

Hyper Sniper

Introduction

A study of the Mario Party 6 minigame “Hyper Sniper”, repurposed for PT use. The current build focuses on the development of an environment to emulate a minigame where players shoot targets and compete for score. New studies have begun to explore utilizing more serious games for rehabilitation through principles such as meaningful play and challenge [1]. We wish to test whether the utilization of mechanics and concepts that are already found to be engaging to people in normal circumstances can be refactored for rehabilitative use.

Program Overview

When you start the app, you begin at a menu where you can add players and start the game.

Instructions are built into the menu, but to repeat them:

- The goal of the game is to compete on a board to shoot point targets and collect points faster and more accurately than your opponents.
- A randomly generated panel of targets moves from the bottom to the top of the playfield over 30 seconds and you must move your reticle and shoot a target.
- Smaller targets are worth more points.
- This game supports usb controllers (no keyboard controls.) if you have any controllers plugged in to your computer, press any button and the game will register a player to that controller.
- This game supports up to 4 players, they can move their reticle with the left stick and shoot with the right button.
- Once you have added all the players you want to add, press the top button to start the game.
- When you complete a game, you will be sent back to the menu, but you will need to re-add players.

Code Overview

Open Unity Hub and Click Add and select this project folder to add it to your list of projects and you can then open the project from Unity Hub.

Editable settings are in modules on the inspector window when selecting an object.

Look for these modules if you want to change these settings:

ShootBall.cs:

LaunchForce = change this value if you want to change the speed at which players fire projectiles.

GameTime.cs

gameLength = change this value if you want to change the length of the playtime. Useful if you are making the playfield longer.

CreateWall.cs

risingSpeed = change this value if you want to change the speed at which the target wall rises during the game

regionSize = change the Y value if you want to increase the length of the target wall to add more targets on generation. ((go back to createwall and set up the regionSize and backPoints variables to be dependant on a single float to determine backwall height))

PointCode.cs

pointValue = change this value if you want to change the amount of points each target is worth. There are 5 different targets so you are free to change all or none of them.

References:

[1] Burke, J. W., McNeill, M. D. J., Charles, D. K., Morrow, P. J., Crosbie, J. H., & McDonough, S. M. (2009). Optimising engagement for stroke rehabilitation using serious games. *The Visual Computer*, 25(12), 1085–1099. <https://doi.org/10.1007/s00371-009-0387-4>