



Namal University, Mianwali

Department of Computer Science

**Project Proposal for
University Timetable Management System**

CSC-225: Software Engineering

Fall 2025

Requirement Provider (Client):

Abdul Rafay

Submitted By:

Sr. No	Name	Reg. No
1	Hammad Shabir	NUM-BSCS-2024-25
2	Ahmer Sultan	NUM-BSCS-2024-03
3	Husnain Ali	NUM-BSCS-2024-26

Submission Date: 09 November 2025

Contents

1	Introduction	2
2	Problem Statement	2
3	Project Objectives	2
4	Stakeholder Identification	3
5	Software Development Methodology	4
6	Tools and Technologies	5
7	References	5

1 Introduction

In any education system, time management is very important for both teachers and students. A managed schedule helps everyone stay organized and focused. It is important to balance study time, free slots, and other activities. When a timetable is clear, well planned, and clash-free, students can easily follow their routines and teachers can take their classes without confusion. It defines the schedule of classes, allocates venues, assigns instructors, and ensures that academic activities are processed in an organized manner. Therefore, a proper, efficient, and effective timetable is required to support smooth learning process in the university. The proposed **University Timetable Management System** aims to provide a centralized platform where clash-free timetable will be ensured. It will provide convenience to everyone involved in timetable managing and scheduling process. Moreover, the system will allow end-users (Students) to view their classes timetable in a mannered way.

2 Problem Statement

The current traditional way of managing and creating university timetables often causes problems like overlapping classes, venues being booked for more than one class simultaneously. This increases inefficiency when there are multiple departments which is a problem in making sure teachers, students, and classrooms are all available at the same time. In addition to that, makeup classes and checking free time slots are done informally through Whatsapp groups by Class Representatives (CR), which sometime leads to delays and conflicts. This makes it harder for students to follow their schedules and for teachers to plan their work. Moreover, timetable is shared through mails, so when there is some conflict, it is resolved and shared again which makes accessing it harder when there are multiple schedules have been shared. Hence, there is a clear need for a proper centralized timetable system that can handle multiple departments, identify conflicts so that can be resolved early, and make education easier.

3 Project Objectives

The major objective of this project is to implement a management system that will provide a platform for creating and managing class schedules and timetables effectively for faculty and students of the university. This project aims to replace the typical paper-based system and provide a centralized digital solution.

The key objectives are:

- To provide a platform where administrators can conveniently create, update, and

manage timetables for all departments and courses.

- To help administrators in finding and resolving clashes between classes, instructors, and class venues.
- To implement role-based access for administrators, faculty, and students.
- To design a user-friendly web interface that provides ease of use for all users.
- To allow students and faculty to view updated schedules through dashboards online whenever changes occur.
- To store previous timetable records for future reference.

4 Stakeholder Identification

The development and implementation of the **University Timetable Management System** involve multiple stakeholders, where everyone has their own role, responsibilities. The identified stakeholders are as follows:

- **University Administration:** Responsible for checking overall system deployment and making sure that the system aligns with organization's policies. They also provide resources and approval for implementation of the system.
- **System Administrator (Admin):** The admin is responsible for creating, updating, and maintaining the timetables for all departments and resolving scheduling conflicts. Admin is also responsible for creating and deleting user accounts.
- **Faculty Members:** Faculty members will use the system to view their teaching schedules, assigned venues, and timings.
- **Students:** These are the end-users who will access the timetable to view class schedules, timings, and assigned venue details. The system makes sure that students always have access to the most updated timetable.
- **Course Coordinators:** Help admin by providing course details and requirements of department. They play a major role in ensuring that timetable data remains accurate without any conflict between classes.
- **ITSC / IT Support:** Oversees labs allocation and their schedule. IT admins also assist in scheduling as they are responsible for labs and their usage.

- **Requirement Provider (Client):** Provides project requirements, feedback and continuously communicates during the development process to ensure that the system meets the university's needs and requirements.

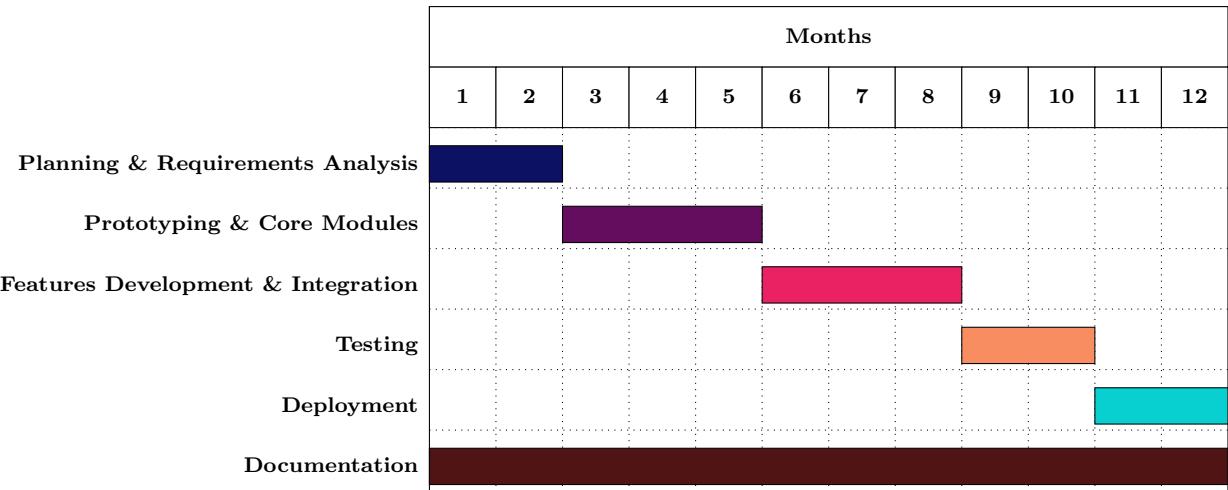
5 Software Development Methodology

For this project, we will use the **Agile (Scrum)** development process. The reason for choosing Scrum is that this project has many constraints due to the multiple roles involved in it, and we need to have continuous communication with Client/RP. Even after extensive requirement analysis, requirements might evolve, Client may want more features, or might not features already developed. So, having an iterative approach is beneficial for effective development and Scrum will provide us a facility to accommodate these changes. In Scrum, system is divided into sprints where each sprints is communicated with Client, feedback is noticed and changes are applied which in result rectifies the overall system. Continuous involvement of stakeholders shall assure us that developed system meets user requirements and needs and issues are identified early which makes development effective. According to one year development schedule, project will be divided into phases as **Scrum sprints. Each sprint spans one month**, hence there are 12 sprints such as:

Phase	Activities
Planning & Requirement Analysis	Requirement gathering, defining project backlog, and sprint planning.
Prototype & Core Modules	Develop core modules (user authentication, database setup, basic CRUD operations), front-end interface, and first testing.
Feature Development & Integration	Develop additional features (advanced modules) and integrate modules, do unit and integration testing.
Testing & Refinement	Perform system testing, bug fixing, perform refinement based on user feedback.
Deployment	Final deployment and user training.

Table 1: Project Phases

Timeline - Gantt Chart



6 Tools and Technologies

- **Backend:**
 - MySQL or PostgreSQL (if advanced features are required)
- **Programming Languages:**
 - Node.js for server-side scripting
 - HTML, CSS, JavaScript (or any framework like React/Angular)
- **Web Server:**
 - Apache or Nginx (if better performance is needed with high traffic)
- **Design & Prototype:**
 - Figma

7 References

- [1] B. Joy, “Software Engineering Project: Robo Hatch,” *GitHub Repository*, 2021. [Online]. Available: <https://github.com/basudebjoy/Software-Engineering-Project-Robo-Hatch/>. [Accessed: Nov. 8, 2025].
- [2] University of Washington, “Beer Recommendation System Project Proposal,” *CSE403: Software Engineering Course*, 2012. [Online]. Available: <https://courses.cs.washington.edu/courses/cse403/12sp/Projects/proposals/brdmstr-proposal.pdf>. [Accessed: Nov. 8, 2025].

- [3] H. Techie-Menson and P. Nyagorme, “Design and Implementation of a Web-Based Timetable System for Higher Education Institutions,” *Int. J. Comput. Appl.*, vol. 7, pp. 1–13, 2021.
- [4] OpenAI, “ChatGPT (GPT-5 Mini) Prompt Responses,” 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [5] AI prompt, “make a project proposal template in latex as described in given document,” OpenAI ChatGPT (GPT-5 Mini), Nov. 8, 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [6] AI prompt, “look at the content of title page, like is it grammatically correct and punctuations too?,” OpenAI ChatGPT (GPT-5 Mini), Nov. 8, 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [7] AI prompt, “how to add ieee style references in latex?,” OpenAI ChatGPT (GPT-5 Mini), Nov. 8, 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [8] AI prompt, “help me with identifying relevant tools and technology for this project,” OpenAI ChatGPT (GPT-5 Mini), Nov. 8, 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [9] AI prompt, “which software development methodology should we use for our project, what i think is that this system is complex and faces lot of constraints, so we should utilize iterative methods like scrum and kanban which focuses on continuous feedback and evaluation?,” OpenAI ChatGPT (GPT-5 Mini), Nov. 8, 2025. [Online]. Available: <https://chat.openai.com/>. [Accessed: Nov. 8, 2025].
- [10] AI prompt, “can you make gantt chart for schedule given below?,” Claude (Sonnet 4.5), Nov. 8, 2025. [Online]. Available: <https://claude.ai/new/>. [Accessed: Nov. 8, 2025].