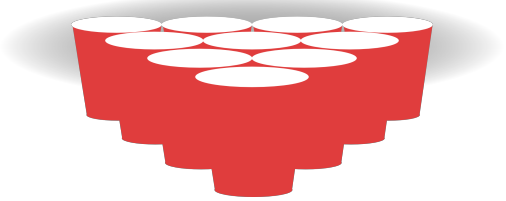
rerack.me



PRELIMINARY DESIGN DOC

Overview

Purpose and goals

Brief description of system to be built
Key goals and purpose (what problem does it solve?)

We would like to build a system to track the results of people's pong games so that they can see how they stack up against their friends. Every time someone plays a game of pong, they can put their results into Rerack, and rise in rank within their group of friends by winning games. The motivation behind this project is to add an additional social and competitive layer to a popular game to make it more fun. There are currently no solutions available to track scores and results in pong specifically, so we will be the first entrants into this specific market.

Motivation for development (eg, deficiencies of existing solutions)

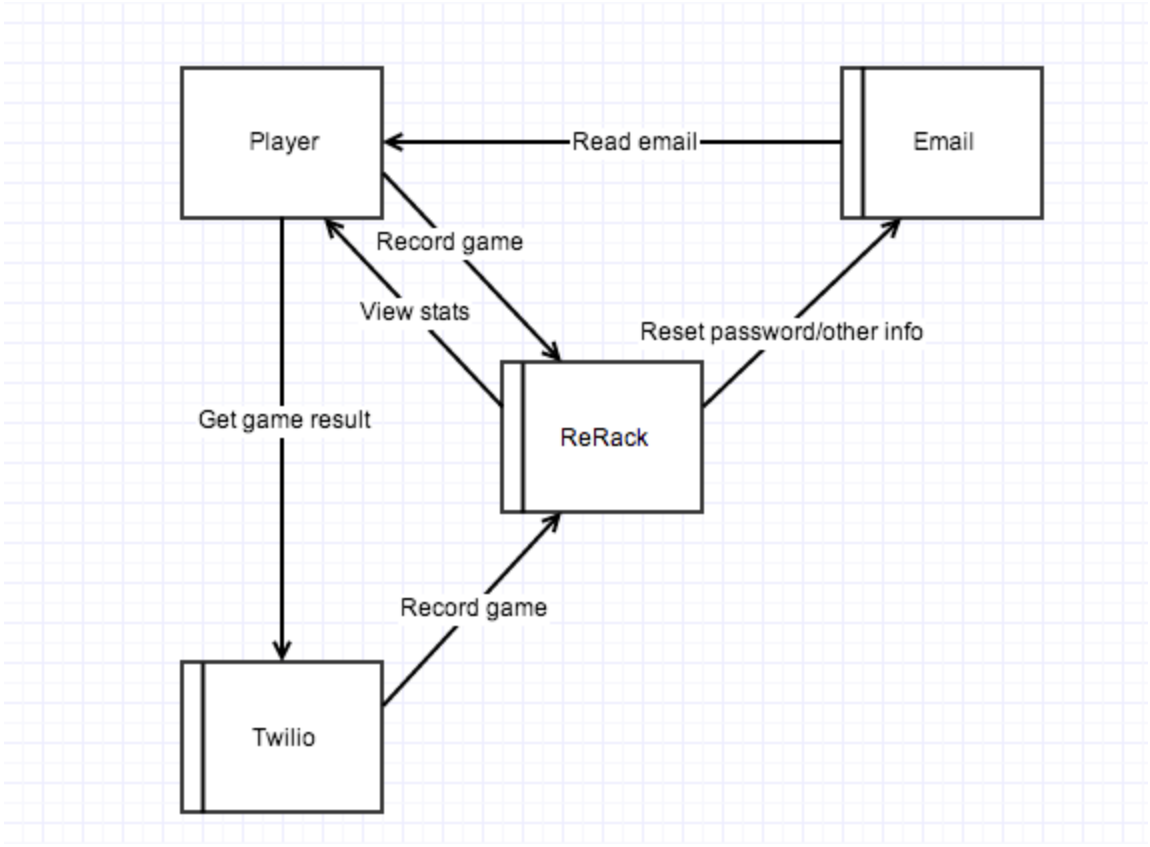
There is not a problem with pong and the way it is now but our team sees potential in adding an element to the game. There is a competitive pong league but it is not very popular and most recreational pong players do not care about pong as a sport. Our application adds a layer of this competitiveness to recreational pong. Recreational players can track their games to make it a little more fun and competitive.

Currently people can track games of any sort using IMLeagues or a variety of other web apps but there is no custom app for pong. We believe that all of these apps have a high activation energy which acts as a barrier and prevents people from recording their pong games. With an application that is tailored to pong players will have a smooth and pleasurable experience.

We are also going to try and add features that make pong a more equitable game. There will be a handicap system where games can be modified to make the games fair. If a very high rated team is playing a team at the bottom of the rankings, the high rated team will be instructed to start with less cups in order to level the playing field.

Context diagram

Establishes boundary of system
Interactions between system and external entities



Concepts

Key concepts

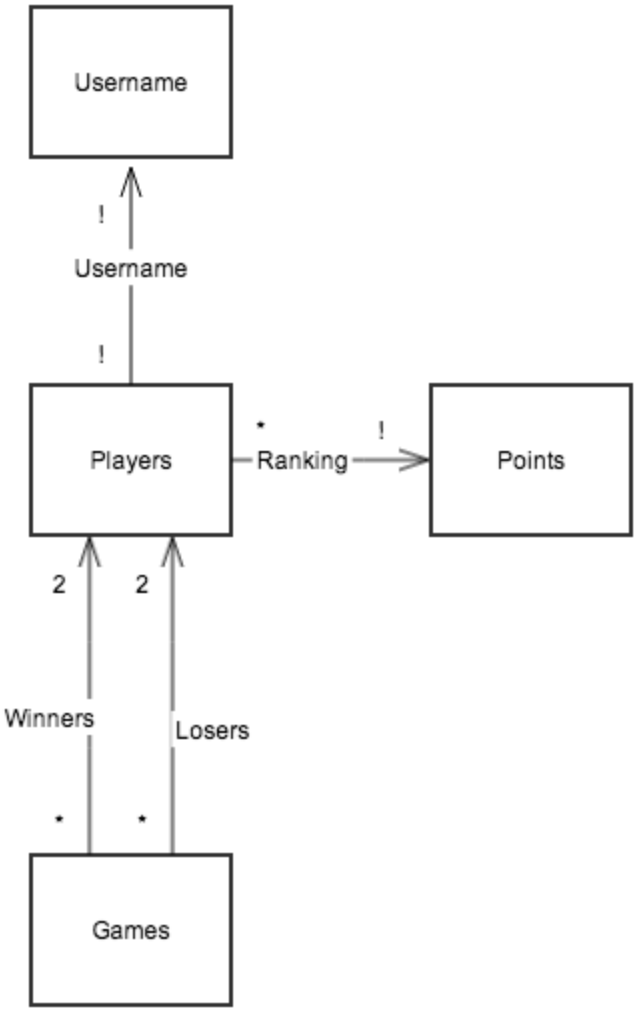
Brief explanation of key enabling concepts

Game: Involves 4 players split into 2 teams; a game of pong is then played with these two teams. At the end of the game one team is designating the winning team and the other the losing team.
Player: Registered or unregistered user who is one of 4 players in a game and becomes a winner or loser.
Ranking: The ranking sorts players according to their history of games. Players are able to move up in the rankings by winning games.
Points: Assigned score to a user that represents how often a player wins at pong.

Group: A collection of people who will have their own rankings. Games played within a group contribute to rankings within the group. A player can belong to multiple groups. An example of a group is a company, fraternity or school.
Rules: A set of rules for games of pong within a group

Data model

Data model of application state
Schema representation details excluded
Syntactically valid diagram with consistent naming & layout
Generalization used appropriately
Names of sets and relations well chosen
Designations in accompanying text of non-obvious elements



In our data model, each player has a username. While this is just a field of the model we decided to include it because the username is important in the app as an identifier of a user. In the final version a user will have an email and a phone number. These two fields and the username will

all be a unique. The player will then be identifiable by any of these three fields. For the MVP, each user will have an email and username but the identifier will be the username. Thus we thought username was important enough to include in the data model.

Behavior

Feature descriptions

Succinct but precise descriptions of each feature

Record games

- Players will be able to record games played. One user will be able to record the game for the four participants.

View rankings

- The rankings list will be displayed on a page so the rankings of all players will be displayed.
- On the home page the user will see rankings for the top three places and a cutout of the full rankings chart highlighting their ranking and their neighbor. This will incentivize them to try and beat their neighbors.

Confirm Games

- The admin of a group can establish that games will only be recorded after confirmation. An email/text will be sent to the other team once a game is reported to show a game is confirmed.

Handicap

- Players will be able to enter in a hypothetical game and the app will give a handicap for the game. The higher ranked team may need to start the game with less cups as to make the game even.

Game feed

- A list of recent games will be displayed to highlight action. It will show what is going on and further encourage competition.

View previous results (Game History)

- On their profile page a user will be able to view their past games and results. They will also be able to track their progress and skill over time.

Groups

- Players can set up groups to have rankings only within a group of people
- Groups will be defined by the players
- Any game that is played where all players are a member of that group will contribute to rankings in that group. Thus if all four players are members of two different groups that game between them will count in both groups.

Privacy

- Players have a log-in password. Later on, we plan to provide texting/email account confirmation.

Security concerns

Summary of key security requirements and how addressed
How standard attacks are mitigated
Threat model: assumptions about attackers
Our app should be secure enough that attackers can't retrieve user data they are not allowed to see, or modify other players' data. It is safe to assume that social game data isn't a high priority target for hackers, so we should only have to guard against common simple attacks, most of which are prevented by following standard web application design guidelines.

Most potential security concerns will be mitigated by using various features of the Ruby on Rails framework. This will include tracking user login, making sure that requests can't be spoofed as if they are coming from a different user, and preventing SQL injection attacks by always sanitizing user input before inserting it into the database.

To prevent players from submitting invalid data that could cause errors down the road, we will use client and server-side validation to check form input and requests to modify information.

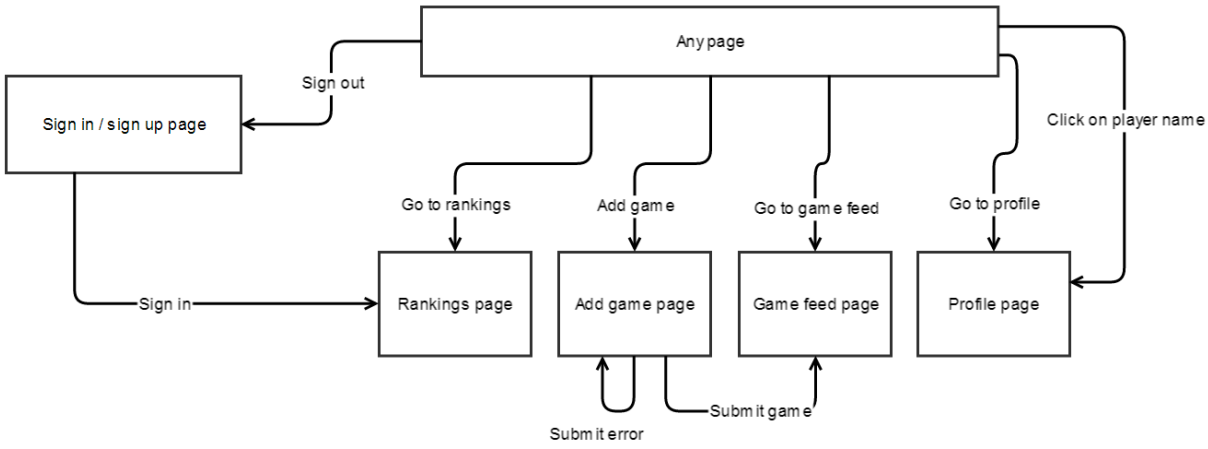
User interface

Wireframes for application
Flow between pages indicated, with named actions
Errors accounted for

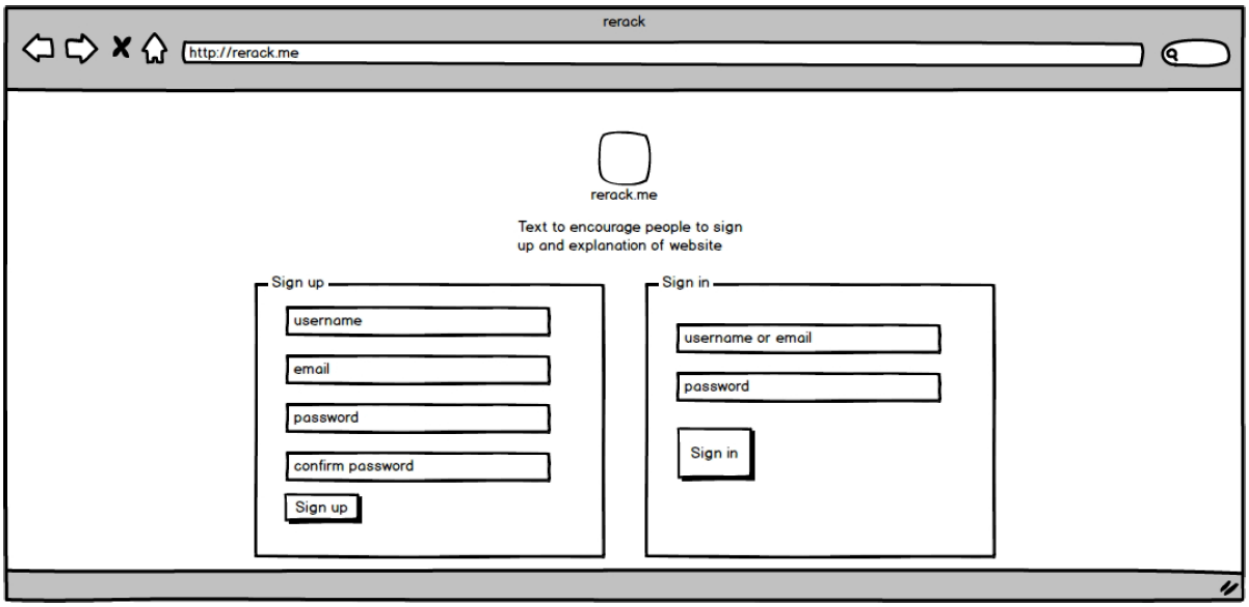
Note about Errors

Form errors will be displayed on the page while the user is typing in input. A user will only be able to submit invalid data to the server if they either have javascript off or are intentionally trying to game the system. Of course, the data will be validated on the server as well.

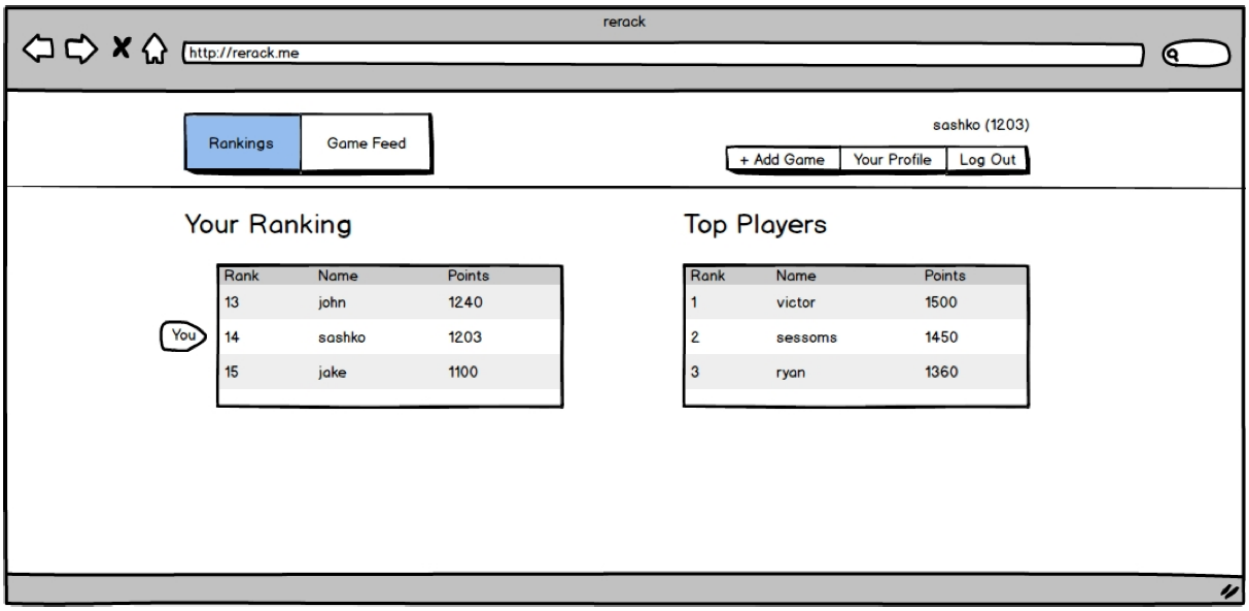
Page Flow



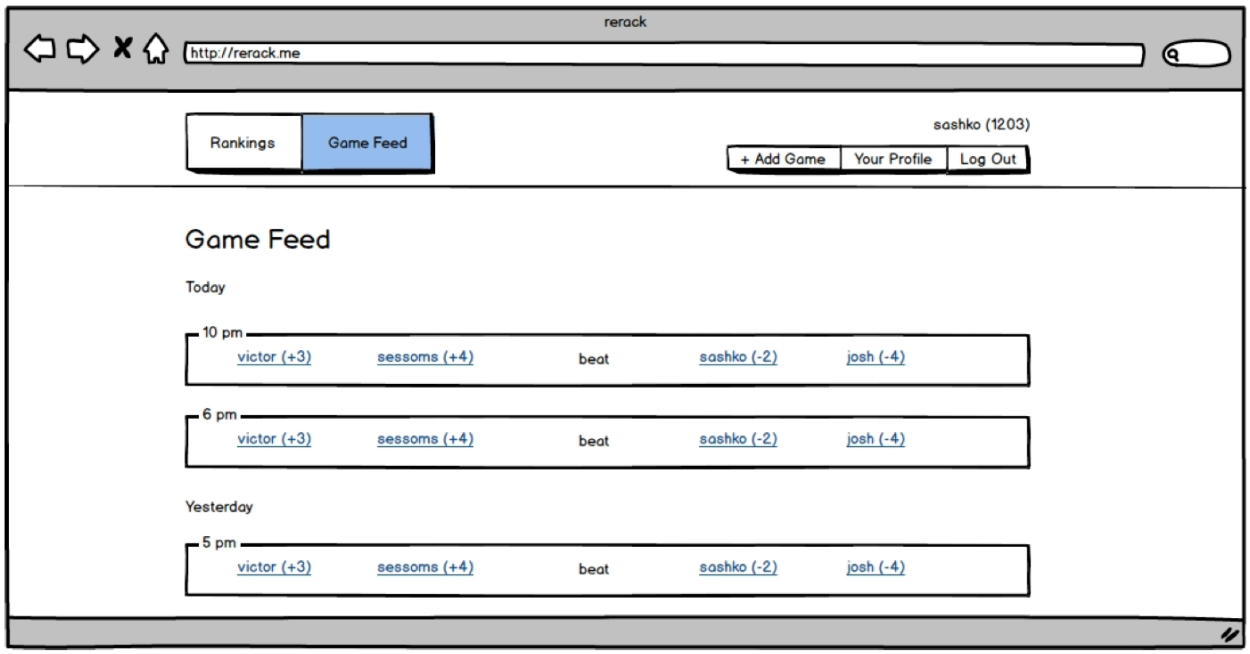
Sign Up Page



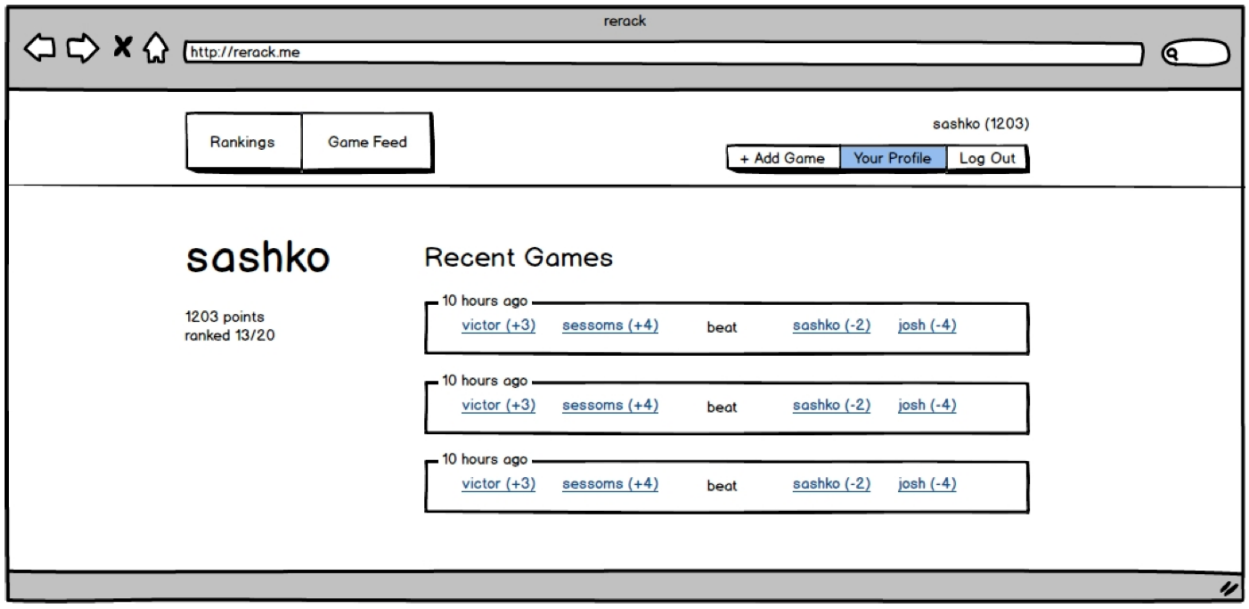
Rankings Page



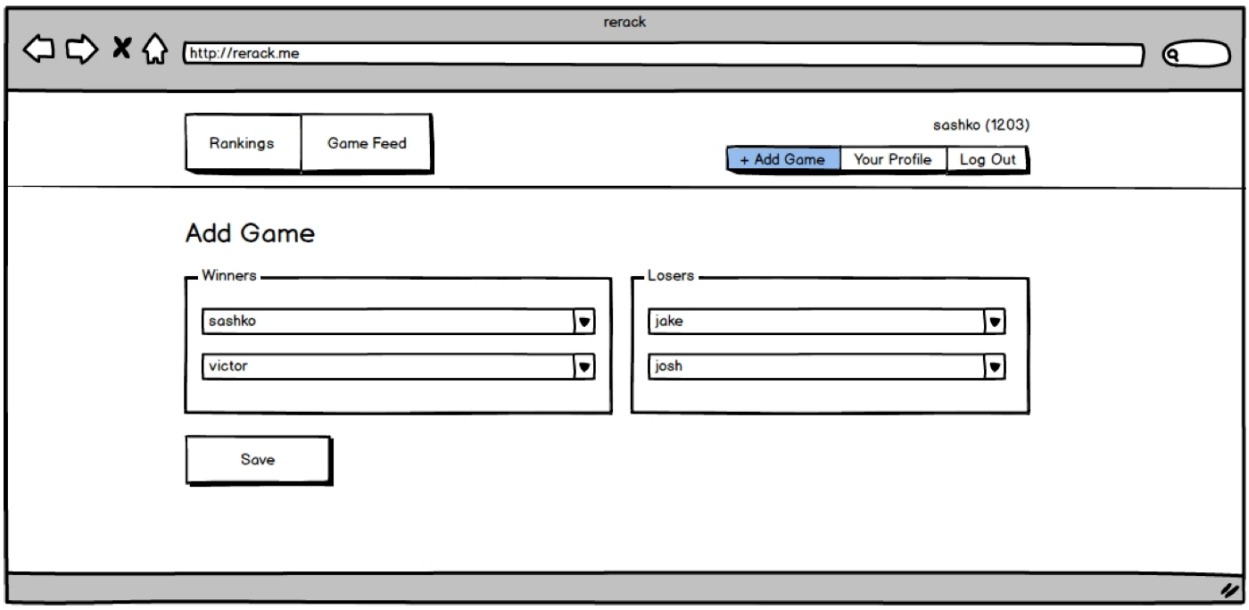
Game Feed Page



Profile Page



Add Game Page



Design challenges

List of problems to resolve in concepts, behaviors or implementation
For each problem: options available, evaluation, which chosen
Notes on code design: schema design choices, abstractions

- Should players be partitioned into groups?
    - Pros:
        - Allows for players to be grouped together easily
        - Prevents everyone that uses the site from seeing your rank
        - Adds its own form of security and validation by making members get accepted into groups
    - Cons
        - Makes it difficult for new members to participate if they don't have an immediate group
        - Who controls which groups are made and prevents duplicates?
        - Have to handle players that belong to multiple different groups and how to rank them.
    - Decision:
        - Chose to postpone Groups
            - Adding groups would add an extra layer of complication to the website that could deter people that simply want to log their score.
            - Without groups, players could more easily play friends that may not belong to the same living group, workplace, etc.
            - We will add groups to the final version
                - Groups in the final version will be implemented in such a

way that allows for players from different groups to play each other easily, but serve as a quick way to view rankings within a subset of players

- Should you be able to email in results?
    - Pros:
        - Could be done from phone
    - Cons:
        - Extra complication that will eventually be replaced by texting or mobile app
        - Not much easier to email than to log onto site and enter in information
    - Decision:
        - Stick with logging in for MVP and switch straight to SMS for Final project
- Should we have features for teams and tournaments?
    - Pros:
        - Would have added another social aspect to the site
    - Cons:
        - Without groups its not really a feature
        - With groups its hard to decide how ranking would be done with teams
        - There are a lot of extra complications for implementation
    - Decision:
        - No teams or tournaments
            - Allows for better ranking system
            - Less confusing for players
            - Without groups it is not a useful feature
- Should we have usernames to display or just use names?
    - Pros:
        - We can regulate the system so that any username is not used more than once
        - Allows privacy to those who don't want their name affiliated
    - Cons:
        - Have to retrieve other players usernames to submit a game, rather than just the names of your friends
    - Decision
        - Games of pong are as often played with/against strangers & acquaintances as with good friends, so you may have to ask for your opponents name anyway. Hence, we will accept the cost of asking for others' username, because unique identifiers will make texting in the game much easier and offers an additional privacy setting.
- Handicap
    - Pros:
        - Would allow for people that didn't feel as good about their pong skills to play others with a high ranking
    - Cons:
        - Add extra coding aspect

- Decision
  - Add handicap feature
    - will only help increase traffic to website
    - help increase the fairness of games and rankings