Product Planning

Introduction

The goal of our game is to encourage people to interact with each other in a setting that is occupied by people that generally do not know each other. The nature of the game should encourage all attendees to participate. Furthermore our game will address the general need for entertainment during waiting periods, specifically before an event starts or in between talks. When the attendants of the event are part of an already established group (such as for a workshop) this game can also be used to help team building.

Crucial in our game is that it should encourage participation. Our game does so by making all players feel relevant and encouraging participants to move around using motion controls. Furthermore the game encourages the players to be competitive as the entire auditorium will be split into teams. We believe that this split will give players a sense of belonging to a team, even if players are unfamiliar with each other.Additionally, because we pit the two teams against each other we create a common objective amongst those in the same team that we believe puts them into a same mindset.

High-Level Product Backlog

The MoSCoW model for creating an auditorium based game is given below. Features are ordered arbitrarily; A feature being listed first does not indicate priority. More detailed explanation on each feature is listed in the User Stories section.

Must Have:

Description:

A must have is a feature that our game must have in order to be a working product, ready to be played by actual people.

- Main Screen featuring main game interface.
- Controller Screen featuring individual player interface.
- Interface that enables users to connect to the main game.
- Interface that splits users between a number of teams.
- Boats that represent each team and their respective status.
- Figures that represent each player and their respective status.
- A resizable grid of a size based on the number of players in the team featured on the main team boat.
- A Constant stream of obstacles that present themselves as a danger to each player.
- The ability for player characters to be removed from the team boat.
- The ability for player characters to return to the team boat.
- The ability for players to dodge obstacles.
- Teams must reflect auditorium layout.
- Team boats move towards an end goal at a given speed.
- The speed with which the boats move is determined by the amount of player characters reside on the boat at any given time.
- Graphical Representation for each boat
- Graphical Representation for each player character
- State that represents a given team having won or lost the game

Should Have:

Description:

These are features that are not necessary for the game to be playable, but increase the playability and enjoyability of the game.

1. Difficulty Modes
2. Penalties and Boosts based on poor or strong performance of players
3. Idle Player Detection
4. Multiple graphical themes for player characters
5. Animation when being removed from the boat
6. Animation that indicates the speed of the boat
7. Animation that indicates the player character being hit by an obstacle
8. Animation that indicates a penalty or a boost
9. Dodging obstacles based on movement-based controls
10. A line based map that shows the boats position on the track.

Could Have:

Description:

These are features that can be implemented if we have time left. These are features that are add flavour to the game, but don't increase playability or enjoyability as much as the should haves.

- Distinct characters for every player
- Customizeable levels
- The line based map fading when the boats approach each other.

Won't Have:

Description:

These are features that will not be implemented,either because we do not have enough time or because they are just not feasible

- Animated Lore
- Android App
- Support for players without smartphones

Roadmap

This roadmap represents the global planning of this project in terms of milestones per sprint from week 4 onward to week 8. The goal of this roadmap is to create guidelines for creating working versions at the end of each sprint. Our plan is to release the beta during week 8 so we will not be planning any additional features after this point (week 9 through 11).
When a task depends on other tasks to be completed first this is indicated through a annotation such as “(4.1)”, indicating that a task depends on task 1 in week 4 to be completed first. Dependencies are associative. This means that if task 1 depends on task 2, and task 2 depends on task 3, then task 1 also depends on task 3.

Week 4:

1. Main Screen featuring main game interface.
2. Controller Screen featuring individual player interface.
3. Interface that enables users to connect to the main game. (4.1)
4. Boats that represent each team and their respective status.
5. Figures that represent each player and their respective status.
6. Graphical Representation for each boat (4.4)
7. Graphical Representation for each player character (4.5)
8. Team boats move towards an end goal at a given speed. (4.4)

Week 5:

First Playable Spikes

1. A Constant stream of obstacles that present themselves as a danger to each player. (4.1)
2. The ability for player characters to be removed from the team boat. (4.6)
3. The ability for players to dodge obstacles. (5.1) (5.6)
4. State that represents a given team having won or lost the game (4.8)
5. The speed with which the boats move is determined by the amount of player characters reside on the boat at any given time. (5.2)
6. Dodging obstacles based on movement-based controls
7. A line based map that shows the boats position on the track.

Week 6:

1. A resizable grid of a size based on the number of players in the team featured on the main team boat. (4.6)
2. Interface that splits users between a number of teams. (4.3)
3. Teams must reflect auditorium layout. (6.2)
4. The ability for player characters to return to the team boat. (5.2)

Week 7:

1. Difficulty Modes
2. Animation when being removed from the boat (5.2)
3. Animation that indicates the speed of the boat (5.5)
4. Animation that indicates the player character being hit by an obstacle (5.2)

Week 8:

Beta

1. Penalties and Boosts based on poor or strong performance of players (5.2)
2. Idle Player Detection (4.3)
3. Multiple graphical themes for player characters (4.6)
4. Animation that indicates a penalty or a boost

Definition of Done

The definition of done is composed such that whenever a task is completed every member of the team should have an understanding of what it means for a task to be completed. Additionally this means that this task can then be removed from the product backlog.
A task is done, when:

1. Criteria of the User Story must have been met.
2. Code must have been unit tested and have a coverage of around 80%
3. Features must be play tested to see if the functionality complies with the requirements
4. Code must be commented and have JavaDocs
5. Code must pass static testing tools

User Stories of features:

This explains all requirements defined in the MoSCoW table through user stories. The number represents their position in the roadmap. (*.x) indicates that a feature has not been planned in the roadmap, but may be implemented when we can invest the time. This indication is limited to only Could Have features.

(4.1) - Main Screen featuring main game interface:

As a user,
When I am playing the game,
Then I can view the current status of all the characters and constructs in the game as well as the game progression on a main screen that is tied to the host.

(4.2) - Controller Screen featuring individual player interface:

As a user,
When I am connected to the game through my personal smartphone,
Then I can view information of my own character such as its position on the screen and its current status.

(4.3) - Interface that enables users to connect to the main game:

As a user,
When I have yet to connect to the host,
Then I want to be able to connect to the host through my personal smartphone and get assigned a representative character on one of the boats based on the team I was assigned to.

(4.4) - Boats that represent each team and their respective status.

As a developer,
When I assign users to a team and the users are about the start the game
Then I want to create abstract constructs that represents their team’s boat and its current progression in the game.

(4.5) - Figures that represent each player and their respective status.

As a developer,
When I assign to a team and a boat,
Then I want to create abstract construct on the boat that represents my unique characters and their current status in the game in terms of being on or off the boat.

(4.6) - Graphical Representation for each boat

As a user,
When I am playing a character on a boat,
Then I want to see a nice graphical representation of that boat and all the other boats in the game on the main screen.

(4.7) - Graphical Representation for each player character

As a user,
When I am playing a character on a boat,
Then I want to see a nice graphical representation of that character and all the other characters in the game on the main screen.

(4.8) - Team boats move towards an end goal at a given speed.

As a user,
When I am playing the game,
Then I want to be able to reach my objective of reaching the end of the track in order to reach the goals that are set for me. Therefore I want the boat I am on to move towards that objective at a certain speed.

(5.1) - A Constant stream of obstacles that present themselves as a danger to each player.

As a user,
When I am playing the game,
Then I want to be able to interact with the game through having to react to obstacles that are coming my way in order to present danger in the game. I want these obstacles to either be procedurally generated or generated on a per level basis in order for levels to feel varied and reduce the amount of “same-y-ness”.

(5.2) - The ability for player characters to be removed from the team boat.

As a user,
When I get hit by an obstacle,
Then I should fall off the boat as a penalty for not reacting properly in order to present a reason to play optimally. Falling off the boat means I am unable to participate in the game for a short time.

(5.3) - The ability for players to dodge obstacles.

As a user,
When an obstacle is approaching me,
Then I want to be able to react appropriately to that obstacle by inputting some sort of control that corresponds to me dodging that object. Inputting the correct control at the correct time means I dodge the object and allows me to stay on the boat.

(5.4) - State that represents a given team having won or lost the game

As a user,
When I have reached the end objective,
Then I want to see the game reflect that me and my team have won the game, and that the other team has lost.

(5.5) - The speed with which the boats move is determined by the amount of player characters reside on the boat at any given time.

As a user,
When I find that some of my adversaries have fallen off their boat,
Then I want to see their boat losing speed corresponding to the amount of characters that have fallen off so I can the impact of mine and my other team members performance.

(5.6) - Dodging obstacles based on movement-based controls

As a user,
When I am actively trying to dodge obstacles,
Then I want to be able to perform those dodges through actual physical movement in the real world by using controls based on my own smartphone's gyroscope and accelerometer in order for me to feel more immersed in the game.

(5.7) - A line based map that shows the boats position on the track.

As a user,
When I am performing my best to try and win the game for my team,
Then I want to be aware of how well my team is performing through a line based map that shows my boat's current position on the track.

(6.1) - A resizable grid of a size based on the number of players in the team featured on the main team boat.

As a user,
When I am assigned to a team and a boat,
Then I want to be able to clearly view my own character on the main screen. Therefore I want the boat to feature a grid of a large enough size based on the amount of current players to accommodate for this.

(6.2) - Interface that splits users between a number of teams.

As a user,
When I am connected to the game,
Then I want to be assigned to a team while maintaining some semblance of balance in terms of number of players per team.

(6.3) - Teams must reflect auditorium layout.

As a user,
When I am sitting in an auditorium,
Then I want to be playing a game with those around me to create some feeling of cohesion with those around me. Therefore I want teams to be assigned based on my position and those of the people around me.

(6.4) - The ability for player characters to return to the team boat.

As a user,
When I have fallen off the boat,
Then I want to be able to return to the boat after certain criteria such as reaching a checkpoint are met in order for me to participate once more.

(7.1) - Difficulty Modes

As a user,
When I am about the start playing this game,
Then I want the game to be challenging but manageable. Therefore I want to be able to select a difficulty mode that is best suited to my own skill level.

(7.2) - Animation when being removed from the boat

As a user,
When I have been hit by an obstacle,
Then I want to see a graphical effect that indicates this clearly through animation.

(7.3) - Animation that indicates the speed of the boat

As a user,
When I am playing the game,
Then I want to see a graphical effect that indicates the current speed of the boat through animation so I can see our prowess in relation to the other teams.

(7.4) - Animation that indicates the player character being hit by an obstacle

As a user,
When I fail to dodge an obstacle,
Then I want to see a graphical effect that I've been hit through animation as a clear indication of my failure.

(8.1) - Penalties and Boosts based on poor or strong performance of players

As a user,
When my team and I are performing well,
Then I should be rewarded through a noticeable boost in speed.

As a user,
When the enemy team is performing poorly,
Then they should be penalized through a noticeable loss in speed.

(8.2) - Idle Player Detection

As a user,
When I see that one of my teammates has stopped participating in the game but has yet to disconnect from the game,
Then I want to see that the game detects that that person has become idle and automatically remove that player from the game so I can feel that my team is working together as they should.

(8.3) - Multiple graphical themes for player characters

As a user,
When I am playing the game,
Then I want to be able to identify my own character easily. Therefore I want to see multiple graphical representations of playable characters to make characters more distinct.

(8.4) - Animation that indicates a penalty or a boost

As a user,
When my team has been rewarded by a speed boost,
Then I want to see a graphical effect that indicates this speed boost through animation.

As a user,
When the enemy team has been penalized by a decrease in speed,
Then I want to see a graphical effect that indicates this decrease in speed through animation.

(*.1) - Distinct characters for every player

As a user,
When I am playing the game,
Then I want to be able to identify my own character easily. Therefore I want the graphical representation of my character to be unique.

(*.2) - Customizable levels

As a user,
When I have played this game many times,
Then I want to be able to customize my own levels in order to add more levels and possibly more challenge to my gaming experience.

(*.3) - The line based map fading when the boats approach each other.

As a user,
When my boat and the enemy boat are near to each other,
Then I want to see them approaching each other on the screen; not just through the line based map. Therefore I want the line based map to fade away to the side of the screen to allow the screen to show the near collision between the two through animation.