Project Features List:

1. Website
    a. Web app for the service to run on. Includes database, frontend, and backend.
2. Course Scheduler:
    a. Build semester course schedule based on required courses for major and preferred times and preferred amount of credits
3. ‘Rate my Professor’ with FCQs
    a. FCQ data (student input) if possible
4. Degree Audits
    a. Collect information about taken courses, compare with requirements for degrees, recommend next step classes
5. Database
    a. Storing user info, store login information, courses, FCQ data, etc.
6. Profile Pages
    a. Courses, clubs, major, year, where your from.
        i. Autocomplete course input with course data available

Requirements:

1. Website:
    a. Functional Requirements
        i. Presents the User with a homepage advertising the features of the webapp. The User has the option to log-in, create an account, or submit a ‘contact us’ form. If creating an account, the user will be asked to input their classes taken, major, name, If logged in, the user’s homepage will be their profile page. From there, they can access the various services the application provides: the course scheduler, rate my professor, degree audits, and viewing pages of other users.
    b. Non-Functional Requirements
        i. Cookies will be used to determine if a user is logged in. Submission of a contact form will send an email to us. User data is stored in our database, which will populate their profile page.
2. Course Scheduler:
    a. Functional Requirements
        i. GUI to build schedule
        ii. Allow users to choose classes and put them into a next-semester planned schedule
        iii. Maybe pull degree requirements, allow for users to input previous courses
        iv. Save in-progress schedules so users can return and continue working on multiple sessions.
    b. Non-Functional Requirements
        i. Pull classes from ODA data
        ii. Parsing data from courses to ensure that all course times are listed and do not overlap

iii. Indexing courses in database with relevant requirements, usual courses to be taken after

3. 'Rate my Professor' with FCQs
    a. Functional Requirements
        i. Suggest professors based on FCQ ratings
        ii. Allow students to browse professors and view FCQs
        iii. Search professor feature
        iv. Integrate with course scheduler to provide professor information about various sections
    b. Non-Functional Requirements
        i. Receive and parse data from CU
        ii. Database to hold and allow access to all data
        iii. Allow professor ratings to be viewed in class scheduler
4. Degree Audits
    a. Functional Requirements
        i. A student is able to run a degree audit for any degree. This will be linked to the course scheduler and they will be given recommended schedules based on their audits, preferred times and preferred number of credits
    b. Non-Functional Requirements
        i. Will take input data from CU office of data analytics spreadsheets
5. Database
    a. Functional Requirements
        i. Populates data on loaded page
        ii. User adding info saves into database
    b. Non-Functional Requirements
        i. Setup relational database
        ii. Understand what data we have available from ODA
        iii. Python scripts to save into our database
6. Profile Pages
    a. Functional Requirements
        i. Should display user image and information
        ii. Should be able to connect or "friend" other users
        iii. Should display recommended people with similar courses, interests, etc
    b. Non-Functional Requirements
        i. Private user information should be flagged so only person who placed the input can see it. Only the user should be able to edit their profile
        ii. Data structure to store name, courses, clubs, major, year, etc
        iii. Save this information to database

Project Plan:

- In order to organize our project, we will be using the project management tool, Trello
- https://trello.com/b/hOC82D2X/hackits
- We will all be working together on each feature so our skills can be best implemented

- The order in which we will be completing the features is indicated on the Trello