**Title:** Software DMT

Who: William McLaughlin, Sam Britten, Joe Taylor, Kyle Leonesio

Project Description: SoftwareDMT is making a casual, 3D physics-based "rogue-like" game called SoftwareQuest that plays across brief episode-like levels with a self-aware software development theme that introduces the player to elements of the software design and production process.

Gameplay is a physics-based action-adventure where the player controls an avatar that must navigate through an environment, fighting enemies in real-time using items and powerups collected. Adversaries, equipment, power ups etc, will follow a software-development theme such as collecting skills like "Agile" (the agile development method), finding companions like the "3D modeler", and overcoming obstacles like the "Clueless producer."

SoftwareQuest is for gamers, who want to have fun in between classes or assignments while picking up industry knowledge, the game "Software Quest" is a 3D "Rogue-like," that is entertaining but also informative and inspirational. Unlike the leading rogue-likes such as "Risk of Rain 2" that challenge a player to merely overcome an evil adversary, our product challenges the player to assemble a team of talented developers, collect valuable skills and methodologies used in the industry, and implement them in an effective way to successfully develop a project.

Project Tracker: https://trello.com/b/Xm0X1gIM/204-5-project
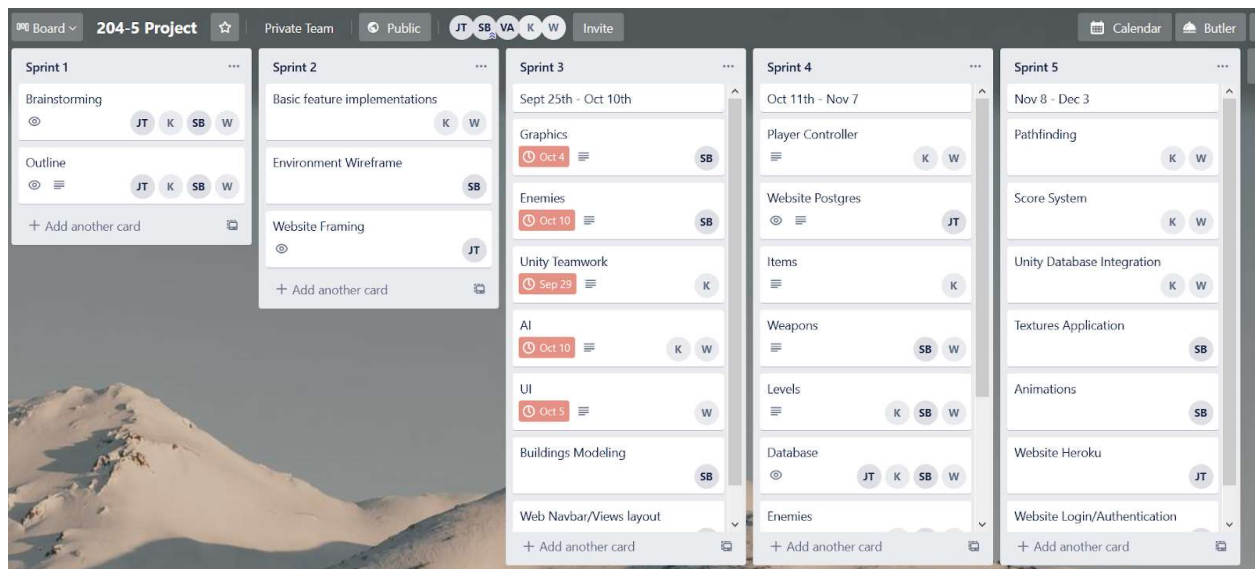
VCS: https://github.com/CSCI-3308-CU-Boulder/204-5
(Note: as noted in the readme, the github was only used for storing some code and backups, Unity Collab was used as the primary VCS/file sharing for the unity project, as there is no public link for this, screenshots from commits within that program are below.
Website was developed by a single member and stored locally on machine with example/demo and final version pushed to git, and hosted on heroku, screenshots below)

Contributions:

Kyle Leonesio:
Technologies: Unity (C#), Unity Collab, PostgreSQL/pgAdmin, Heroku, visual studio, github.
I Worked Mostly in unity but also worked with postgre and heroku. My primary contribution was getting inventory working and the ability to pick up items in game. I also made the death scene with the ability to submit scores and made the hud display for current score. Lastly I worked with William to get database integration set up so I could submit scores and he could login.

William McLaughlin:
Technologies: Unity (C#), Unity Collab, PostgreSQL/pgAdmin, Heroku, Visual Studio
Features: Player movement / shooting, enemy movement / damage, game login
Summary: I worked primarily in unity building the game, my focus was on player and enemy movement. I set up a player character that can walk around and shoot, as well as take damage; then made enemies which could move and follow the player, as well as attack the player and take damage from the player. I also built the game login, allowing the user to play and store scores under their account. Kyle and I worked together on the initial unity to database integration.

Sam Britten:

Technologies: Blender, Krita, Quixel Bridge, Quixel Mixer, Unity, Unity Collab, Visual Studio, Github
Languages: HTML, Python (Blender Scripting), C (wrote script for game installer)
My contribution to this project was primarily to make assets for the game. The main software I used for this was Blender, where I modelled, unwrapped, and textured assets, and animated characters. I also created and edited textures using Quixel and Krita. I then imported my assets to Unity where they are rendered, and used Unity collab as version control and to share my work with my teammates. In addition, I created a small portion of the website (assets page). Finally, I wrote a C script to create an installer for the game so it is easy for users to set up.

Joe Taylor:
      Technologies: VSCode, Github, Trello, Heroku, Postgresql, HTML+JS, EJS
      Features:
- Project Management / Trello board and github
- Created the webpage through an iterative experimental process while learning webdev. Iterations included a framework in react/react native, JS single-page design, HTML/JS templates such as pug/pugjs.
- Experimented with database design in MySQL and locally stored database functionality.
- Switched to postgresql database for heroku hosting and website integration.
- Website:
  - A leaderboard page displaying the highest all-time scores in the game, pulls from the remote heroku database to stay updated.

- A registration page so users can sign-up with a username and password. (To avoid bots and spam downloads etc.)
- A wiki-like page with gameplay-related information and design process
- Game download link for the user to be able to download the game
  - Database:
    - High score storage with two columns: username and score achieved.
    - User registration information stored on a separate table.
    - Website will call to this database to update the leaderboard page

Commit Screenshots:

Unity Collab Commit History:

Sam Britten Commit Screenshots:

Commits on Nov 13, 2020

Add files via upload ...
ssaammbb committed 24 days ago
Verified | 6599428

Commits on Oct 30, 2020

Add files via upload
ssaammbb committed on Oct 30
Verified | 718dc99

Commits on Oct 16, 2020

Add files via upload ...
ssaammbb committed on Oct 16
Verified | 63bbd6e

Commits on Sep 17, 2020

test4
ssaammbb committed on Sep 17
2b8d088

working
ssaammbb committed on Sep 17
588625e

tets
ssaammbb committed on Sep 17
6c6badf

Sam
ssaammbb committed on Sep 17
d79ca21

Commits on Dec 7, 2020

Add files via upload
ssaammbb committed 2 minutes ago
Verified | 4b18f28 | <>

Add files via upload
ssaammbb committed 5 minutes ago
Verified | 02295fc | <>

Will and Kyle's Script Commits:

Add files via upload

Browse files

Note: these scripts don't work on their own, they work in the unity project we have created, thye do not show all the work done on the project but at least show some of the code being done.

⌥ master

🔴 kyleJL314 committed on Oct 30 [Verified]   1 parent e8772d4   commit 8ad2469e2df7c48b15c723b6143ef1b4b2a5b84d

Showing 6 changed files with 264 additions and 0 deletions.   Unified | Split

> 23 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/GameData.cs 📋   ...

> 11 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/ItemBeavior.cs 📋   ...

> 142 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/inventoryManager.cs 📋   ...

> 35 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/item.cs 📋   ...

> 13 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/sceneChanger.cs 📋   ...

> 40 ⬛⬛⬛⬛⬛ Code Components/kyle's_Code/testSql.cs 📋   ...

final scripts from unity projects

Browse files

added to properly named folder and deleted old versions, please note the scripts do not work independently of the unity project they were created in.

⌥ master

🔴 WillM5462 committed 3 days ago [Verified]   1 parent 34793df   commit b7ef3f33cbe23520e8b8697116c559e9d2916c60

Showing 7 changed files with 311 additions and 0 deletions.   Unified | Split

> 37 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/gun.cs 📋   ...

> 58 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/login.cs 📋   ...

> 30 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/mainmenu.cs 📋   ...

> 37 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/mouselook.cs 📋   ...

> 62 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/playermovescript.cs 📋   ...

> 48 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/target.cs 📋   ...

> 39 ⬛⬛⬛⬛⬛ Code Components/Will's_Scripts/targetmove.cs 📋   ...

Joe Taylor's heroku/github builds/commits:

jota2125@colorado.edu: Build succeeded
Dec 3 at 6:27 PM · View build log

jota2125@colorado.edu: Deployed a23c31c8
Dec 3 at 3:56 AM · v10 · Roll back to here

jota2125@colorado.edu: Build succeeded
Dec 3 at 3:56 AM · View build log

jota2125@colorado.edu: Deployed 6daefaa7
Nov 19 at 5:13 AM · v9 · Roll back to here

jota2125@colorado.edu: Build succeeded
Nov 19 at 5:13 AM · View build log

jota2125@colorado.edu: Deployed 362fa188
Nov 19 at 4:55 AM · v8 · Roll back to here

jota2125@colorado.edu: Build succeeded

Deployment:
Go to the app website: https://joe-cu-app.herokuapp.com/ Create or login to an account, and click the link to a Google Drive folder in the top right. This will take you to a download for the installer, which will walk you through the install process. Run the game, login to your account in game, and play until you die (hit by 10 enemies, or fall off the map). Press submit to send your score with your username to the leaderboard, the leaderboard can be seen on the website.

You can also visit this link to download the game directly:
https://drive.google.com/file/d/1nW8lO7_QOgFZxal0LirDT5yz502AR5kV/view?usp=sharing