

Team 11's Sprint 1 Requirements Artifact for Schedule Builder - Addison Bartelli, Anya Combs, Janna Dungao, Hunter Long, Elizabeth Miller, Marco Martinez Reyez

1. System Overview

The schedule builder will provide an intuitive user interface that allows users to build schedules at the University of Kansas for past, current, or future semesters. It will allow users to save built schedules and edit the saved schedules. It will also prevent users from adding courses with conflicting meeting times.

2. Functional Requirements

As a user, I want a display page that shows records of past courses.

2.1. Display Page – Elizabeth

- FR1.1: The system will display semesters with available data in tabs.
- FR1.2: The user can click on the displayed semesters.
- FR1.3: The user can click on saved schedules for the same semester for comparison.

As a user, I want to see relevant course information for the course I searched.

2.2. Display Course Meeting Times

- FR2.1: The system will display the times a course meets in a week on a calendar with the time ranging from 7:00 AM to 8:00 PM.
- FR2.2: The system will display the course code on each block on the calendar.

As a user, I want the application to have a database storing course information.

2.3. Setup Database to Store Course Information - Elizabeth

- FR3.1: The system will hold course information in a database, which is currently only hardcoded.

As a user, I want the course database to store relevant information.

2.4. Add Course Names, Credit Hours, Descriptions, and Prerequisites to the Database - Elizabeth

- FR4.1: Currently, the schedule builder displays the course name on its block on the weekly calendar.

As a user, I want to be able to switch between semesters.

2.5. Add Tabs to Choose a Semester to Schedule – Elizabeth

- FR5.1: The system provides tabs to switch between currently hardcoded schedules that are saved in the code: Fall 2024, Spring 2025, and Fall 2025.

3. Nonfunctional Requirements

As a developer, I want to create a development environment with Next and TypeScript installed and configured

3.1. Environment Setup -- Addison

- NR1.1: The project will be defined by a package.json outlining all dependencies.
- NR1.2: The project will have a file structure separating the source and built files.