**Final Project Peer Evaluation**

Team: 302

Team member: Ge Bian, Sean Kuang

**Contribution:**

* Ge Bian (50%)
  + User Interface design and implementation (All .html file and main.js)
    - Built the welcome page with hand-drawing user instruction
    - Used Ajax to switch between login page and signup page
    - Homepage and game scene design
    - Drawer menu (Pure Drawer) which displays user information and two ranking boards
    - Help window inside the game scene with hand-drawing instruction
  + User information processing:
    - Sign in with/without Google account (Google OAuth) and Sign up
    - User info model construction and validation
  + Implement the “Global ranking board” and the “Best matches board”
    - Based on the descending sort of the score for each game mode separately
  + Game mode
    - Helped Sean rewrite the initial game implementation with keyboard mode (bird.js, pipe.js, sketch.js)
    - Implement voice mode using p5.js (sketch.js)
    - Game mode selection window using p5.js (sketch.js)
  + Game finish and restart implementation
    - Pop up a customized game over dialog which shows the final score
    - Implement game restart after click the restart button in the game over dialog which reload the page and update both ranking boards
  + Project deployment
    - Deployed the project on AWS EC2 and Apache2 HTTP Server
    - Changed database to MySQL
    - Important information hiding, includes secret keys, google service key and password and database password
  + Error evaluation and handling . If there is an error occur, an customized alert dialog will pop up and give hint message. (SweetAlert)
    - Cannot leave any input area blank in both Sign in or Sign up dialog
    - Sign in error checking
      * Username does not exist
      * Password doesn’t correct
    - Sign up error checking
      * “Password” and “Confirm Password” are not same
* Sean Kuang (50%)
  + Game scene implementation
    - Build auto random generated Map with increasing difficulty(mapGen.py)
    - Build js object model for game objects(bird.js, pipe.js)
    - Implement game using p5 library(sketch.js)
    - Implement global view(globView.js)
    - Implement game operations(move, start, finish)
    - Game UI integration
  + Game model
    - Semi-auto send gaming state and input stream when playing using ajax call every 5 seconds
    - Enable user to resume previous game when logged back in
  + Web socket
    - Enables seeing all the players playing in the same scene in real time
    - Enables a global view where new gamer can drag slider to see global players
  + HTTP -> HTTPS
    - Encrypted server and applied new ip for HTTPS to enable voice mode in chrome

Except the separate contributions listed above, here are the answers to the sample questions:

● Did you (and your partners) set reasonable expectations for what tasks you would complete and by when you would complete them, and did you meet those expectations?

**A:** We finished most of the tasks, except the game replay feature, and we met the timeline expectations.

● Did you accurately communicate your skills, availability, and time constraints to your teammates so that your teammates could set reasonable expectations for your work?

**A:** Yes.

● Did you attend all project meetings and arrive on time, so as not to waste your teammates' time?

**A:** Yes.

● When faced with project problems (whether technical or social), did you communicate those problems to your teammates constructively and unemotionally so that your teammates could adapt their behavior as needed?

**A:** Yes.

● What fraction of the project do you believe that each team member completed?

**A:** We believe that we both completed half of the project tasks.