Group Members:
Eunsoo Roh (roh7)
Josh Halstead (halstea2)
Tom Bogue (tbogue2)
Joe Ciurej (ciurej2)
Kyle Nusbaum (kjnusba2)

CS 411 Project Report

Project:

We built a browser-based web application to help students deal with turbulence in academic deadlines.

Many generic class scheduling solutions are available online. However, none of them offer tools to manage deadlines as ours does. Our site gives an easy way to view all your upcoming deadlines so you can plan to work on them accordingly. For large classes, these deadlines will automatically appear once enough people have added them, so you won't miss out on deadlines you weren't aware of. Furthermore, we offer a meeting scheduler to schedule group meetings at times where everyone in the group is available, so they can work together on a deadline (working on a project, studying for an exam, and all the rest.)

Functionality:

Data:

Advanced Features:

Our first advanced feature is what we call the deadline aggregation.

Our second feature is the group meeting scheduling. The meeting scheduler allows a private group to find a time to work together on a deadline by giving a list of times when everyone that needs to make the meeting is free, and allowing the user to choose from these times and send an email to each group member informing of the meeting. This turned out to be very challenging, involving many different components of the project. The availability of each group member is based on their Google Calendars, so we had to work with the Google API to get this information. Users our asked to give our app read access to their Google Calendars, and this access is stored with the user. The scheduling itself uses an algorithm we wrote to determine times where everyone is free, taking into account the meeting duration, everyone's available times, and overnight "off-limits times" the user chooses (so that meetings are not scheduled for silly times like three in the morning.) The available times are used to populate a modal, the user can choose one and give an email message, and we craft and send the email to all the group members in the meeting.

Challenges and Plan Changes:

Labor Division:

Eunsoo:

- Database creation and server maintaining.
- Development of abstraction layer of the database. This is code that allows us to interact with the database without directly constructing SQL queries everywhere.

Josh:

- Scraping course data from the UIUC course catalog.
- Major work on the request handlers which bridge the front end to the back end.

Kyle:

- Also worked on the abstraction layer of the database.
- Also worked on the request handlers.
- Wrote user email verification and scheduler email sending.

Joe:

- Designed and created the website's front-end pages.
- Also worked on the request handlers.

Tom:

- Wrote scheduling and aggregation algorithms.
- Google API authentication and integration.