Team:

sdmay24-42 Class Scheduler [Team 2]

Members:

Ryan Tullis
Dillon Gesy
Nicolas Figueroa Calderas
Charles Hosier
Abdullahi Abdullahi

Advisor:

Thomas Daniels

Client:

Vicky Thorland-Oster
Tina Prouty

Notable Absences:

N/A

Kickoff meeting on January 19th, 2024:

Agenda:

- Discuss where everyone was at in the project
  - Ryan set up a very early prototype for our scheduler.

- Discuss what needed to be done and by who
  - Create issues for people to do in order to participate in the project.
  - Everyone needs to play with the Visual Studio IDE, .NET MAUI project in order to get comfortable in developing our project using it.

- Discuss what needed to be changed from our initial design
    - Change our tech stack - we decided to move away from Eclipse and Google Web Toolkit. Instead, we chose to move forward with a .NET MAUI project. The database, .SQLITE, remains the same.
- Discuss what needed to be changed about the group from 491
    - Meetings between members, the advisor, and the client need to occur more frequently. We did not meet enough in 491. This class will be even more difficult if we keep the 491 meeting structure.
- Discuss what needed to be done following the faculty meeting in 491
    - We need to define our minimum product for the client.
    - We also need to create openings for future developers to add more than the minimum product.
    - Edit our design document

Meeting on January 24, 2024, at 4:15pm - 5:00pm with client:
Email sent to request a meeting: January 22, 12:50pm
Meeting confirmed: January 22, 1:40pm
Summary of points:

The main point of this meeting was to work with the client to get actual requirements for our project. The following are requirements listed in priority order

- **Requirement #1:** View/edit past schedules
    - Let our client be able to view past year’s schedules with our program to cross reference when making their schedules
    - Let our client be able to import a past year’s schedule with our program in order to create a new schedule
- **Requirement #2:** Filter visible classes (basic)
    - Let our client be able to filter the visualizer by multiple things:
        - Classes
        - Colleges
        - Labs, Recitations
        - Days

- **Requirement #3:** Exportation
    - Give our client a compiled list of data of what they scheduled using our program
        - For example, exporting to Excel each class and its data, such as when it was scheduled, section number, professor, etc.
- **Requirement #4:** Importation Architecture
    - Allow our client to import schedule data
        - For example, being able to import data from Fall 2024. We will not have this data as we will be graduated, so our user needs to have a way to import it.
    - Allow our client to import new class times data
        - For example, classes in the past semester have started at 8:00. Now, they can start at 7:45.
    - Allow our client to import degree flowcharts
        - Being able to import degree flowcharts will help with conflict checking in Requirement #5
- **Requirement #5:** Warnings
    - Conflict check, through the flowcharts, whether or not a class can be scheduled at a time.

Meeting on January 24, 2024, at 9:00am - 10:00am with Advisor:
Email sent to request a meeting: January 22, 1:00pm
Meeting confirmed: January 22, 1:40pm
Summary of points:

- Set up a view model for the classes
    - Some things to consider are start times/end times
    - Consider what constraints are constant and what can be changed

- The direction we are going in is good - but the more we plan the easier this will be
    - Consider finding where the least overlap-y classes are to make scheduling easier
- Get to coding and get our data set up to be accessed

Meeting on January 26, 2024:

- Discuss our milestones for 492
    - Full data modeling
    - Displaying all courses visually
    - Filtering
    - Exporting/Importing
    - Testing
- Discuss where everyone is at in the project
    - Getting used to the framework for the project so we can execute milestones effectively
- Discuss the next steps
    - Work to set up data modeling and our backend
    - Work with the framework to help connect the backend to the frontend