Elbowroom progress report

Of work done by Jacob Goldberg, Joel Berman and Kyle Jacobs
Intended for Jacob Clarkson

Reporting on 10 – 8 – 2015 to 21-08-2015

Finalized on 21-08-2015

Contents:

1. Task Activities

1.1. Level Design*: this activity involved creating a room design blueprint, and joining multiple instances of this blueprint with connecting platforms to create a complete racetrack.*

1.1.1 Referenced Use Case: on completion of this activity the use case "Room Navigation" will be fully functional.

1.1.2 Personal Contributions:

Jacob Goldberg:

- Created room model.
- Setup initial lighting.
- Created 3rd person camera
- Implemented a main "Home screen" room.
- Created the platforms connecting the rooms and made them move.

Joel Berman:

- Added color to the room model
- Created colored props
- Created reflective lights
- Added in-game text
- Improved scene lighting

1.1.3 Test Preparation and Outcomes:

Tutorial Difficulty

- Feedback form: http://goo.gl/forms/3mpkyl5nST

1.2 Character Movement: *Avatar movement states were mapped to input keys. The player then influences these states and the appropriate action is called. E.g. pressing "W" will cause the avatar to enter a running state that will increase their forward velocity and play the appropriate animation.*

1.2.1 Referenced Use Case: since the animations are not properly implemented, the "fluid character movement" use case is has not been implemented fully.

1.2.2 Personal Contributions

Joel Berman:

- implemented customizable character controller
- setup animation controller
- fine-tuned character movement speeds to suit game feel
- edited animations to fit character better
- implemented wall jump

Jacob Goldberg:

- implemented the forward roll movement
- implemented player logic, so that the player can only move when alive

1.2.3 Test Preparation and Outcomes:

- Forward rolling close to a trap sometimes results in the player moving through the trap. Testing roll on various test game objects with varying shades, widths, heights and triggered animations will help us narrow down the problem.

1.3 Trap Design*: created a base trap with attributes that should be contained in all traps, and made more specialized traps derive from the base trap. Successfully implemented several traps including: a spinning platform, a windmill, turrets and moving walls.*

1.3.2 Personal Contributions

Jacob Goldberg:

- Created the spinning platform trap.
- Created the growing wall trap.

Joel Berman:

- Created turret trap
- Create moving wall trap
- Created windmill trap
- Created popup trap

Kyle Jacobs:

- Cube field
- Created waypoint trap
- Helped create moving platform script

1.3.3 Test Preparation and Outcomes:

- All traps implemented have been tested in isolation from the main game. We played through multiple interactions with the traps to ensure that their behavior is consistent.

2. Planned Activities

2.1. Multiple Levels*: having implemented the level blueprint, we will begin designing rooms three, four and five and implement them in the next level.*

2.2. Audio and Scoring: *the player's quickest and previous times need to be displayed per level on exit. Background music and sound effects which suit the game feel need to be implemented.*

2.3. Networked Multiplayer: *This activity involves using Photon, a unity networking plugin, to set up a server and allow players on different machines to play against each other.*

2.4. Intelligent Rooms: *This activity forms the AI component of our game. We will design a room which predicts the player's direction and creates traps in real time to impede the player's progress.*

3. Problems

3.1. Networked Multiplayer*: it is taking longer than expected to understand how to use the networked multiplayer plugin "Photon". The more features we add to the game at this stage, the more difficult it is going to be to make the game multiplayer through a network in the future.*

3.2. Scoring: *the scoring system has presented a problem mainly because we are still unsure as to how players will be represented in the game, especially after the networking code is implemented.*
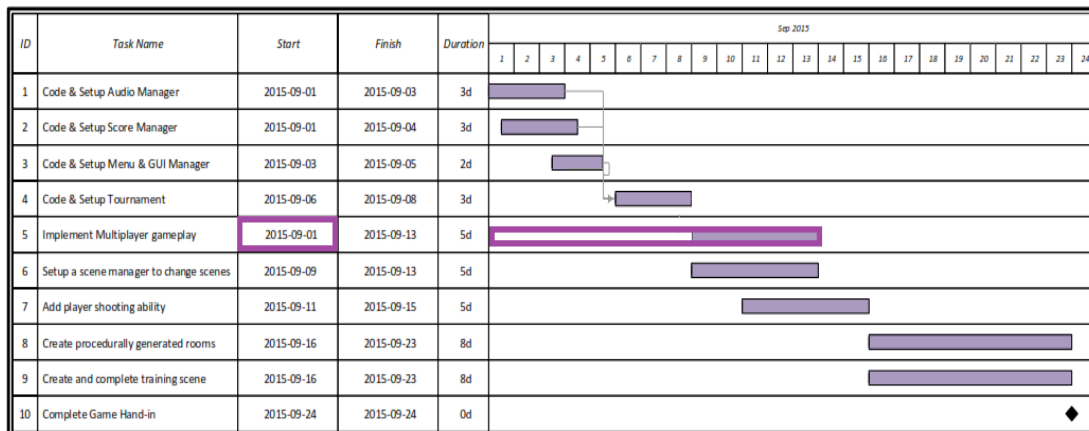
3.3 Animations: *the animations we received were not set up properly, so we have not had the opportunity to judge if the final animations suit the game's look and feel. This problem occurred on the 5th and is currently being resolved.*

3.4 Incomplete Tasks: *the team has met all deliverable due dates specified in the project plan, except the 6th item: create a crumbling wall trap. We concluded that it will not be a feasible mechanic in a multiplayer game as it means the second player has to wait for the wall to regenerate, which will be frustrating for them and is not conducive to the fast-paced style we are aiming for. To resolve this we created an alternate windmill trap which better fits the game's style.*

4. Revised Schedule and Goals

3.1. Networked Multiplayer: *given the difficulty of implementing a networked multiplayer game we have allocated eight more days of work on its implementation. Two team members will work on networking whilst the other continues fleshing out the GUI and Audio components of the game.*

Fig 1. *Multiplayer implementation has been given 8 extra days*

ID	Task Name	Start	Finish	Duration
1	Code & Setup Audio Manager	2015-09-01	2015-09-03	3d
2	Code & Setup Score Manager	2015-09-01	2015-09-04	3d
3	Code & Setup Menu & GUI Manager	2015-09-03	2015-09-05	2d
4	Code & Setup Tournament	2015-09-06	2015-09-08	3d
5	Implement Multiplayer gameplay	2015-09-01	2015-09-13	5d
6	Setup a scene manager to change scenes	2015-09-09	2015-09-13	5d
7	Add player shooting ability	2015-09-11	2015-09-15	5d
8	Create procedurally generated rooms	2015-09-16	2015-09-23	8d
9	Create and complete training scene	2015-09-16	2015-09-23	8d
10	Complete Game Hand-in	2015-09-24	2015-09-24	0d

4. Notes on Terminology: *This report contains some words which have a specific meaning in the context of the game.*

Trap: a trap is a game object that the player should avoid. It is used very broadly in this context. A trap could be a simple spinning cube that prevents the player from moving in a certain direction, or a platform which shrinks when the player touches it.

Platform: a simple cube which the player can stand on.

Room: a platform with walls on either side, which contain a number of traps that the player has to navigate through.

Race: a race in Elbowroom is organized as a set of rooms, each of which are connected by a number of platforms. The player navigates through the rooms and jumps onto the platforms to the next room until they reach the final door, completing the race.