

Proposal - Due 11/1/2021:

- A description of your application's functionality and motivation – What will it do and why would someone want something that does that. (250 words)
- An outline of how your proposed project will meet the minimum project requirements.
- An outline of the ways your project will go beyond the minimum requirements.
- Assignments of who plans work on what parts of the application. (This is not necessarily final. Often you will find that some pieces are more or less difficult than you originally imagined.)
- The link to the GitHub repository (or other hosting space) you will be using as you build your application.

Minimum Requirements:

HTML + CSS

Python-Flask

WTForms

Flask-SQLAlchemy

Flask-Login

- Allow users to create accounts and store appropriately hashed passwords. This includes running your server over HTTPS.
- Accept user data and store it in your own database that you have designed and created.
- Provide that data in a meaningful way through a well-formatted (and hopefully attractive) web interface.
- Serve a reasonable purpose. (You are not permitted to create a trivial application that nominally meets all requirements, but has no real purpose.)

To get an A:

The particular application chosen by your group likely invites the use of many additional technologies. Some extra features may simply make more complex use of required technologies.

e.g.

- RESTful API and Ajax
 - OAuth (client/server)
 - Email confirmation on accounts for password reset
 - Multi-factor Authentication
 - Cloud Hosting
 - Live Multi-User Interaction
-

UR DINE AND WHINE

With the current fiasco of the dining services as well as an ongoing outcry of bad food, raw food, and sometimes expired products, we have identified a problem that needs to be addressed on campus. For this project, we have decided to create a website as a solution to this problem - ***UR Dine and Whine***, that allows students and faculty on this campus to review the various dining areas (with static menus, i.e. Connections/Grab & Go, etc.) and the items on their menus. This would be useful in providing feedback to the dining staff of which stations need work done to improve the quality of food while also giving consumers information about what they are eating and what's the best item on the menu. In addition, this may also incentivize dropping items from the menu or increasing the inventory of certain popular items so they do not run out.

To meet the minimum requirements, users will have to make an account if they want to leave reviews on food items at the different dining establishments. Otherwise, they can just view other people's reviews. Users who have made accounts will be able to edit their information and have it updated if saved properly. All user data will be stored in a database alongside their comments.

We also have a plan to have a commenting system so that users can interact with each other, a live multi-user technology we hope to integrate into the site. Another feature that we aim to have is the ability to change and reset users' passwords with email confirmation (using smtplib). In addition, we might also web-scraping (through Selenium or BS4) the dining center webpage for Douglass Dining Center to actually include a dynamically changing menu if time permits.

Assignment of Responsibilities:

- Database - Uzair Tahamid Siam
- Styling - Jacqueline Moon
- Login/Registration - Uzair Tahamid Siam
- HTML templating - Jacqueline Moon
- Others - TBD

To summarize, most back-end work will be done by Uzair Tahamid Siam and the front-end will be done by Jacqueline Moon. Of course, there are bound to be overlaps but this is the high-level plan.

Link to Github [repository](#)