**Project 2**

**Design Document: Super Mario Bros**

Jenis A. Modi

Justin K. Shelton

Xin Tang

[jenis.modi@wsu.edu](mailto:jenis.modi@wsu.edu)

[jkshelton@gmail.com](mailto:jkshelton@gmail.com)

wxt38@hotmail.com

November 7, 2012

**Table of Contents**

1. Executive Summary 3
   1. *Game Title 3*
   2. *Game Genre 3*
   3. *Abstract of game story 3*
   4. *Game play 3*
2. User Interface 4
   1. *User interaction 4*
   2. *Game first look 4*
3. Game Entities 5

*3.1 Background 5*

* 1. *Objects and their interaction 5*

4. Goals 6

*4.1 Player goals 6*

5. Difficulties 7

6. Development Strategy 7

7. Low bar/High bar 7

*7.1 Low bar goals 7*

*7.2 High bar goals 7*

8. References 7

1. **Executive Summary**
   1. *Game Title*

Super Mario Bros

* 1. *Game Genre*

Platforming game (Multiplayer)

* 1. *Abstract of game story*

Super Mario Bros follows the story of Mario trying to find Princess Peach, who has been kidnapped by the evil Bowser. The story is of very loose association with the game and is mostly determined by the player.

* 1. *Game play*

The original Super Mario Bros game contained the below features:

* Different kinds of enemies to make game more interesting.
* Certain enemies fire bullets.
* A count down timer granting additional points at the end of each level.
* Coin collection in order to increase number of lives/points.
* Breakable walls and walls contain bonus power-ups but are visually indistinguishable from other walls.
* Existing power-ups in the original Super Mario Brothers game:
  + Flower gives the ability to shoot fireballs.
  + Mushroom to increase the size of player.
  + Stars grant a temporary invincibility.

The new features plan to include in the game:

* Areas of the map that can only be accessed by cooperation of players such as a higher platform that can only be accessed by one player making a jump while directly under that player.
* Feather power-up – will decrease the falling speed temporarily when the player presses the jump button.
* Battle-mode – players will pitch against each other in an area where they have make the most points in a given time period or outlive the other player.

1. **User Interface**
   1. *User Interaction*

* The primary movement controls will the directional keys to move the character left or right and the space bar for a single jump.

Other in-game keys include:

* M – Mute the sound
* P – Pause/Un-Pause the game
  1. *Game First Look*
* The game will be scored according to the following:
* Killing enemies
* Breaking rocks
* Obtain power-ups
* Touching the check-point flag pole
* Pass the level before the timer ends
* The game ends if:
* Player lost all available lives.
* Player beats the other player in battle-mode
* Different power-ups:
  + Add something here

1. **Game Entities**
   1. *Background*

The game will have a side scrolling image.

* 1. *Objects and their interaction*
  + Walls – player can stand on top of walls and they can be destroyed. Some walls are actually hidden treasures that can give coins or a power-up.
  + Coins – coins give the player points
  + Power-ups – certain power-up can on appear when the player is not in base level. At base level, player can only get mushrooms and there will only be wings and flowers power-ups after the player is ‘leveled up’.
  + Enemies:
    - Koopa Troopa “green turtle” – If the player stomp on top of the turtle or a hit from below then it will retract into its shell. The turtle dies if the player touches the turtle again while it’s retracted or fireballs.
    - Bowser – Bowser is the final boss of the game, he requires multiple stomps in order to die.
    - Goomba – They walk in a straight line. Fireball, stomp, or a hit from below will kill the little guy.

1. **Goals**
   1. *Player goals*

* Short term goal:

- Beat the game before the time is up

* Long term goal:

- Beat the game with the highest score possible.

1. **Difficulties**

Making the game functionally the same as the original Super Mario Bros game, from look and feel to the way physics are handled.

1. **Development Strategy**

We are going to develop this game from scratch. The hardest/most difficult part of the project is “platforming” and “networking”. Platforming in games is not an easy thing. We have to take care of certain things in platforming. For example, player has to stop the walking or jumping if he hits a wall, he has to fall down of platform if end is reached. He also has to stop falling if he lands on another platform. To approach “platforming” in the game, we are going to start looking in some of the existing “platforming” games developed in java. Then divide the part in pieces. For example, we will control the movement of player by having Boolean flag for falling or jumping. We will also make sure that, the game is tile based to check for each and every smooth collision in the game. So, at this point of time, we are assuming that the movement of player will be the same as what we had in our project 1 games1.

For “networking” aspect, we are first going to see what kind of networking we need to have in the game. We might end up with “semi-dumb” approach for this part of the project. We will take input from professor on this part of the game and also look into some of the tutorial on world of internet.

**Development Process:**

**Milestone 1:** **Get basic physics/spritesheet working in the game**

**Roles:**

Justin/Xin: will play important role in get things going in terms of simple physics and investigation on it.

Jenis: will look in to some of existing games which have platforming and then give the input to the team.

**Target:**

Alpha day

**Milestone 2: Get networking in the game**

**Roles:**

Jenis: will work extensively on this part of the project and get it working.

Justin/Xin: will help in integrating the code with actual physics.

**Target:**

Last presentation of the game

**Milestone 3: Collision detection and AI demo**

**Roles:**

Jenis/Xin: will divide collision detection between different objects and then complete it as soon as possible.

Justin/Jenis: will work on AI demo.

**Target:**

Last presentation of the game

**Milestone 4: Final Game**

**Roles:**

Jenis/Justin/Xin: will complete the integration of the game, finish the low bar goals and try to work on high bar (if get more time).

**Target:**

Last presentation of the game.

1. **Low/High bar**
   1. *Low bar goals:*

* A working clone of the original Super Mario Bros, featuring roughly three levels from the game in the fashion of the first level, some midgame level, and Bowser’s Castle.
  1. *High bar goals:*
* Full network support for two player sidescrolling.
* Additional powerups.
* Realistic Mario physics.
* Innovative gameplay mechanics.

1. **Technical Showpieces**

For Technical showpieces, we will have AI demo and networking in the game.

* 1. AI

For this part of technical showpiece, we will have an Artificial intelligence demonstration of the game. At the start of the game, if player does not press any key for 5 seconds, the player will start playing game by itself. It will have a random play mode every time he plays the game. This part will be more interesting for game as it will also give basic idea about the game to the user. We might also try our hands using some of existing data structures to implement this feature of the game.

So, for this part to be working, we will have timer in the game. So, if the time threshold to game is met, the player will go ahead and start the game automatically. It will have random event every time it starts the game. I have looked in to the Breakout demo piece developed by Fredton Doan and will try to use similar concept in this game. The approach for this game might be different, but idea is going to be the same. Also, I can try to use some of the path-finding algorithms for making this demo.

* 1. Networking

We had talked about this feature in the class, but after seeing your comments – we might end up with single player in the game. But this was the idea which we had if we have to implement it. Probably, we might keep this as a High bar goal:

For this part of technical showpiece, we will have networking in the game. We will have 2 players in the game and both players will be controlled by different machines. This will be cool stuff in the game as two players can kill the enemies and it will make the game more interesting.

We will have 2 players in the game, which are going to be controlled by movement of each other. For example, if player 1 is at west-end of screen and player 2 is at east-end of screen, player 2 will not be able to move ahead. In order for player 2 to move ahead, player 1 has to move from west-end of screen to ahead. If this feature is not applied, then we might lose player 1 on other part of the screen, and might not able to track him.

For development purpose of this feature, we will analyze the various networking approach, and apply the appropriate approach to this game.

1. **References**

[1] http://higherorderfun.com/blog/2012/05/20/the-guide-to-implementing-2d-platformers/