

User Manual

Assignment 2

Joseph Bess, John Herrick, Laura Shub

Game Description

Our game is a generalization of classic Pong to a 3D world with an accurate physics simulation. As each round goes on, the ball slowly gains speed until eventually someone makes a mistake. We added a small twist however and introduced a color aspect. The ball is randomized to be either red, green, or blue and can only be hit by a paddle of the matching color. We added a set of controls for the player to change the color of the paddle so they can match the color of the ball, adding an additional challenge.

Objective

Just like classic Pong, the goal is to score points. The first player to 7 points is declared the winner. A point is scored by hitting the ball into your opponent's goal area without them hitting it back.

Controls

In between points the ball is at rest until the user presses `space`. This gives them a moment to collect themselves in between points.

The player controls the paddle by moving the mouse around the screen. If the ball collides with the paddle and the two colors match, the ball bounces. Otherwise, the ball explodes and a point is scored by the opponent. Pressing W, A, and D change the color of the paddle to blue, red, and green respectively, allowing the user to match the color of the paddle to the color of the ball.

While the game is playing, pressing `esc` pauses the game. Pressing `esc` while the game is playing will exit the game. While paused a GUI appears, allowing the user to resume, quit, and change volume controls.

Features and Extra Credit

We also mentioned these in the README.txt but we wanted to restate them here for ease of access.

Improved Camera Controls (1 pt)

- The main focus in Pong is controlling the paddle and the user shouldn't have to worry about controlling the camera. Because of this we decided to implement a camera system based around the movement of the paddle so that any time the user moves the paddle the camera intuitively follows. Additionally, when the user pauses the game with `esc`, the WASD EQ keys allow the user to move the camera in a global view throughout the game board. When the user resumes, the camera snaps back to the intuitive mode.

Background Music (2pts)

- We have included background music in our game. When the user pauses, an option pops up to control the volume of specifically the background music.

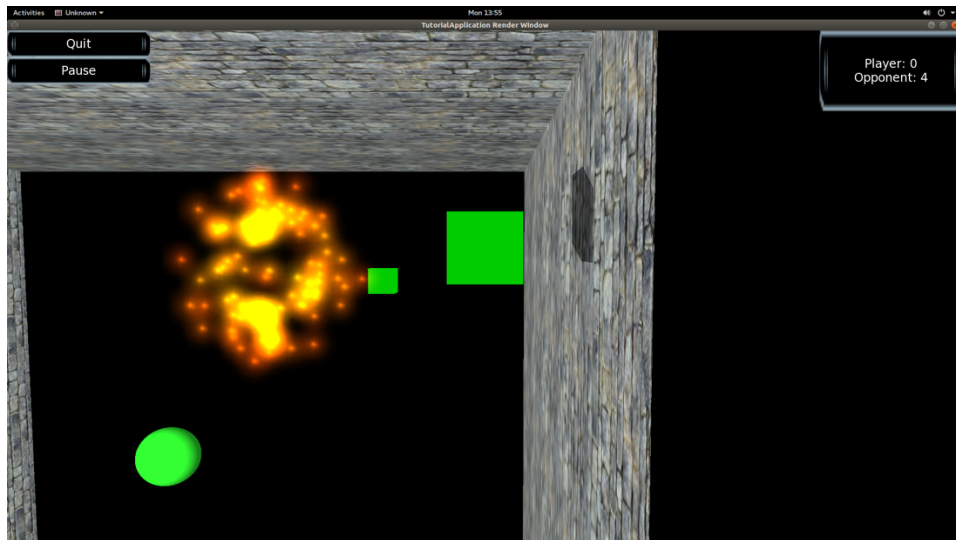
End Condition (2 pts)

- The game ends when one side reaches 7 points. Once that condition is met, the player is presented with an end of game screen displaying the score. If the player wins fireworks play as well as a victory sound.



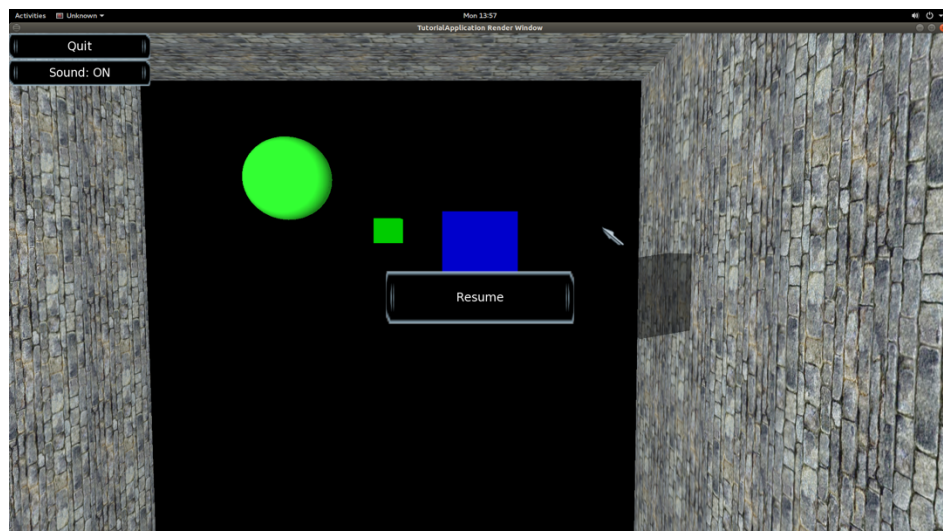
Particle Effects (2pt)

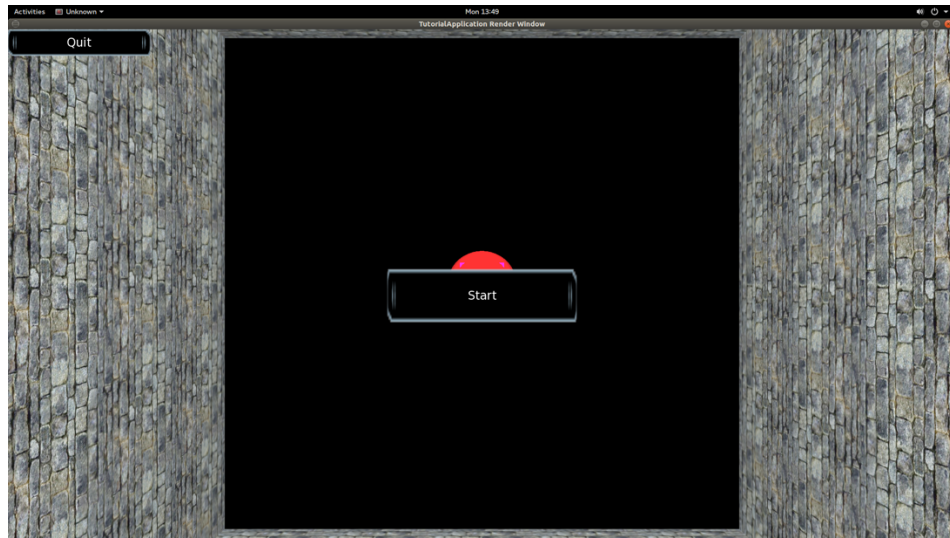
- We implemented particle effects and any time the player or opponent scores, the ball explodes into a buch of colorful particles. Additionally, when the player wins we play fireworks in the background, built using a series of particles and particle emmitters.



Player Interfaces (3 pts)

- Upon launching the game the user is presented with a start button which functions as the Main Menu. After clicking start, the game begins and the user can pause with `esc`. After pausing, the physics stop and the user has the option to change sound options, continue the game, and quit the game.





Controls Quick Reference

- `esc` - while playing pauses, while paused exits
- `WAD` - changes the color of the paddle while playing
- `WASDEQ` - controls the camera while paused
- `F` - artificially triggers fireworks for testing purposes
- `space` - gets the ball moving when it is standing still
- `mouse` - moves paddle while playing, interacts with GUI otherwise