Contents

1 Unauthenticated User

- The User Loads The Page

  - The User is prompted to enter a username
  - The Username is sent to the server
  - The Server checks to see if there is a person with that name
  - The Server sends affirmations back to the client if the name is not taken
    * Exception: the username is taken
    * The User changes their name
  - The Server enters the user into the queue to wait for a match
  - The matchmaker makes the match between two users

2 Authenticated User — In Game

- The User Enters A Turn

  - The User Receives The Current State of The Game From The Server
  - The client renders the interface using the Canvas API
  - The User Interacts with The Interface to set up the next turn
  - The User sends the data to The Server
  - The server waits for the other player, if he hasn't responded already
  - The server test-renders the data
  - The Server sends out the response
    * If one of the players has won

* The total state changes

- The Client Is Sent A Ping In Game

  - The user does not reply ping
    * Exception: the user does reply
    * Nothing happens
  - Close the user's game in the server system and notify the other player that the other player has won by default

- The User Has Won The Game

  - The client displays a victory message
  - The user pushes a button to go to the next game
  - The user enters the matchmaking queue again

3   Admin

- There is a Bug or Design Flaw in The Server

  - The server is shut down
  - The programmer makes Necessary changes are made to the front-end and the back-end
  - The server is rebooted with the necessary changes