

Milestone 2

Milestone Two – Project Plan

Project Features list	<ul style="list-style-type: none">• Document (as a bullet point list) the major FEATURES of your software application.• Each features should have a TITLE followed by a brief DESCRIPTION of that feature.
Requirements	<ul style="list-style-type: none">• Document the key FUNCTIONAL and NON-FUNCTIONAL requirements for a minimum of SIX features. Follow one of the formats/templates provided in class. Your application will likely have many more features, and you should create requirements documents for all the features. But for this milestone, you need only turn in SIX requirements documents.
Project Plan	<ul style="list-style-type: none">• Create a Project Plan in your chosen Project Management tool. May be a GANTT chart, Kanban board, or other suitable format.• The Project Plan should include each Feature that is to be developed.• The Project Plan should indicate the SEQUENCE of Sprints. That is, the plan should show in what order the features will be developed (designed, coded, tested, integrated.)• The Project Plan should show for each feature WHEN (by calendar date) the feature will be worked on.• The Project Plan should show for each feature WHO will be working on the development of the feature.• The Project Plan should result in a completed, working application that is ready to present by April 26.

2

Features (Kind of just throw anything down there)

- User registration (Save profiles, login, save favorites, have a history) Underage Option
 - Have users create a profile that will be saved into a database
 - Be able to login using a username and password
 - Have a favorites restaurant list that is saved to the user profile
 - Have a check-in/visited history for the user profile

- Have an underage check mark to denote that alcohol is not a legal option
 - Have the ability to make a reservation
- Filter restaurants by various aspects such as:
 - Price
 - Type of food/alcohol served
 - Hours
 - Location
 - Favorites
 - recommended
- A tool that is used to sort your preferences on what you're looking for. If you can't decide on an option then a happy hour place will be recommended for you based on previous visits to the establishment as well as places you haven't visited before.
- Search - a search bar giving ability to search for restaurants by name or by keywords relating to the restaurant and its food/drinks
- Ratings/Reviews
 - For now, just put the stars you would give the place. Maybe later implement actual reviews from users.
 - Attach restaurant's average score to the restaurant profile. Have users' own score tied to their individual profile.
- Social features/check in?
- Inputted database of restaurants (Stay in Boulder proper for now)

Requirements:

- User Registration:
 - Functional requirement: Allow a user to login with username and password, access account-related information such as favorites
 - Non-functional requirement: Maintain a database of the user data, when user inputs their information that data is retrieved from the database and loaded into the website so they can enjoy full access to their preferences
- Filters:
 - Functional requirement: Allow user to filter based on various criteria
 - Non-functional requirement: Store data on restaurants based on these various categories and then list them based on user specifications.
- Search Bar:
 - Functional requirement: Allow a user to search for a specific restaurant
 - Non-functional requirement: Receive user input, search through list of restaurants for the inputted language, return results.
- Ratings
 - Functional requirement: Allow a user to give different ratings to different restaurants and access them later
 - Non-functional requirement: Receive user input, store this in database associated with their username and password and other account information so that it loads when they login in the future.
- History
 - Functional requirement: Maintain data as to the places the user has been before
 - Non-functional requirement: Receive user input, store this in database associated with their username and password and other account information so that it loads when they login in the future.
- Favorites
 - Functional requirement: Allow users to designate a restaurant as a favorite.
 - Non-functional requirement: Receive user input, store this in database associated with their username and password and other account information so that it loads when they login in the future.

Project Plan:

- Features to be developed:
 - Database: (Week 1 3/11)
 - Set-up

- User Database: Kiana, Devin, Casey
 - Columns:
 - Name
 - Username
 - Password
 - Primary Key
 - Favorites
- Restaurant Database: Markus, Olwyn, Adam
 - Columns:
 - Name
 - Hours
 - Price
 - Ratings
 - Location
 - Alcohol
- User Registration: (Week 2 3/18)
 - Profile Creation Page: Olwyn
 - Log-in page: Markus
 - Favorites Page: Kiana
 - History Page: Adam
 - Location: Casey
 - Saving profile to database: Devin
- Filters: (Week 3 3/25)
 - Price: Devin
 - Alcohol: Markus
 - Hours: Olwyn
 - Location: Casey
 - Recommended: Kiana, Adam
- Ratings/Reviews: (Week 4 4/1)
 - Upload: Casey
 - View others: Devin
 - Edit: Markus
 - Delete: Kiana
 - Bribe: Olwyn
 - User recognition: Adam
- Subsequent Weeks:
 - Add on and tweak!

Trello Link: <https://trello.com/b/tD9jaXv6/thirsty-tracker>