

## Sprint 2 Planning Document

Group 18

### HvZ Master App

Rob Mantzoros, Kyle Rodd, Paul Steele, Wells Lin, Manasi Goel

#### Objectives

- Fix mistakes from last sprint
- Tagging implemented
- Run a full game
- Have separate interfaces for admins and players

**SCRUM Master** : Paul Steele

**Meeting Schedule:** Tuesday 3:00, Monday 6:00, Friday 11:30

**Risks/Challenges:** Fixing mistakes from last sprint, consistently committing work, testing code.

#### Task List

Task	Time estimate	Assigned
1. As developers, we want to implement more than one game capability		
Create a table for games with info for game code, endtime, and begun flag.	2	Kyle
Add game field to other tables	3	Kyle
Modify db_handler to handle game info	2	Kyle
Create new screen for admins creating game	3	Wells
Server handles game property	3	Paul
Acceptance: Allow admins to use app to create games that are stored in a database		
2. As developers, support proper HTTP methods		
Have server handle JSON objects from client with proper HTTP methods	4	Paul
Have HTTP client send JSON objects to server with proper HTTP methods	5	Rob
Acceptance: Server request should be done by POST and GET HTTP requests		
3. As developers, differentiate admin app and player app		

Be able to show different screen depending on admin/player. Admins should have buttons for creating games. Players should have buttons to join games.	3	Wells
Be able to access different functionalities depending on admin/player. Admins should be able to create games. Players should be able to join a single game.	3	Wells
Acceptance: Admin and player will be able to view and use different functionalities in the app		
4. As an admin, have the app be able to randomly select original zombies when starting the game		
Random humans are chosen and changed to zombies	2	Paul
Acceptance: When a game begins, a definable amount of players are automatically converted to the zombie team.		
5. As a player be able to see mission objectives		
Create a screen to view missions	2	Manasi
Have app be able to retrieve list of missions	3	Manasi
Have server handle request for missions	4	Rob
Acceptance: users can see missions, and their objectives in the app.		
6. As an admin be able to create mission objectives		
Create a screen to add missions	2	Manasi
Have a button to create new missions	1	Manasi
Create a screen to store attributes of mission	3	Manasi
Create a table to store missions	2	Kyle
Have an admin be able to delete missions	2	Rob
Handle request for delete missions	2	Kyle
Acceptance: Admins can add missions, and their objectives in the app.		
7. As an Admin, be able to create revive codes for zombies that converts them to humans.		
Have server randomly generate revive code	2	Rob

Have database store revive codes	1	Kyle
Acceptance: Be able to generate a revive code that can only be used once		
8. As a player, be able to select from a list of games to join		
Create new screen for viewing a list of games	4	Wells
Have server handle join game requests	2	Paul
Have database be able to add users to games	2	Kyle
Acceptance: Be able to see a list of games in the app and click on one to join.		
9. As a player, be able to register a tag through the app.		
Create screen for tagging	3	Manasi
Send server tagging information	4	Rob
Have server handle request and respond	1	Paul
Show cool down timer when tagged	3	Manasi
If human, convert to zombie after timer ends	1	Paul
If zombie, convert to human after timer ends	1	Paul
Acceptance: Able to send tag data to server and receive a response/verification		
10. As a player, be able to see a score screen when the game session is over.		
Create a screen for when the game ends, with info for number of humans, number of zombies, number of tags, number of revives, and number of missions completed.	2	Wells
Have the server send a message when the game ends	2	Paul
Acceptance: Upon the game ending, all users should be able to view the scores of all the players in the app.		
11. As zombie, be able to enter revive codes that convert me back into the human team.		
Create a screen to enter a revive code	3	Wells
Have client send a request to server	2	Rob
Have server send response to client	2	Paul
Have database update accordingly	3	Kyle

Acceptance: Upon entering a code generated by the admin, the player's game state should turn into that of a human again.		
12. As zombie, be able to see a timer until player is kicked out of the game for not tagging constantly enough. (if time permits)		
Add timer for zombies to user main screen	2	Wells
Have server remove player from the game they are in if time runs out	2	Paul
Reset timer when player has tagged	2	Manasi
Acceptance: Have a timer that shows the time elapsed since last tag which can be seen at any time through the app		

### **Hours per person**

Paul: 22

Kyle: 19

Wells: 20

Manasi: 19

Rob: 19

\*Hours estimated are accounting for implementation time and not accounting for learning hours of new technologies.

### **Remaining Items**

- As an admin, I'd like to...
  - directly change database information through the server console.
  - change game parameters, such as cool down timers, through the app.
  - be able to create boundaries for the game session through an in-app map interface.
  - push text notifications to the players.
  - be able to receive complaints from the players.
  - be able to set up polls for player bounties.
  - be able to view poll results and set up bounties for certain players.

- As a player, I'd like to ...
  - be able to see notifications from the admins.
  - be able to send complaints to the admins.
  - view in-game statistics
  - be able to view the set up map boundary.
  - be able to drop out of the game through the app, disabling app features for that user through the duration of the game.
  - be able to vote in a poll for bounties.
  - be able to receive bonus points for tagging a bountied player.
  - look on the map and see gps location of other players. (if time allows)
  - be able to group chat between a limited number of players. (if time allows)

#### **Non-Functional Requirements**

- The application should be able to run on android devices.
- The interface should be smooth and fluid without stutters.
- The phone app should be a UI for the players, sending work to be done on the server.
- Admins should have unlocked features in app.
- The server should handle all requests and do the bulk of the work, while storing information into the database.
- A minimal amount of data should be sent through the app to avoid data overages for the user.
- The server should be able to handle 500 players at once.
- The main screen should be intuitive and have a low number of button presses to get to the desired action.
- The main screen should have a grid layout with sub menus appearing when certain boxes are pressed.
- The app interface should change themes depending on player status (if time allows).
- The application, and server should be able to handle login information securely.