**Milestone 2**

**Team Number: 106-3**

**Date:** March 10, 2019

**Team Name:** The Drinksta Dudes

**Team Members:**

* Andrew Yee
* Trevor Stanley
* Holden Kjerland-Nicoletti
* Teagan Peters
* Joshua Hamel
* Maxwell Sechelski

**Application Name:** Drinksta (like drink instagram)

**Product Features List:**

*-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*

* **Login and or sign-up**:
  + Either login with an existing account or
  + register an account
  + User must confirm they’re 21 or over
* **Website accessibility/compatibility:** 
  + a web browser that displays/formats automatically on either on a PC, Tablet, or Smartphone
* **Drink recipe recommendation:** 
  + Algorithm that finds drink recipes based on the ingredients you enter & theme
  + Hence, a major feature is entering ingredients
* **Rating:** 
  + Every drink recipe can be rated
* **Sharing:** 
  + Every drink recipe will have a sharing feature amongst users
* **Rewards:**
  + Earn reward points from sharing drinks and liking drinks; reward points can be used at your local liquor store
* **Drink Name Search:**
  + Users can search for the name of a specific drink
* **Add a drink:**
  + If the drink doesn’t already exist in our system, a user can specify it and add it.

**Requirements:**

*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.*

1. **Login/Sign-up:**
   1. Name: Registration/User Portal
   2. Basic Description:
      1. Website requires a user to login or create an account if they don’t already have one. User will have to have/create a password that is at least 7 characters long and has 1 special character
   3. Actors: All users who interact with the website
   4. Preconditions: Assume that the user is already registered or needs to register
   5. Basic & Alternative Flow: Display area for user to enter their credentials and also display option for user to register
      1. If they click “register”, a new dropdown menu is present allowing them to specify credentials they want to set.
   6. Exception Flow:
      1. If the user enters an incorrect username or password they can opt to reset it or create a new account.
   7. Post Conditions: The user will be registered
2. **Website accessibility/compatibility:**
   1. Name: Website Design
   2. Brief Description: The website must be able to be easily used on multiple different devices (PC, mobile) and web browsers (Chrome, Firefox, etc.). Therefore, the website should be fully functioning and intuitive to use. The design of the website should be user friendly but also look nice. The user should have a good user experience using the website so that they will continue to use the site.
   3. Actors: User and Front-end website
   4. Preconditions: Assume that the website can be accessed online.
   5. Basic Flow:
      1. When a user enters the website, it will be clear how to navigate the website, register/login, search drinks, and accessing other features.
      2. The user will be able to click on several different links that will go to the correct pages on the website.
      3. The drink pages will also clearly display the ingredients of the drink, a picture of the drink, and its rating.
   6. Alternative Flow:
      1. The website should look more or less the same whether if it is being accessed on a PC or mobile device.
      2. It will also be consistent across multiple browsers such as Chrome, Firefox, Safari, and Internet Explorer.
   7. Exceptional Flow:
      1. If the user does not have internet access the website will not be able to load.
   8. Post Conditions:
      1. The user understands how to use the website and the user experience is consistent on different platforms.
3. **Friends:**
   1. Name: Friends list
   2. Basic description: Add/delete other profiles. See recommended/favorite drinks from friends.
   3. Actors: Users
   4. Preconditions: User has friends
   5. Basic Flow:
      1. Webpage to see current friends. When searching for friends all the names will be converted to lowercase, so that case will not matter. Friends data will be in the database. To delete a friend, there will simply be a button to delete friends. It will delete your id on the database from your friends and yours list.
   6. Alternative Flow:
      1. If friend is not in your friends list then you can easily add one
   7. Exception Flow:
      1. Cannot add a friend if they don’t have an account
      2. If they don’t have an account, there will be an option to send them a link to the home page so they can add you
   8. Post Conditions:
      1. The friend has been added or invited
4. **Ratings**
   1. Name: Ratings
   2. Basic description:
      1. Leave a rating for a particular drink. A simple 1 - 5 star system
   3. Actors:
      1. All users will be able to view the aggregate user rating and leave a rating themselves.
   4. Preconditions:
      1. Assume that the website can be accessed online and communicate with a database to access and store rating data. It is also assumed that the database has been set up and can handle requests.
   5. Basic Flow:
      1. When looking at a certain drink’s page, the rating will be visible to all users and all users will be able to rate the drink themselves from the page. The aggregated user rating will be included with the drinks description tab, with the ability for users to select their own star rating.
   6. Alternate Flow:
      1. If a user has already selected a rating and needs to change what they have entered, the website will allow for a change. The user will simply have to select their desired rating and the website will send an updated value to the database.
   7. Post Conditions:
      1. The user will be able to see their own selected rating, and the value they have selected will be sent to the database, where the aggregate rating will be updated.
5. **Rewards:**
6. Name: Reward Points
7. Basic Description:

* Allows users to gain reward points by sharing, liking and adding new drinks to our dataset.
* Can get more points if drinks you made have a higher rating or more popular.
* Points can be used to get discounts and other rewards from affiliated liquor stores, bars, etc.

1. Actors: Users
2. Preconditions: Registered account
3. Basic Flow:

* Gain certain number of points depending on
* Maximum amount of reward points daily/weekly.
* Use points on rewards

1. Exception Flow:

* Cannot reshare, relike, remake the same drink for unlimited rewards.

g. Post Conditions:

* “Bank” that holds points
* List of rewards where it can be used

1. **Drink Name Search:**
   1. Name: search engine
   2. Brief Description: The user can click the search feature and look up the name for a specific drink.
   3. Actors: The user and the database
   4. Preconditions: The search input function connects/interfaces with the database
   5. Basic Flow:
      1. The name of a drink is entered in either upper or lower case letters. The search function will always convert these letters to lower case and then query the database to find the drink.
   6. Alternative Flow:
      1. If the drink cannot be found, the user will be presented with an option to add the drink. Clicking this option will take them to another page that present fields for them to enter ingredients, a picture, and associate theme/mood.
   7. Exception Flows:
      1. The database will have all of the alternative names for drinks possible.
      2. If a user tries to enter a drink name that is simply an alternative for a different drink, a warning will be presented and they will be redirected to that page.
   8. Post Conditions:
      1. The user either adds a new drink successfully or is redirected to the page where that drink exists
2. **Drink Ingredients Search:**
   1. Name: Search by Ingredients
   2. Brief Description: Users can enter the ingredients they have currently and the website will find all the drinks in the database containing the ingredients and show the user what they can make.
   3. Actors: Users and database
   4. Preconditions: Assumes that the user is able to connect to database in order to filter what drinks can be made
   5. Basic Flow:
      1. Users Enter the ingredients they have into the search section.
      2. After searching, all drinks are shown that contain just their ingredients.
   6. Alternative Flow:
      1. Users want to look up drinks that contain all their ingredients plus other ingredients.
      2. Check an option to show all drinks containing their ingredients and others, leading to more possible drinks.
   7. Exception Flow:
      1. There is no drink in the database with any of the ingredients entered in the search.
      2. A “No drinks found” message is then displayed.
   8. Post Conditions:
      1. A drink is presented to the user based on the ingredients or the drink isn’t found and the option to add a drink is presented.
3. **Add a new drink:**
   1. Name: Enter a drink that doesn’t exist
   2. Brief Description: The user can search for
   3. Actors: user and database
   4. Pre-conditions:
      1. The drink doesn’t exist in our database.
      2. The system searches our database using the listed ingredients to make sure the drink doesn’t exist already under a different name.
      3. This search function will leverage the “Drink Ingredients Search” described above
   5. Basic Flow:
      1. After using the “Drink Name Search” function described above and finding that the drink doesn’t exist, the user will click a presented option to add the drink.
      2. Clicking this link will redirect them to the “add a drink page”
      3. This page will have fields where the user can add ingredients, themes, and descriptions/instructions for adding the drink
   6. Alternative Flow:
      1. The user is at the “add a drink page” and upon entering ingredients, the website presents a warning that the drink already exists in the system under a different name.
      2. If this is the case, the user will be redirected to this page
   7. Exception Flows:
      1. The user enters incorrect information in the fields, such as numeric values instead of a recognized text string.
      2. If this is the case, or the user fails to enter an adequate amount of information for a given drink, then the website will give a warning and highlight in red the fields that need to be changed before the drink can be added.
   8. Post Conditions:
      1. The drink has been added to the database and is visible on its own unique drink page.

**Project Plan:**

**(see Gantt Chart)**