Project Assignment 11
Updated Interaction Design Documents

Stakeholders and Users

- Head of the CS department (Dr. Ott)
    - Primary
    - Expert Users
- Whoever is making the Schedule
    - Primary
    - Expert Users
- Faculty/Staff/TAs
    - Secondary
    - Novice Users

Personas

- Available on UX consultant design support documentation

Nominal Use Scenario

- An administrator navigates to the app to add a course. The admin selects the semester to view. They choose to view the schedule on a calendar view. They then look for a spot in the calendar to add a 3rd-year course and add the course, filling in major details including course name, number, section, instructor, and room number. The admin checks the calendar view and notices that their course has been added and contains no conflicts.

- A Teacher's Assistant navigates to the app to view all 3rd year CS courses. The TA selects a semester to view. The TA chooses to view the semester in a list view. Within the list view, the TA applies a filter to only view 3000-level courses. The filter is applied and only 3000 level courses are shown within the list view. The TA can switch to the calendar view to see all 3000-level courses on a calendar.

- An administrator who wishes to generate a new schedule navigates to the app. Within the app, the admin selects the desired semester from a dropdown list of semesters, for example, Fall 2024. Upon selecting Fall 2024, the admin views a 'notes' section that contains helpful notes from prior schedule creation semesters that give tips on building the schedule. Using this information, the admin updates course constraints. The course constraints may notify the admin of any restrictions with the current schedule. If present, the admin can go to a list or calendar view of the courses and select a course to modify by updating its time, location, or other information. Upon making course changes, the course constraints find that the schedule meets all requirements. From here, the admin can choose to add more courses or use the schedule as built for its intended semester.

- An administrator navigates to the app to add a course section. They fill out all necessary information about the section, but fill in the start time as 2:00pm and end time at 1:50pm. The app recognizes that the allotted time for the course does not work since the end time is before the start time. The app warns the user about the error adding the section and does not add it to the database.

Use Environment

- Any user must have access to a computer and internet connection in order to use the app. The app's website is also available on mobile but is not built to be used as such. The most common environment would be on campus in an office or workspace, as it would see the most use time from administrators building the schedule.

Hierarchical Task Analysis

Enter the CS Scheduling App
- Login as an editor
  - Choose a semester
    - Create schedule
    - Search for courses
    - Add a course
      - Add critical details about course
    - View constraints
      - Add a constraint
        - Choose what to constrain and apply
    - View notes
      - Add a note
  - Choose a view style (calendar vs. list)
- Login as a viewer
  - Choose a semester
    - Search for courses
    - View constraints
    - View notes
  - Choose a view style (calendar vs. list)

Database Schema

List of Domain Classes:

- Department - list of departments
- Instructor - list of instructors
- Course - list of courses
- Room - list of rooms
- Section - The specific section for a course

Domain Class: Instructor - Instructors teach classes and belong to a department

- Instructor_ID - int, The instructor ID
- Department - department, foreign key into department table
- Name - The name of the instructor

Domain Class: Department - Departments offer a set of courses and have a set of professors that belong to it

- Name - The name of the department
- Courses - course, foreign key into course table
- Instructors - Instructor, foreign key into professor table

Domain Class: Course - A course offered by a department

- Course_ID - String, The course ID
- Name - String, The name of the course

Domain Class: Room - A room available for classes

- Room_ID - int, The course ID
- Building - int, The building number of the room
- Room_num - string, The building room number (G005)
- Class_Type - String Array, The type of classroom (Computer lab, group spaces, etc.)
- Availability - Time array, weekly room availability for scheduling courses

Domain Class: Section - The section for a course

- CRN - String, The CRN of the course
- Course_ID - Course, foreign key to course
- Instructor_ID - Instructor, foreign key to instructor
- Room_ID - Room, foreign key to room
- Time - Time, The time during which the class starts (9:30am, 1:30pm, etc.)
- Dates - String, The weekdays the class takes place (MWF, TR, etc.)
- Duration - Int, The duration of the course
- Year - Int, The year the course takes place
- Semester - String, The semester the course is taught (Spring, Summer, Fall)