we'll roll out the complete system. We'll keep monitoring and evaluating it to make sure it meets the project's objectives, cuts down scheduling time, reduces manual workload, and genuinely helps teachers manage their timetables more easily.

5. Project Objectives for Fall 2025

Our project objectives for this semester is essentially a “Minimum Viable Product” or MVP of our project, consisting of three main pillars: A.I. scheduling engine, Frontend Development, and Backend Development,

Before starting our project, we need to **gather requirements** from our stakeholders on a variety of topics, e.g. variations of class timetables, soft constraints in creating or scheduling such timetables, specific personal preferences from the teaching faculty and more. In the end, we will summarize our findings by listing out said requirements and going through several discussions in order to rank on which requirements we shall prioritize when building our system; this process is done for us to be able to plan a proper roadmap on developing our system along with minimizing any obstacles that may occur during our development stage.

After retrieving our constraints and requirements, we will proceed to start the process of **developing our A.I.** scheduling engine, we have started our research stage on the best solution in solving the problem as we have outlined in our “Project Approach” section in which we will go through a series of discussions and have a final decision on what would be the way for us to proceed with the development process. This is done in order for us to critically examine each available option and consider which solution is best suited for our project.

Along with the development of our A.I. engine, we plan to start on planning out our **frontend**, i.e. the user interface and the technology stack. We want to make sure that our user interface will be easily accessible, understandable and intuitive for our users to reduce as many obstacles that they may encounter during real-life usage. On the other hand, we will also start our **backend** development process in which our goal is to create a backend system that is fast, efficient, able to handle a significant amount of requests from the frontend and is able to communicate effectively with the database.

In summary, as we have described above, our goal is to complete the three main pillars of our project and to form a minimum viable product (MVP) that is able to provide our stakeholders with core functionalities of the system, e.g. class scheduling, timetable creation, etc. With our MVP finished, we can continue to add additional features and optimize our system even further.

6. Task Breakdown for Fall 2025

We have divided the project into key tasks to clearly organize development and ensure steady progress. Each task focuses on a specific aspect of the system, from requirement gathering and AI research to frontend and backend development, testing, and deployment.

Collecting Requirement & User Surveys

We will gather detailed requirements by interviewing personnel involved in scheduling and observing their current workflows. This includes visiting schools to understand the existing processes, challenges, and constraints firsthand. We will also focus on maintaining strong, ongoing relationships with stakeholders, ensuring continuous feedback and alignment throughout the project.

AI Model / Algorithms Research

We are currently working on research algorithms and AI training techniques related to scheduling problem including:

- Genetic Algorithms
- Constraint Satisfaction Problems
- Heuristic Search
- Reinforcement Learning

We are examining which methods may be most suitable for our system and considering the possibility of combining approaches into a hybrid solution

AI Scheduling system

Besides selecting an AI scheduling method, we need to design a structured dataset as input for the AI, based on the data provided by stakeholders. We will then proceed to the training process, starting with schedules that only include basic hard constraints and gradually increasing the size and complexity of the dataset. Once the AI can reliably handle hard constraints, we will introduce soft constraints and optimization objectives to improve the quality of the schedule and maximize user satisfaction.

UI/UX Design

We will define how the product should feel and function for users, focusing on clarity, ease of use, and efficiency. A user flow will be created to match user goals and tasks, ensuring smooth navigation through key features such as timetable generation, schedule review, and adjustments. The interactive user interface will be designed in Figma, and feedback will be gathered from users to refine and improve the design before implementation.

Architecture Design

We will select a framework and technology stacks that supports the user journey and meets system requirements, ensuring an intuitive interface for administrators, teachers, and

students. The data flow will be designed to clearly define how the frontend, backend, AI engine, and database are communicated, with modularity and scalability.

Frontend Development

We will build the interface of the scheduling system to make it easy and intuitive for administrators, teachers, and students to use. Users will be able to view and manage timetables, interact with AI-generated schedules, handle substitutions, and export schedules seamlessly. The design will focus on clarity, smooth navigation, and responsiveness, so that performing tasks feels natural and straightforward on both desktop and tablet devices. The frontend will also closely follow our Figma designs, incorporating user feedback to continuously improve the experience.

Backend Development

We will build the backend to power the scheduling system, managing user accounts, role-based access, and all scheduling logic. We will design the database to securely store user data so that each person can only access the schedules and information relevant to their role, as well as the generated timetables. We will also create APIs to connect the frontend and backend, ensuring smooth data flow and integration with school management systems while keeping data secure and consistent.

Testing & Deploying & Evaluation

We will start by deploying and testing the system in a controlled development environment, where the team can safely identify and fix any issues. Once the system is stable, we will release a demo version in a real environment, focusing on core functionality and observing how it performs under actual conditions. During this stage, we will gather feedback from administrators, teachers, and students to understand usability, effectiveness, and satisfaction, and use this feedback to refine and improve the final system.

7. References

- Bonetta, G., Zago, D., Cancelliere, R., & Grosso, A. (2023). *Job Shop Scheduling via Deep Reinforcement Learning: A Sequence to Sequence approach* (No. arXiv:2308.01797). arXiv. <https://doi.org/10.48550/arXiv.2308.01797>
- El-Sakka, T. (n.d.). (PDF) *University Course Timetable using Constraint Satisfaction and Optimization*. ResearchGate. Retrieved September 8, 2025, from https://www.researchgate.net/publication/296805727_University_Course_Timetable_using_Constraint_Satisfaction_and_Optimization
- Hambali, A. M., Olasupo, Y. A., & Dalhatu, M. (2020). Automated university lecture timetable using Heuristic Approach. *Nigerian Journal of Technology*, 39(1), 1–14. <https://doi.org/10.4314/njt.v39i1.1>
- Müller, T. (n.d.). *Constraint-based Timetabling*.
- Thakare, S., Nikam, T., & Patil, M. (2025). (PDF) Automated Timetable Generation using Genetic Algorithm. *ResearchGate*. <https://doi.org/10.17577/IJERTV9IS070568>
- Wright, M. (2025). (PDF) School Timetabling Using Heuristic Search. *ResearchGate*. <https://doi.org/10.1057/jors.1996.34>
- Zhang, L., & Lau, S. K. (n.d.). (PDF) Constructing university timetable using constraint satisfaction programming approach. *ResearchGate*. <https://doi.org/10.1109/CIMCA.2005.1631445>