

Team Name: Game Fame

UNIs: gc2767, sa3522, arz2116, sl3948

Part 0: Python + Linux

Revised Project Proposal

We wish to create an application which allows users to make an account, log in, and invite their friends to 2-player games like tic-tac-toe, hangman, and poker. Although there are platforms which already support some multiplayer games, we wanted to build something simple and easy to use with the most popular games played. Classics like tic-tac-toe and poker are always going to be around and having a quick and responsive UI on the web with no ads or distractions may be a nice way to relax with a slightly competitive edge.

These are the main components we want to incorporate into the app:

1. Implement game logic
 - a. Tic-Tac-Toe
 - b. Hangman
 - c. Poker
2. User authentication via email
 - a. In order to start games, users will have to create an account with their email addresses
 - b. Firebase has built-in support for authentication via email
 - i. Through this service, we can also have users request for their resetting their password, changing their email address, etc.
3. Email invite system
 - a. Users will have the option to start games with friends via username. If the username is not found, the user can send an email invitation to his/her friend to register onto the web app
 - b. We will be using the Gmail API with our game account to send emails for invitations
 - i. For example, if you wish to send an invitation to yourfriend123@gmail.com, we will be able to use our account gamefame@gmail.com to send an email with a generic prompt
4. Active Games
 - a. Each user will have access to all active games being played
 - b. Firebase has support for storing key-value pairs, and we will use something along the lines of:

Key: email; Value: {

Game1: {

Opponent,

Board

}

}

5. Player History

- a. For every user with an account, there will be a lookup system which will show the stats of the user (games won, win percentage, etc.)
- b. Since Firebase allows us to store key-value pairs, we will have something along the lines of:

```
Key: email; Value: {  
  Games won,  
  Games lost  
}
```

Technology Stack

The APIs we will be using are the Firebase API (Pyrebase for Python) for user authentication and backend services and Gmail API for emailing invitations to games. We will be using Electron.js for cross-platform desktop capability.

Python, HTML, CSS, JavaScript, Electron

User Stories

Minimal Viable:

1: As a user, I want to log into the website so that my game history is saved. My conditions of satisfaction are being able to see my wins and losses for the games I have played.

Testing for User Story 1:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password
- b. Output: Upon successful user authentication, there is a tab available to the user which, when clicked, shows the # of wins and # of losses

2: As a user, I want a simple UI so that I can play tic-tac-toe with my mom! My conditions of satisfaction are that there are no ads, we log in via email, and the app is hosted on a website.

Testing for User Story 2:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password, friend's email address
- b. Output: Upon successful user authentication, there is a button for "Start New Game" which, when clicked, will let the user enter an email address to start a game
 - i. In case the email address does not exist in the database, the user will have the option to send an invitation to the gaming platform

3: As a user, I want a menu of available games so that I can see all the games that I can play with my friends. My conditions of satisfaction are a grid or list of playable games.

Testing for User Story 3:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password
- b. Output: Upon successful user authentication, there is a button named "All Games" which, when clicked, will take the user to a page with a list of all playable games

4: As a user, I want descriptions of games so that I can understand games that I am not familiar with. My conditions of satisfaction are that every game must contain a description as well as instructions.

Testing for User Story 4:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password
- b. Output: Upon successful user authentication, there is a button named "All Games" which, when clicked, will take the user to a page with a list of all playable games. Under each game title, there will be hyperlink which says "Instructions." If the user clicks the hyperlink, it will show the user instructions for the game (i.e. General rules, how scoring works, etc.)

5: As a user, I want to be able to switch between games that I am playing so that I am not forced to play only one game at a time with just one friend. My conditions of satisfaction are I must have a list of active games so that I can easily switch between games.

Testing for User Story 5:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password
- b. Output: Upon successful user authentication, there is a button named "Active Games" which, when clicked, will take the user to a list of all active games being played against friends. There will be a "Play" button for each game in the list and at any point, the user can click on that button to resume a game with a friend

6. As a user, I want to have an in-game text chat so that I can talk to my friends during games without having to use an external messaging service. My conditions of satisfaction are that messages display in real-time, have timestamps, and are persistent.

Testing for User Story 6:

- a. User input: Email address (i.e. testaccount@gmail.com) & Password, friend's email address
- b. Output: Upon successful user authentication, there is a button named "Friends." If the user clicks this option, he/she is redirected to a page which lists all the people the user is friends with. For each member of the list, there is a "Message" button which, when clicked, leads the user to a chat with the friend.

If Time Allows:

a: As a user, I want leaderboards so that I can keep track of points I have gained by winning games. My conditions of satisfaction are that users must be ranked in order based on the amount of points they have gained from winning.

b: As a user, I want shareable links so that I can send a link to a friend and play games. My conditions of satisfaction are I can click on a game, create a link, and send it to a friend and play with them.