Software Requirements Specification for

Course Planner for Faculty

Version 1.0 approved

Prepared by Group 9

IIT JODHPUR

05/03/25

[1] *Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.*

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
+			

1. Introduction

1.1 Purpose

This project aims to build software for courses of faculty members. It will help the faculty to organize and give their time schedule.

1.2 Document Conventions

Every requirement statement has its own priority.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

- ***Faculty Members** – To understand how the system organizes their course schedules*
- ***Academic Administrators** – To manage faculty-course allocations*
- ***Software Developers** – To implement and maintain the system*
- ***Testers** – To validate requirements and system functionality*

1.4 Product Scope

The Course Planner will provide a structured platform for faculty to design, organize, and manage course schedules, content, and instructional strategies. It will include features for syllabus creation, lesson planning and help them to manage their daily schedule. The planner will ensure alignment with academic goals, optimize workload distribution, and enhance time management. Designed for ease of use, it will support digital accessibility, collaboration, and continuous course improvement, ultimately fostering a more efficient and well-structured academic environment.

1.5 References

- IEEE template

2. Overall Description

2.1 Product Perspective

The Planner is a faculty-centric tool designed to enhance academic planning by course structuring, scheduling and schedule management. By replacing manual planning methods with a digital, user-friendly interface, it improves efficiency and alignment with academic standards. The planner supports faculty in scheduling the events of their day.

2.2 Product Functions

Weekly Schedule View: Visualize and adjust weekly course schedules.

Data Storage: Securely store course and schedule data.

User Management: Create and manage user accounts and preferences

Change Password: User can change password

Added "Export Schedule as PDF" feature for faculty to download schedules.

Improved "Get Available Slots" feature for better accuracy.

High-Level Function Groupings:

Course Management: Add/edit courses, conflict detection

Schedule Management: Allocate time slots, adjust weekly view.

User Interaction: Account creation and preferences

2.3 User Classes and Characteristics

User classes are:

- ***Faculty of IITJ***

Characterstics:

F

2.4 Operating Environment

OPERATING SYSTEMS

Operating system	**Compatible editions**
Windows 11, Update (version 24H2) and newer	**64-bit**
Windows 11, Update (version 23H2)	**64-bit**
Windows 11, Update (version 22H2)	**64-bit**
Windows 11	**64-bit**

Operating system	Compatible editions
Windows 10 2022 Update (version 22H2)	**32-bit and 64-bit**
Windows 10 November 2021 Update (version 21H2)	**32-bit and 64-bit**

	Requirement
Processor	*1 GHz*
RAM	*512 MB*
Minimum disk space (32-bit)	*4.5 GB*
Minimum disk space (64-bit)	*4.5 GB*

HARDWARE REQUIREMENTS

2.5 Design and Implementation Constraints

Implementation constraints include:-

1) *Memory limitations*
2) *Weak Security*
3) *Hardware constraints (not having a dedicated server)*

Minor optimizations to improve performance on low-end devices.

2.6 User Documentation

The following user documentation components will be delivered along with the **Course Planner for Faculty** software to assist users in understanding and using the system efficiently.

User manuals:

A comprehensive **User Guide** detailing how to navigate and use the Course Planner.

Step-by-step instructions on **creating, modifying, and managing course schedules**.

Explanation of system features such as **conflict resolution, availability checking, and reporting tools**.

Online Help System:
Integrated **Help Section** within the software, providing context-sensitive guidance.

Searchable FAQs and troubleshooting steps.

Quick Start Guide:

A **one-page summary** covering essential steps to begin using the software.

Provided in **PDF format** and included in the software installation package.

2.7 Assumptions and Dependencies

The following assumptions have been made during the development of the Course Planner for Faculty system.

User Access & Roles:Faculty members and administrators will have **distinct roles and permissions** within the system.Only authorized users will be able to modify course schedules and assign faculty

User training:Faculty and administrators will undergo **basic training** before using the system.Online help and tutorials will be sufficient for most users to understand system functionalities.

Internet connectivity:The software will be **web-based**, requiring a stable internet connection for real-time updates.If deployed as an offline application, local storage and periodic synchronization mechanisms will be required.

3. External Interface Requirements

3.1 User Interfaces

Screen layout will be 1920 X 1080 . Standard buttons and functions are applicable such as return . Error messages will be displayed on new windows. Software components needed are Node js, MySQL ,XAMPP and Electron. A Database will be employed to store the list of courses. My SQL database has been used to update and store the course list.

New Button: "Export as PDF" added to the UI.

Minor UI refinements for better readability.

3.2 Hardware Interfaces

Supported device types are Windows and Linux Users. Nature of data is mostly strings. Communication protocols to be used are TCP/IP.

3.3 Software Interfaces

Software Interfaces include connections between MySQL and PHP server for processing data about the User ID and Login. We are using MySQL version 8.0 and PHP version 8.4.3 and Node.js Version 22.13.1.We are also using the Electron library from Node.js. Data items coming into the system are the User Id and Password and coming out of the system is a truth value whether the user Exists or not. The services needed are PHP server and a MySQL connector. Nature of connection is TCP/IP.The data that will be shared across software components are the course list and slots.Updated MySQL version to 8.1 for improved stability.

3.4 Communications Interfaces

HTTP will be used to send the data collected to the web server . This will be used to get the data from the user and validate it from the SQL database.

4. System Features

4.1 Add Course/ Drop Course

4.1.1 Description and Priority

The feature will allow the user to add/drop courses to modify his timetable. It is of high priority as it is essential to register the User's Courses and make his timetable.

4.1.2 Stimulus/Response Sequences

User logs in the system and then clicks the option to add a course.

4.1.3 Functional Requirements

Functional requirements include:-

HTTP Browser

Product will respond to invalid inputs via a dialog box.

REQ-1: Browser
REQ-2: TBD

4.2 Get Available

4.2.1 Description and Priority
The feature will allow user to get the slots that he is free today. It's priority is medium.
4.2.2 Stimulus/Response Sequences
User Logs into the system and clicks the Get Available button.
4.2.3 Functional Requirements
4.2.4improved algorithm for more accurate availability results.

4.3 Edit course

5. Other Nonfunctional Requirements

5.1 Performance Requirements

None

5.2 Safety Requirements

Never share your password with another person.

Keep case sensitivity in mind

5.3 Security Requirements

added basic encryption for faculty login credentials.

5.4 Software Quality Attributes

- Adaptability
- Availability
- correctness

5.5 Business Rules

Access based on roles:

- Schedules are proposed by faculty, subject to approval by the department.
- Before submitting schedules, department heads review and adjust them.
- It is the administrators' responsibility to finalize schedules and allocate resources.

Constraints on scheduling:

- No overlapping faculty time slots.
- Prerequisite and corequisite rules must be enforced.
- Faculty can teach up to X courses per semester.

Allocation of resources:

- Assigning classrooms takes into account the size of the class and the equipment needed.
- The priority is given to mandatory courses over electives.

Conflict Resolution:

- Seniority and specialization are considered when resolving faculty scheduling conflicts.

6. Other Requirements:

TBD

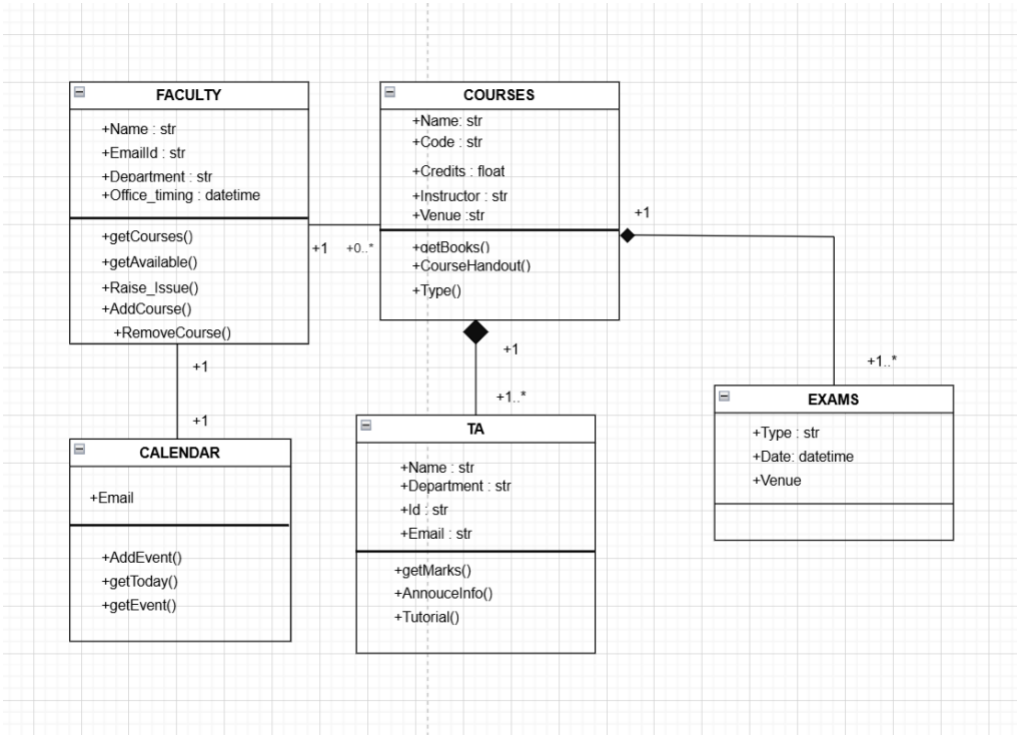
<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

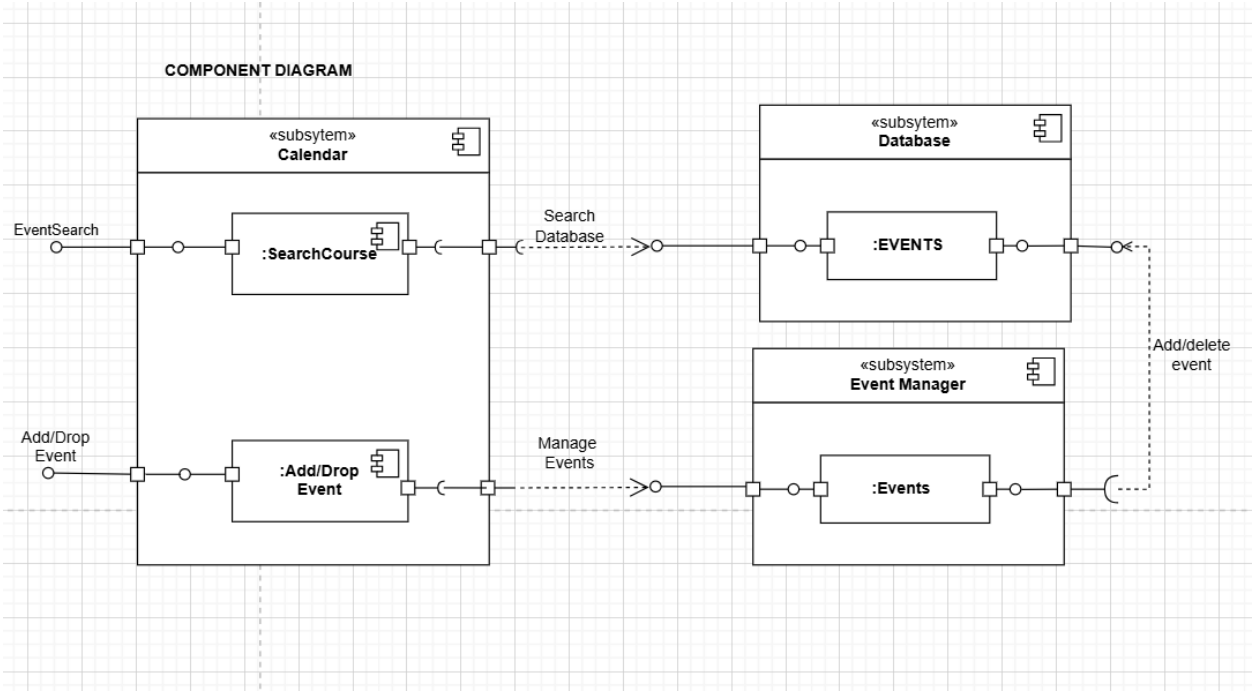
<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

CLASS DIAGRAM



COMPONENT DIAGRAM



Appendix C: To Be Determined List

- *Functions*
- *User Interface*
- *Database*