

Jacob Blomquist u0843402

William Ludwig u0905218

Team Daedalus

Daedalus' Labyrinth

Final Report

Abstract:

Our project is a web maze game that keeps track of high scores and users. Our game is a labyrinth that the player needs to navigate through. There is a mini map that will help the user in their journey. The time is recorded and then the times are stored into a high score database. The fastest times are shown first. There are many added effects that make the game more interesting. We added scary music and rain sound and made it look like it was raining inside of the labyrinth. The 3D scene of the labyrinth is rendered using raycasting.

URL:

We were unable to publish our project to the AWS cloud so we do not have any running URLs.

Introduction:

As stated in our abstract our project is a web game in which the player needs to navigate through a labyrinth. The labyrinth is randomly generated using a depth first search recursive backtracker. We have added many extra features into the game to make it more fun and interesting. There is a timer that keeps track of how long the user has been trying to solve the labyrinth. Once they reach the end this score is converted to milliseconds and that is the score that is sent to the high score database along with their username.

We also implemented a mini map of the labyrinth which helps the user to make it through the labyrinth. The mini map starts out all dark and then as the user moves through the labyrinth the paths that they have already taken get marked on the mini map. This makes it so that it is easier to solve and the player will not keep trying to go down the same paths over and over again.

We have added many other features to the game to make it more exciting. We have added in spooky music and the sound of rain while the player is going through the labyrinth. We have also added in an effect that makes it look like it is raining. This was done by flashing transparent rectangles on the screen randomly. These rectangles had different shades of blues and this effect made it appear as if it was raining inside of the Labyrinth.

The 3D aspect of the labyrinth was implemented by doing raycasting. For each column of pixels, a ray is cast from the player until it intersects with a wall. The distance to that wall is recorded and used to calculate the height of a wall slice that is to be rendered. This makes a pseudo 3D effect. This is the same technique classic games like Wolfenstein 3D used.

Our game is backed by two databases. One database is used for keeping track of all of the high scores. The other database is used to keep track of the users and the roles that each user has. Only administrators have access to delete high scores. Players can play the game, see an overview of the web site, and look at the high scores table, but they are not able to delete high

scores. Both of these databases are seeded with a few values so that it can be clear that they are working right from the beginning.

Feature Table:

Feature Name	Scope	Primary Programmer	Time Spent	File/Function	LoC
GitHub SetUp	Other	Will & Jacob	.5 Hours	N/A	N/A
Creating a basic website	UI	Will	2 Hours	Many Files in model, view, controllers	75
Create highscore model and database for high scores	Backend	Will	2 Hours	Models/HighScores Data/HighScoreDB Data/DBInitializer	125
Implement Users and Roles DB and seed DB	Backend	Will	3 Hours	Scaffolding Data/DBInitializer	85
Attempting to set up AWS	Other	Will	7 Hours	N/A	N/A
Layout adjust of menus	UI	Will	1.5 Hours	_layout.cs	15

Admin high score page	UI and uses backend	Will	1.5 Hours	HomeController AdminHighScore view	30
Improving and Beautifying Web Site	UI	Will & Jacob	3 Hours	Views	50
Maze Generation	Game	Jacob	2.5	maze.js	151
Maze Normalization	Game	Jacob	1	squareMaze.js	52
Minimap Rendering	Game	Jacob	2	sketch.js	240
Player Physics / Collisions	Game	Jacob	4	sketch.js	27
3D Rendering	Game	Jacob	8	sketch.js	156
Rain Rendering	Game	Jacob	1	sketch.js	20
Timing / Win Condition	Game	Jacob	2	sketch.js	30
AJAX Score Recording	Game/Backend	Jacob	3	sketch.js, HomeController.cs	50
Styling Game Canvas in Site	UI	Jacob	1	Game.cshtml, site.css	100

Individual Contribution:

Team Member	Time Spent on Project	Lines of Code Committed
Jacob Blomquist	28	876
William Ludwig	20.5	280

William Ludwig:

I did most of the backend stuff with the databases and I also created the basics of the website and its format. We worked on design and beautification together later. I created the highscore object and created the database that would keep those high scores. I implemented the Users and Roles part of the web application. I seeded both of these databases. I also spent a lot of time of AWS trying to get that to work, but in the end it did not work.

Jacob Blomquist:

I worked mostly on the game itself. I created and tested the game using the [P5.js Web Editor](#). I implemented the maze generator, and raycasting rendering system. I also was in charge of connecting the game's win condition to the ASP.NET application using AJAX. Due to scaffolding, there is a disparity above in line of code committed. From what we can tell, our work was equal and our team worked well together.

Summary:

Our team project was very enjoyable to create. Our team worked extremely well together and we split the work up well. Since the very beginning we had good communication and understood what we wanted to get done and how we were going to get it done. The only obstacle that we came across was deploying to AWS. We worked for many hours on this. We went through the steps many different times and it just never worked.

This was a good final project for us because we ended up using most of what we learned this semester while completing this project. We used HTML, CSS, Databases, AJAX,

JavaScript, Identity, Scaffolding, Migrations, etc. This project helped us to have more practice with each of these topics and to really understand them even better.