**Hop Hop LEGO**

**[Author: Wael Alhamwi]**

**Summer 2016**

*\*Begin to work on your game design doc. It will be necessary to document details of your game no matter how simple your game will be.*

*\* At the least, you must complete yellow highlighted parts of this template. Feel free to add more information that you think is necessary for your game description.*

*\* There is no explicit submission deadline for your design doc alone. Submit your first version along with your first project code. Your design doc might be revised. Submit your final version as a part of your final game package.*

*\* When you design your game, keep in mind that actors that you will be provided with are characters like Ralph and Panda with simple animations. Feel free to find and use other models if you want. In this case, you need to disclose the source for the model. As for the environment, you will be required to construct obstacle courses using simple building blocks like a cube, plane, disk, or sphere. See sample models.*

***Design Inspiration & Guide: See*** [***https://www.youtube.com/watch?v=bc6ov1-KUmQ&feature=related***](https://www.youtube.com/watch?v=bc6ov1-KUmQ&feature=related)

I. GENERAL INFORMATION

This is a platform game about [LEGO] that has to participate in an obstacle course to [Cross a moving obstacles].

II. DETAILED GAME DESCRIPTION

* Basic Concept –

[LEGO] has to run obstacle courses and collect valuable items to clear the game.

* Background Story –

[LEGO] has to participate in [your game story].

\* What is the tone? What is the basic narrative? What is the "heart" of the story? Is it a linear story?

The game has a [Adventure] tone.

The plot is [LEGO need to reach other side of the area before time] The “heart” of the story is [to rescue its stranded panda pet].

The story will be linear.

* Objective –

Get to the end of the level(s).

* Gameplay –

The player will be allowed to run and jump across parts of the map. Also, the player will be able to attach enemies and collect valuable items by [Getting close to it].

* AI component –

Enemy AI will be rudimentary. They 1) walk back and forth, 2) notice the player and change their directions, 3) make a [Throwing rocks] action to block the player from moving forward, 4) […]

* What is the planned interface?

Keyboard for [Movement] and mouse for [None].

* What is the planned perspective (1st person vs. 3rd person)?

3rd person, [top/behind/] view

* What are SW and HW platforms?

SW - Panda3D game engine with Python programming

HW – PC

* What is the basic interactive structure? (e.g. Chapters vs. Great Middle Section, Levels, etc.).

Levels with level transition effects

* What is the "heart" of the gameplay? (e.g. speed, actions, style, continuous, turnbased, etc.?

Platform game with the various obstacles, collectable items, and enemies.

* Does multi-player work?

Only single-player.

* How difficult is the game? How long will it take the average player to complete?

Easy for novice game player. The game doesn’t have the functionality of letting users select the difficulty levels. The game consists of 2 levels. All levels are easy. The second level is slightly more challenging than the first level.

III. OTHER ASPECTS OF THE PRODUCT DESIGN

* Characters --

[Player LEGO] - This is the player character

* Default Properties: How does the player begin the game? As soon as he/she run the game
* Winning: How can the player win? By successfully crossing over all obstacles and meeting with his pet
* Loosing: How does the player lose? By consuming all life’s that the player has or run out of time.
* Health: Instant death in case the player fell off or got hit by the enemy rocks
* Weapons: No weapons
* Actions: Jump, Collect “automatically when getting close to tokens”
* Player Rewards (Power-ups & Pick-ups) Collecting tokens to increase life

[Enemy Blue Gang] – This is the enemy type 1.

* Normal State: What is the object doing if it has not come in contact with the player? Moving side to side and throwing rocks in random directions
* Detection State: What does it take for this object to detect the player?
* Throwing rocks close to player trying to hit it “LEVEL 2”
* Reaction State: What does the object do as an action after passing the reaction state?
* Get back to Normal state
* End State: What happens to the object after player has reacted correctly or incorrectly to object? Get back to Normal state
* Controls --

Control table

|  |  |  |
| --- | --- | --- |
| **Action** | **Control** | **Context** |
| Jump | Space | All levels |
| Forward | w | All levels |
| Backward | s | All levels |
| Rotate-Left | a | All levels |
| Rotate-Right | d | All levels |
| Rotate-cam-Left | q | All levels |
| Rotate-cam-right | e | All levels |
| Exit | escape | All levels |

* World --

The only setting is supposed to be in the obstacle courses [moving objects]

## Level Design –

Laying out the large-scale features of the obstacle course map, such as steps, holes, walls, etc., for players and enemies to move around in. Also,

* Specify scoring systems, allowable moves, time limits, starting resources, etc.;
* Specify the start (entry) and exit locations, teleporters, hidden passageways, etc.;
* Specify locations of various entities, such as enemies, collectable items, etc.
* Specify aesthetic details such as level-specific graphic textures, sounds, animation, lighting and music;
* Obstacles are:   
  moving objects “need to jump on top of it or player will fall” ‘LEVEL 1’.
* In LEVEL 2 the objects can fall down after a random period of time.   
  Rocks coming from enemy toward players.
* Enemy movement: In normal state: enemy moves back and forth but stops every fixed time to throw a rock at a fixed direction.  
  In detection state: enemy will stop and start throwing rocks near player trying to hit it.
* Player will collect tokens to increase its life count every 3 tokens will add 1 life to the original life’s count
* Pet position: End side of the game, behind enemies and opposite side of player.
* Player start position: player will start on the side opposite side of the pet.
* Enemy position is on the same side of the pet and in front of it.
* Graphic textures: Environment is an environment model provided by Panda3d, topped with sky texture and bottom of grass and surrounded by trees.
* Lighting: Global lighting “Day light”
* Describe the on-screen interface (if there is a score and a life gauge... if there is an inventory icon and dialogue choices...), and how it works.

[eg. Life count bar is on the upper left corner of the screen. Number of collected tokens is on the upper left corner below life count bar. Time count down is on the lower right corner]

* Describe all menus in detail, and chart out the "shell" structure.

Main menu consists of Start, Level Select, and Quit.

* Onscreen text messages are also part of the interface -- if not detailing all onscreen messages in this document, describe in general terms what they will be like.

[Lets Go!!!, "GAMEOVER!", "YOU LOSE!!!"]

* Graphics -- Describe the general style of the graphics.

Low poly models.

All models provided by the instructor or from the Internet.

Environment models (obstacle courses) will be procedurally constructed in Panda3D by using simple building blocks such as cubes and spheres/disks.

* Sounds and Music -- Describe at least the general manner in which sound effects will be used in the game.

The sounds are [jumping, footsteps, collecting-effect-sound, enemy-throwing-rocks, player getting hit, losing, winning].

The music are the [] for level 1, [] for level 2, …

* License Exploitation --

None, original universe. Music will be credit to their respect artists.

LEGO Model used in this game is accredited to LewisChen