Identify the primary use cases of your app and write a short paragraph describing what the use case covers.

Brainstorming primary use cases

1. Search for a specific drink/recipe
2. Adding ingredients
3. User authentication (user sign in/sign out)
4. Saving recipes
5. Finding help
6. Sharing recipes/drinks/ingredients

**Use Case List**

|  |  |  |
| --- | --- | --- |
| **ID** | **Primary Actor** | **Use Case Title** |
| 0 |  | Searching for a drink/recipe |
| 1 |  | Adding ingredients |
| 2 |  | User authentication |
| 3 |  | Saving recipes |
| 4 |  | Sharing recipes |
| 5 |  | Finding help |

|  |  |
| --- | --- |
| Use Case ID: | 0 |
| Use Case Name: | Searching for a drink/recipe |
| Created By: | Team 9 |
| Date Created: | 10/05/2016 |
| Last Updated By: | Team 9 |
| Date Last Updated: | 10/05/2016 |

|  |  |
| --- | --- |
| Actors: | Users |
| Description: | Search by ingredient |
| Trigger: | The user clicks on the search bar |
| Preconditions: | 1. Internet connection 2. User agreed to a disclaimer and confirmed that they are 21 and older 3. User is at the home page of the web application |
| Postconditions: | 1. A list of drinks that contain the searched criteria |
| Normal Flow: | 0.0 User clicks on the “search by ingredient” button  0.0 User inputs the name of an ingredient  0.0 User is presented with a list of drinks that include the searched ingredient |
| Alternative Flows: | 0.1 User clicks on the “search by ingredient” button  0.1 User inputs the name of an ingredient  0.1 Ingredient is not in the list of acceptable ingredients; error displayed |
| Exceptions: | 0.1.E.0 The web application displays “No matching results” and asks the user to input another ingredient  0.0.E.1 Our server fails. Network error displayed  0.0.E.2 User runs out of memory allocated to browser; honestly there’s nothing we can do here |
| Includes: | Authorization use case (optional as users don’t have to be logged in to search drinks via an ingredient) |
| Priority: | Highest priority |
| Frequency of Use: | Likely multiple times per session, session length probably averages around 5 minutes |
| Business Rules: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: | Use search history to establish possible suggested drinks |

**Sharing recipes:**

“Sharing recipes” use case is a quick way for people to post content from the “Student Drinker” app to Facebook. Users have an opportunity to share either the recipe they like or their own recipe with other Facebook users. It gives users a better experience of using the “Student Drinker” app, by letting other people know they are using the app.

**Adding ingredients:**

This use case allows users to add a number of ingredients that he or she has in the fridge. This function is only available to users who have an account on the website. Once ingredients are added, the user will have an option to generate a search with the added ingredients. The goal is to allow users to use saved ingredients in the future for additional searches.

**Saving recipes:**

What’s the point of having this resource if one can not store their favorites? This would save time for the registered users, perhaps unregistered users could be recognized through cookies. It would be useful to be able to save recipes to one’s favorites as a person is looking at a list of search results as well as when looking at a page of a recipe. Also, it would be great to get suggestions based on saved list of recipes, which in turn could be filtered, searched through, saved into lists (if there are themed parties, etc).

**Finding Help:**

This use case is a button where the user can press to find customer service information and answers to questions for our web application. After the user presses the button, a new page will show up. On this new page, there will be a section called “Contact Information” where the user can find our email and phone number. There will be another section called “About Student Drinker.” This section gives short paragraphs on the background information about the web application and our team. Importantly, there is an “Faq” section too on the page. It is where the users can find frequently asked questions and answers/solutions to those questions.

**User Sign In:**

This use case is, simply, a way for users to sign in. If the user is not signed in, the top of the screen will have a “Sign In” link (if they are logged in, this will become a “Sign Out”), which will direct them to a page (or, perhaps, a pop-up) prompting them for their username and password (and a link if they have forgotten their username or password, which will be hashed in a database [I like BLAKE]), or to be logged in through Facebook (with one of those nice Facebook-provided buttons). If they have forgotten their password/username, they can input their email, and an email will be generated to reset the password. If, after 3 login attempts, they cannot, they will be prompted with a forgot password page. If after 9 they cannot, they will be unable to sign in.

**Search by Ingredient:**

This is one of the primary use cases of the app: to search for a drink by the ingredients in your fridge. At this point, this works solely by search bar and autofill, but in the future this may include a list box and filter (sweeter vs bitter*er* drinks, time/cost, etc.). The search will present a list of alcoholic beverages including the one or more ingredients that you have searched for (ingredients can be added or deleted), preferably listed in order of popularity, weighted for user preference (perhaps stored user-side as cookies or server-side if the user is registered). The user can then select the drink of choice, and is directed to a page with the recipe.