

Software DMT

By Kyle Leonesio, William McLaughlin, Sam Britten, Joe Taylor

Goal: Build a 3d game in Unity, with an accompanying website, and integrate functions such as database storage and user authentication.

Methodology: Modular, iterative process.

Website

Tools Used: VSCode(5star), Github(4star), Heroku(3star), Postgresql(3star)

Stack: HTML, JS, EJS



Webpage was developed through an iterative process where many approaches were attempted before reaching the final version. Some of the attempted/abandoned versions were based around react native and “single-page” design, and HTML/JS template languages such as pugjs.

Tools Used- Sam Britten

Blender- 5 Stars



Blender was used for 3D modeling, texturing, unwrapping, and animating.

Krita- 5 Stars



Krita was used to draw textures and create normal maps.

Quixel- 5 stars



Quixel bridge and mixer were used to download and edit photo scans.

Visual Studio- 2 Stars



Visual studio was used to create the assets webpage

Github- 3 Stars



Used to backup and store assets and code

Asset tools (continued)

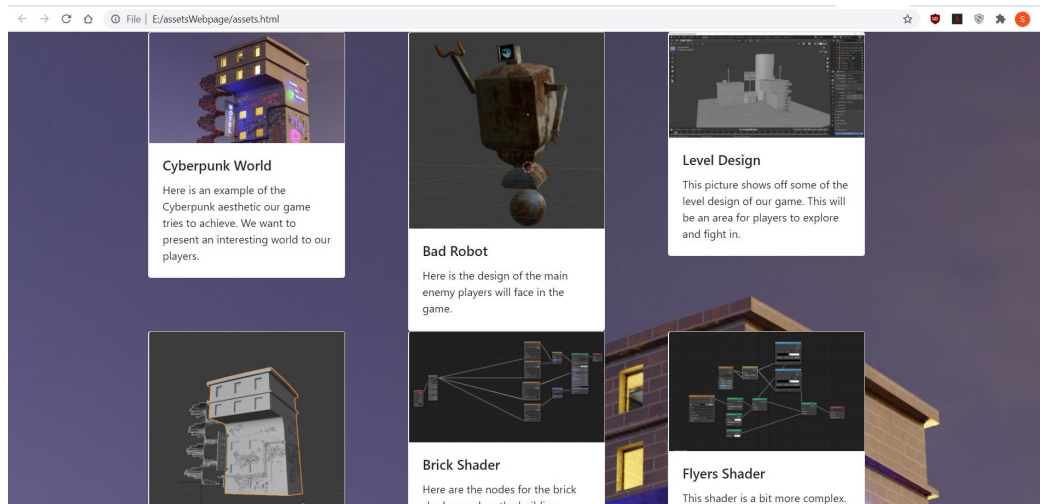
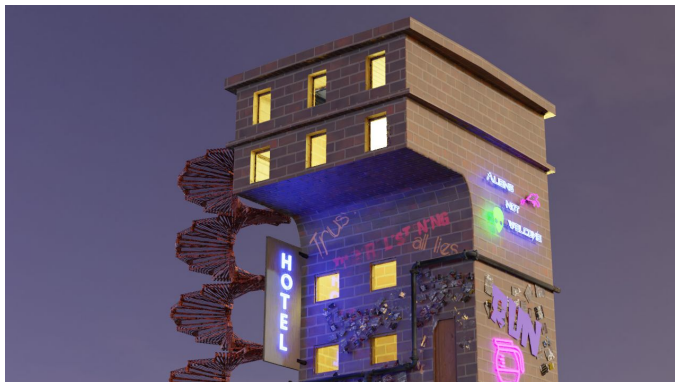
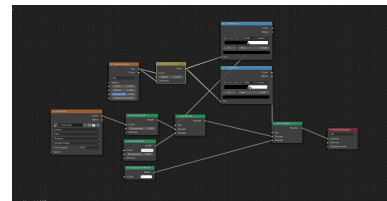
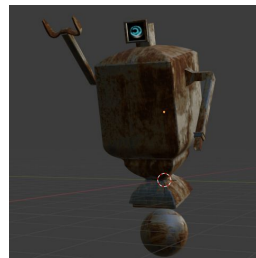
Unity- 4 Stars

Unity is used to render assets in-game.



Unity Collab- 1 Star

Used as version control for the game.



Development Tools Used - William McLaughlin



Unity [4]: game design engine with built in physics and renderers as well as an editor for creating the game.

Unity Collab [3]: github like code sharing system built into unity for sharing the project across devices and users.



Heroku [3]: hosting service for the website and database

PostgreSQL / pgAdmin [4]: database management system



Github [2]: storage and sharing system, used for website and project code backups.

C# [5]: Coding language used in Unity scripts.

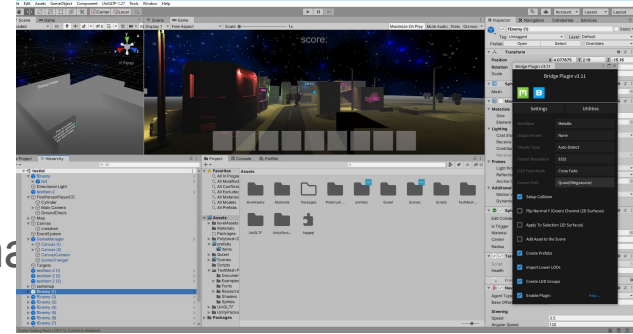


Project: Menu, Players, and Enemies

Main Menu: Simplistic menu allowing the player to start the game or login to their account made on the website.

Player: A controllable character that can navigate the map and shoot at enemies - run, jump, shoot, turn, take damage.

Enemies: Characters that follow and attempt to attack the player which must be killed to increase score - move, follow, attack, take damage.



Current User:

Main Menu

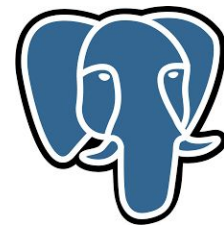
Username:

Password:

Login

Kyle leonesio Development Tools

1. Unity - 4 stars
 - a. Very useful had functionality in it that would be almost impossible without it
 - b. latter half of using in it started crashing
2. Unity collab - 2 stars
 - a. Collab allowed for quick updates between us
 - b. collaboration tool was very finicky and broke with unity updates
3. c# - 5 stars
 - a. c# was the language use in unity and it is a good choice easy to use and learn and fit the game design problem nicely
4. Postgre - 4 stars
 - a. Postgre was used so unity could send score to the site
 - b. It was hard to get working at first because it required separate download to work with c# but it was working it had no issues
5. Heroku - 4 stars
 - a. Where the database was hosted
 - b. Worked fine but pain to learn
6. Github - 3 stars
 - a. Good place to store and share some files
 - b. Didn't mess too well with unity and had to manually upload

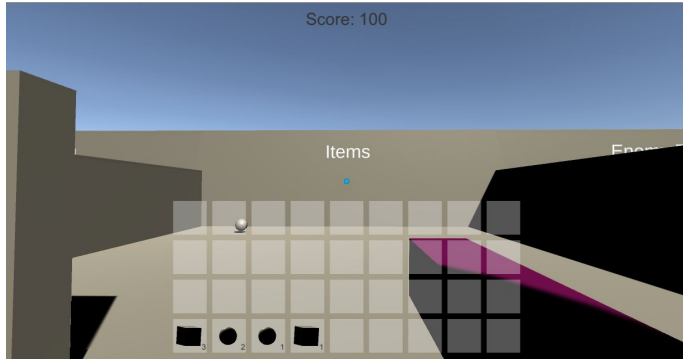


Kyle items/scoring

Item on the ground can be picked up and put in you inventory

Score would be added as you kill enemies indicated at the top

Once player dies by either enemies or falling off give a screen to submit scores then end game



Score:100

Submit

Our Team's Biggest Challenges

Technical ability - learning webdev

Unity collab - initial set up, syncing issues, version control

Postgres initial setup - unity connection

Getting Blender shaders to work properly in Unity

Transferring animations from Blender to Unity