Daniel Lerner, Izzy Gomez, Brian Saavedra, Sara Stiklickas
dibs

Team Work Plan

Stakeholders

- Users
    - Hosts can use the application to get drink suggestions before their events, to prevent their serving areas from getting too crowded, to control the number of drinks available to each guest, and to prevent unwanted guests from taking drinks.
    - Guests can use the application to suggest drinks they want to see at their events, to minimize the amount of time they have to wait near a bar or serving area, and to easily see what drinks are available to them at an event.
- Facebook
    - The Facebook API will be used for identity verification when guests of an event make suggestions or pick up their drinks at the event.

Tasks

Due by 11/13/15

- Initial Design
    - Motivation: Daniel
    - Concepts: Brian
    - Data Model: Sara
    - Security: Izzy
    - User Interface: Sara
    - Design Challenges: Izzy

Due by 11/19/15 (Before MVP is due)

- Revised Design: Everyone
- Create user model: Brian
- Create drink order model: Izzy
- Create suggestion model: Sara
- Create queue model: Daniel

Due by 11/23/15 (MVP is due)

- User controller: Brian
- Queue controller: Sara
- Queue routing: Izzy
- User routing: Daniel
- Integration of the Facebook API for events: Izzy

Due by 12/02/15 (Before Demo)

- Integration of the Facebook API for user verification: Brian
- Notification for user: Sara

- Blacklist Implementation: Daniel
- Drink Limits: Izzy
- Improve UI: Everyone

Due by 12/06/15 (Final code due)

- Improve user suggestion UI: Brian
- Improve user order UI: Sara
- Improve host suggestions UI: Daniel
- Improve queue UI: Izzy
- Security: Izzy
- Clean up code: Everyone

Risks

Implementation Risks

By using Facebook's API, we run the risk of the API not having the functionality we need, which would not allow us to implement some of the features that we wanted our app to have, such as verifying that the user that receives a drink is the one that actually ordered it.

User Risks

For a user of our app, there is the possibility that the drink that the user ordered through our app was taken by someone else, either by mistake or because the host could not identify the person because of a bad Facebook profile picture. This can be mitigated by adding the option of getting previous profile pictures of the user, so that the host can get a good idea of what the person who requested the drink actually looks like.

Another risk that we run is that we run out of drinks during the party, but the stock is replenished at some later time. Our app with its current expected functionality would have no way of taking notice of this. A solution to this problem is to give the host the ability to manually modify the stock in dibs.

Minimum viable product

Our minimum viable product will have all of the core functionality of the app, without the risk mitigations that we have identified so far. The functionality included in the MVP is listed below:

- Event guests can:
    - Suggest drinks before the event
    - Order drinks during the event
    - See the current menu of drinks available
    - Get notified when their drink orders are fulfilled
- Event hosts can:
    - See the list of suggested drinks before the event
    - Access the order queue during the event
        - View drinks in the queue

        - Remove drinks from the queue
        - Notify a guest when his or her drink is ready

What's not included yet in the MVP:

- Blacklisting of guests who order drinks for others
- A limit on the number of drink suggestions each guest can make before an event
- Facebook integration for verification of event guests
- A stylish user interface