Steve Jobs' Rejected Interns

Christopher Gonzalez-Millan
Connor Eamon
Matthew White
Peter LeCavalier
Michael Sciarabba
Remy Vancil

Short Fuze

May 4, 2020

Product Description

Short Fuze is a 2D platformer-type game where Short Fuze, our main character, is tasked with collecting as many coins as possible within a given time period. These coins are scattered throughout a level which consists of various platforms, so the challenge is to traverse this environment as efficiently as possible while also collecting as many coins as possible. Once the player's time has run out, the game ends and the player's score is automatically registered in a database and displayed on a scoreboard, which records and outputs the scores of all players over time. The scoreboard also displays the max number of coins collected by that player in a single game, the total number of coins collected, and the total number of games they've played. This allows players to compete against each other or with themselves.

Project Tracker

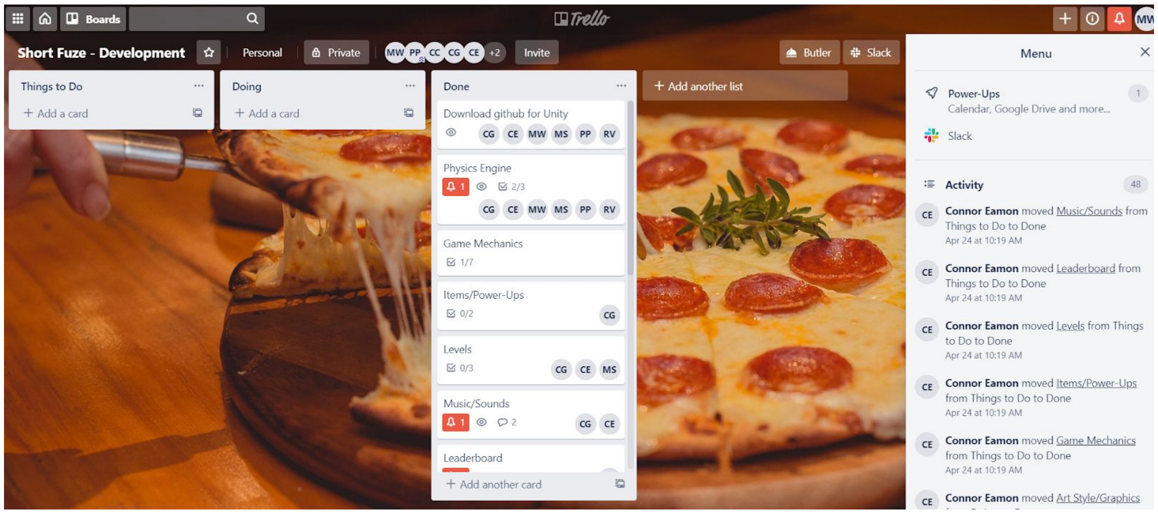
We used Trello, Slack, and GroupMe for project management and communication.

Link to development on Trello:

https://trello.com/b/j0ISDXCU/short-fuze-development

Short Fuze Trello Homepage:

Trello was used to keep track of individual game mechanics in order to delegate responsibility.



Things to Do
+ Add a card

Doing
+ Add a card

Done
- Download github for Unity — CG CE MW MS PP RV
- Physics Engine — 1, 2/3 — CG CE MW MS PP RV
- Game Mechanics — 1/7
- Items/Power-Ups — 0/2 — CG
- Levels — 0/3 — CG CE MS
- Music/Sounds — 1, 2 — CG CE
- Leaderboard
+ Add another card

CE **Connor Eamon** moved Music/Sounds from Things to Do to Done
Apr 24 at 10:19 AM

CE **Connor Eamon** moved Leaderboard from Things to Do to Done
Apr 24 at 10:19 AM

CE **Connor Eamon** moved Levels from Things to Do to Done
Apr 24 at 10:19 AM

CE **Connor Eamon** moved Items/Power-Ups from Things to Do to Done
Apr 24 at 10:19 AM

CE **Connor Eamon** moved Game Mechanics from Things to Do to Done
Apr 24 at 10:19 AM

CE **Connor Eamon** moved Art Style/Graphics

VCS

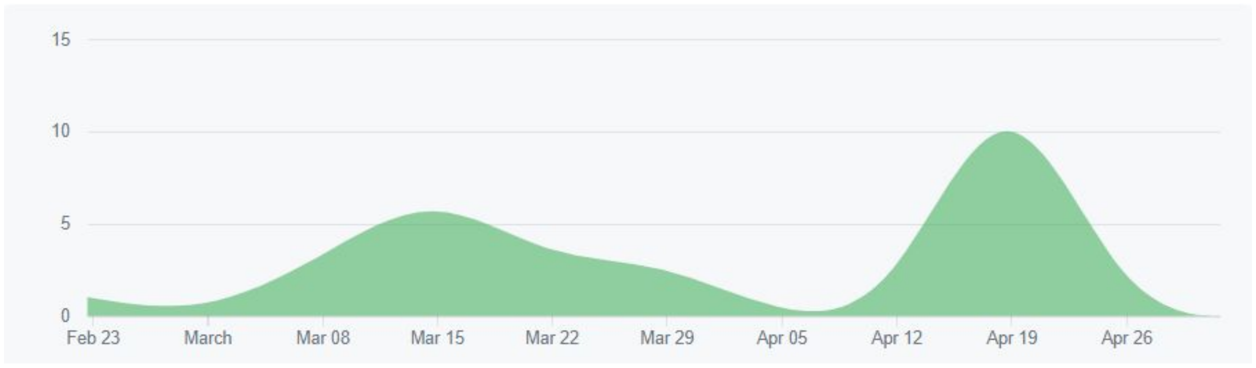
Project code: https://github.com/pele6150/3308_102-5_project_code

Milestones: https://github.com/pele6150/3308_102-5_milestones

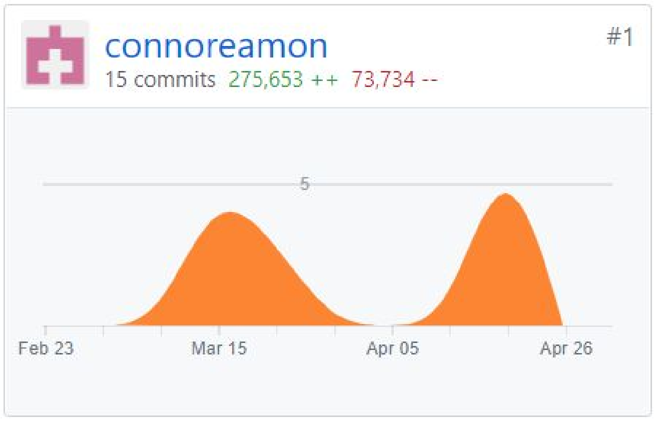
Meeting Logs: https://github.com/pele6150/3308_102-5_meeting_logs

Contributions

Short Fuze Github Commits from February to May 2020



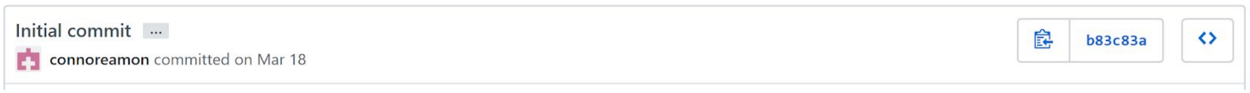
Connor Eamon

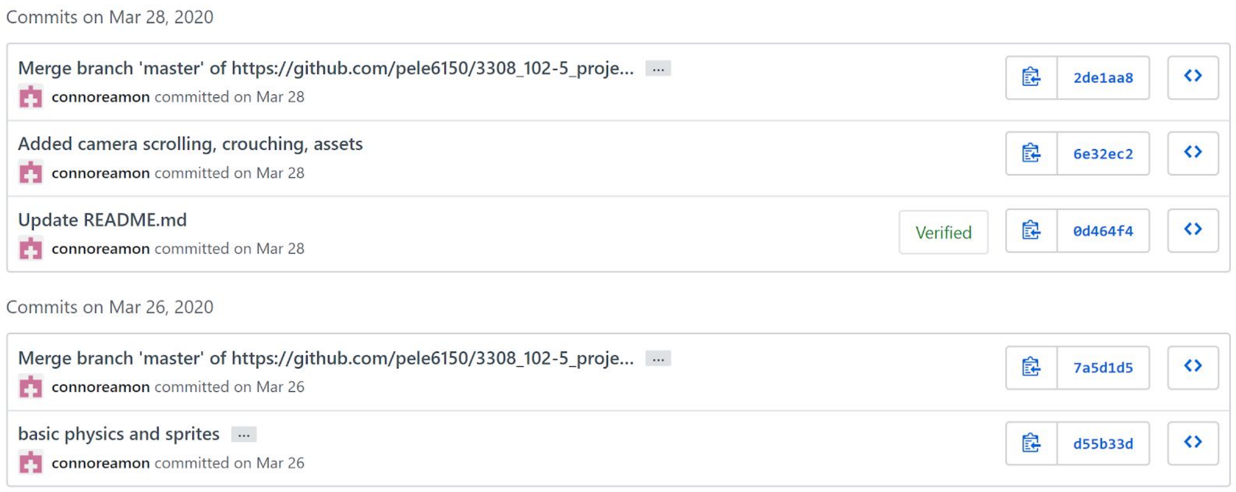

connoreamon

15 commits 275,653 ++ 73,734 --

#1

Wrote player movement, physics, collision detection, and level mechanics. Created coins and coin collection mechanics, as well as in game score keeping. Worked on final level design. I designed the blocks and the grid that make up the levels as well. Also created the prototype main character sprite, which was later beautified and animated. Most of my work was in C# in Unity and visual studio. Some work on graphics was done in an external graphics editor.


Initial commit …
connoreamon committed on Mar 18
b83c83a


Commits on Mar 28, 2020

Merge branch 'master' of https://github.com/pele6150/3308_102-5_proje… …
connoreamon committed on Mar 28
2de1aa8

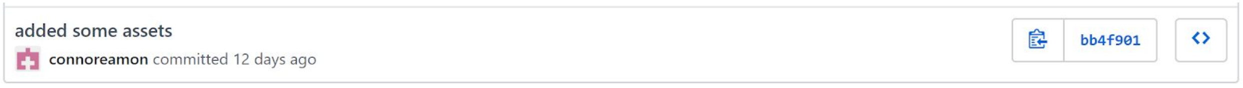
Added camera scrolling, crouching, assets
connoreamon committed on Mar 28
6e32ec2

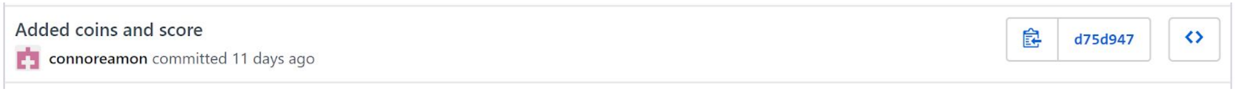
Update README.md
connoreamon committed on Mar 28
Verified
0d464f4

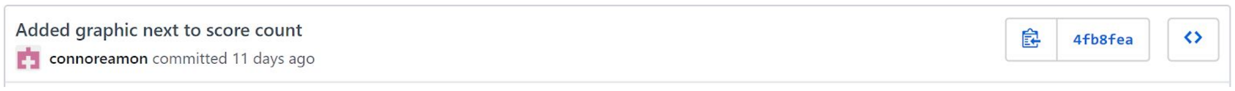
Commits on Mar 26, 2020

Merge branch 'master' of https://github.com/pele6150/3308_102-5_proje… …
connoreamon committed on Mar 26
7a5d1d5

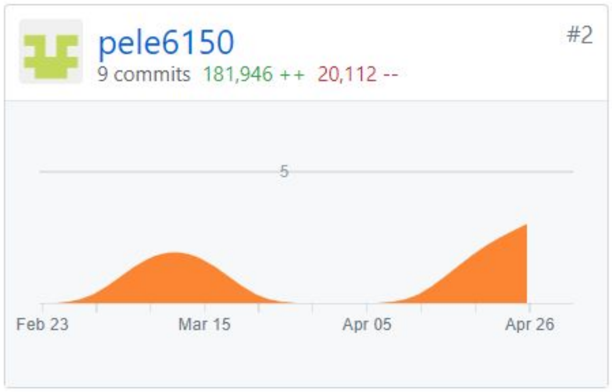
basic physics and sprites …
connoreamon committed on Mar 26
d55b33d


added some assets
connoreamon committed 12 days ago
bb4f901


Added coins and score
connoreamon committed 11 days ago
d75d947


Added graphic next to score count
connoreamon committed 11 days ago
4fb8fea

Peter LeCavalier



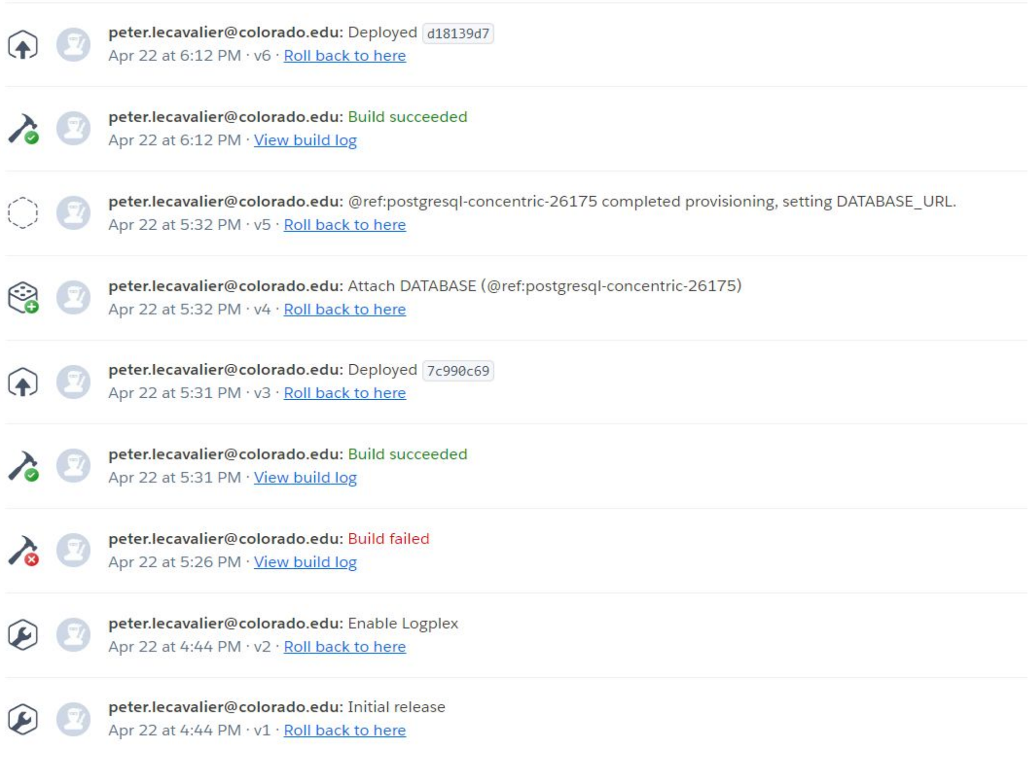
Worked on integration with scoreboard into the actual game executable

Added game over screen

C# scripts to integrate everything with PostgreSQL

Finishing touches on the game to ensure the final executable works

Large chunk of my development was on a separate heroku git repo, see below:



peter.lecavalier@colorado.edu: Deployed d18139d7
Apr 22 at 6:12 PM · v6 · Roll back to here

peter.lecavalier@colorado.edu: Build succeeded
Apr 22 at 6:12 PM · View build log

peter.lecavalier@colorado.edu: @ref:postgresql-concentric-26175 completed provisioning, setting DATABASE_URL.
Apr 22 at 5:32 PM · v5 · Roll back to here

peter.lecavalier@colorado.edu: Attach DATABASE (@ref:postgresql-concentric-26175)
Apr 22 at 5:32 PM · v4 · Roll back to here

peter.lecavalier@colorado.edu: Deployed 7c990c69
Apr 22 at 5:31 PM · v3 · Roll back to here

peter.lecavalier@colorado.edu: Build succeeded
Apr 22 at 5:31 PM · View build log

peter.lecavalier@colorado.edu: Build failed
Apr 22 at 5:26 PM · View build log

peter.lecavalier@colorado.edu: Enable Logplex
Apr 22 at 4:44 PM · v2 · Roll back to here

peter.lecavalier@colorado.edu: Initial release
Apr 22 at 4:44 PM · v1 · Roll back to here

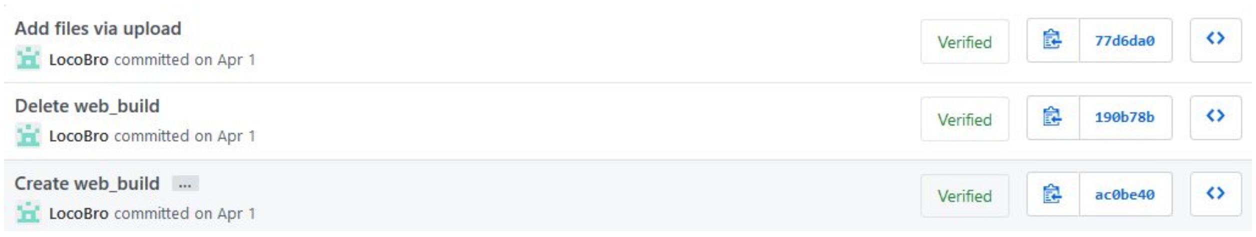
Above is the very first snippet of the activity from the heroku website..

For heroku, I integrated PostgreSQL with a working website congruently with the Unity development

I coded in JS/NodeJS/EJS to get working pages that interacted with the SQL database.

Michael Sciarabba

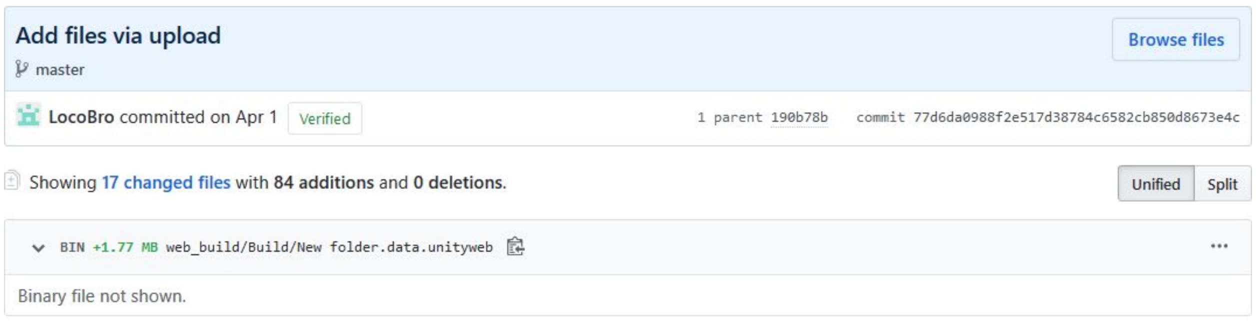
In Short Fuze, I (Michael Sciarabba) had the task of creating builds for deployment as well as creating an interactive login screen.



Add files via upload
LocoBro committed on Apr 1 — Verified — 77d6da0

Delete web_build
LocoBro committed on Apr 1 — Verified — 190b78b

Create web_build ...
LocoBro committed on Apr 1 — Verified — ac0be40



Add files via upload
master

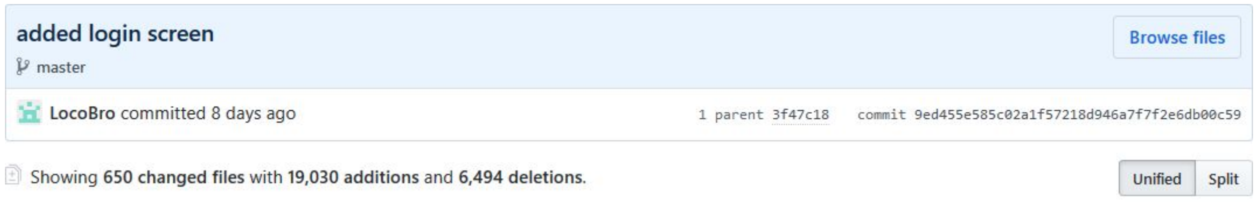
LocoBro committed on Apr 1 Verified 1 parent 190b78b commit 77d6da0988f2e517d38784c6582cb850d8673e4c

Showing 17 changed files with 84 additions and 0 deletions.

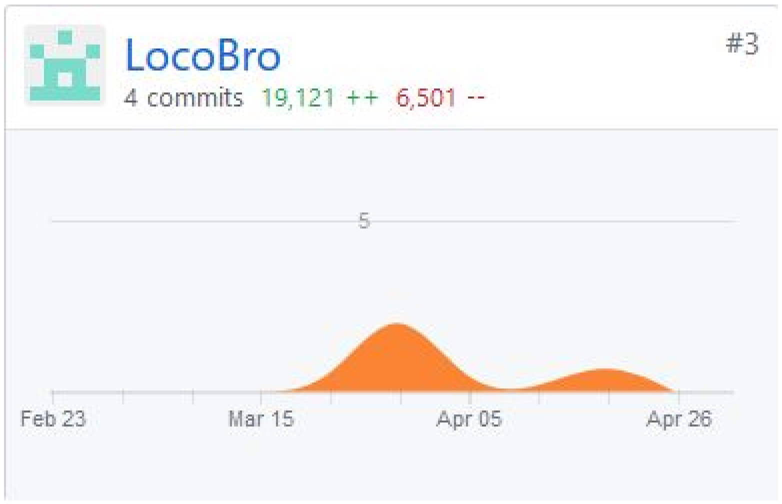
BIN +1.77 MB web_build/Build/New folder.data.unityweb

Binary file not shown.

For the actual building and packaging of the software, we created a WebGL compatible web player for a test environment to see how this type of deployment would work for the product we are designing. As seen above, I built and tested a few forms of the web build, but this form of deployment relied on hosting a large file for assets and scripts to run Short Fuze. After this test, we found that using an executable to run Short Fuze would be the best way to ensure that loading times are limited and the user has a better experience with the game.

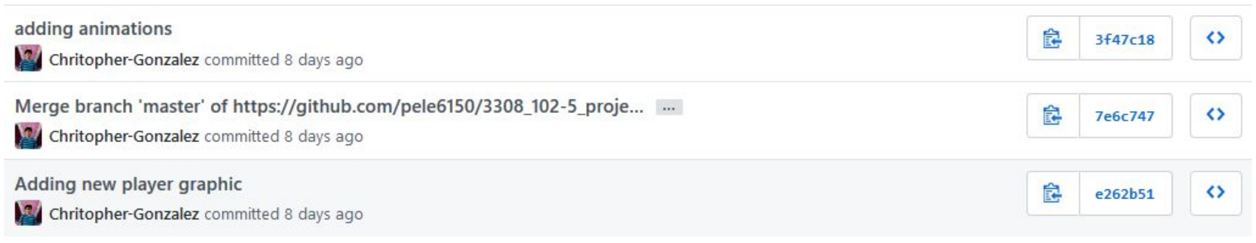

added login screen
master
Browse files
LocoBro committed 8 days ago
1 parent 3f47c18 commit 9ed455e585c02a1f57218d946a7f7f2e6db00c59
Showing **650 changed files** with **19,030 additions** and **6,494 deletions**.
Unified Split

Another of my commits was the login screen. This took a fair bit of time in the project. In our game, Short Fuze, there are 2 scenes used in Unity. Each of these scenes carries with it a handful of C# scripts to essentially dictate commands and states within the game. The login page contained scripts to change between scenes so that once the username and password were inputted, the script would save the input fields and change the scene to start the game. The login page input fields were checked against an existing user base in a text file, but due to privacy concerns we decided not to implement the password save. The username was then saved into a GameObject to be referenced when updating the database for the leaderboard. The login page, upon filling out the required information, would manage the current scene and begin the game.



My commits throughout the semester show a few of our two week sprints in which we added multiple new versions. Throughout the project we worked well together integrating each piece using version control. Another aspect of my contribution in this project was to help initiate contact amongst our team during quarantine. We worked very well when we pushed ourselves to meet.

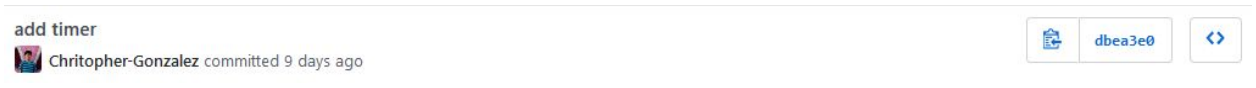
Christopher Gonzalez-Millan



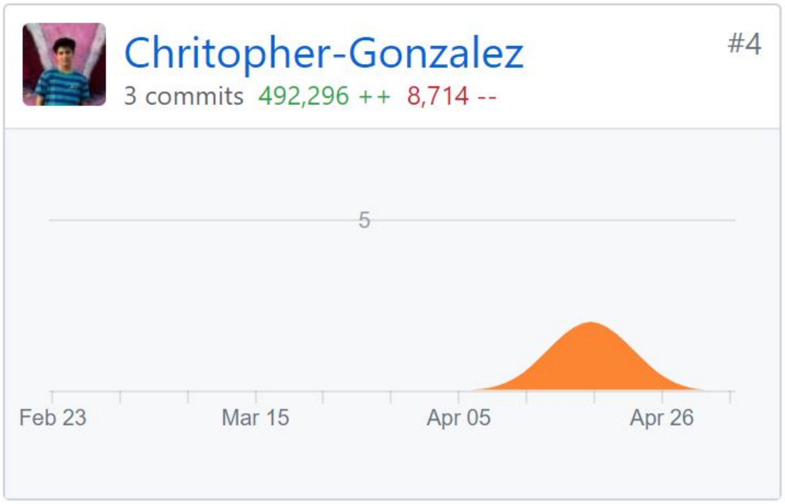
adding animations
Chritopher-Gonzalez committed 8 days ago — 3f47c18

Merge branch 'master' of https://github.com/pele6150/3308_102-5_proje... …
Chritopher-Gonzalez committed 8 days ago — 7e6c747

Adding new player graphic
Chritopher-Gonzalez committed 8 days ago — e262b51



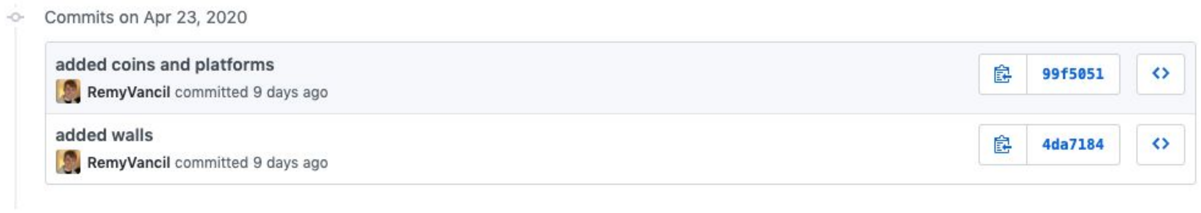
add timer
Chritopher-Gonzalez committed 9 days ago — dbea3e0



One of my major contributions to the project was an in game feature. I created a timer that displayed on the top left corner of the screen how much time the player still had to complete the current level. This was accomplished by a C# script in Unity that also set up the transition to the end game screen.

My next major contribution to the project were the player graphics which I created completely in photoshop. These were then imported into Unity where I used Anima2D and some C# scripts to develop running, idle, jumping and crouching animations.

Remy Vancil



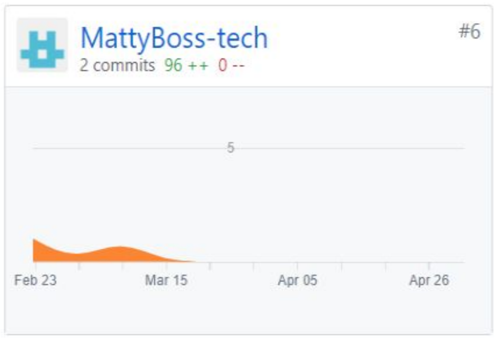
Commits on Apr 23, 2020

added coins and platforms
RemyVancil committed 9 days ago
99f5051

added walls
RemyVancil committed 9 days ago
4da7184

For our game Short Fuze, I (Remy Vancil) was in charge of designing the main level and placing the coins so that they could be collected by the player; I also added walls to box in the level and prevent players from leaving the level. My initial design for the level proved to be too large for our purposes so I scaled it back for a tighter, better-paced experience. Outside of the game I also committed documents, including meeting logs and milestones, and helped to schedule and coordinate meetings.

Matthew White



Matt acted as a scribe helping to organize and direct ideas and approaches in our meetings. Took notes for the team, and kept our thoughts together and made sure that our team’s notes were accessible to everyone. He helped to manage and track our teams progress as our features moved and changed.

Helped to schedule meetings and encourage coordination throughout the transition from in person to remote learning. Created zoom calls and Library room reservations to ensure quality levels of communications as the project progressed.

Matt worked with Mike to create the initial web build and test and test deployment methods. He created the early test platformer to give our team an early game to start developing and testing our C# scripts in early stages while the asset teams were building our game.

Deployment

1. Game Deployment:

   To deploy Short Fuze, we found that the best option would be to use an executable file to package the game and its assets. When working in Unity, we found that using its building settings, we could create a Windows build of our game. This creates an executable file in which our game can be run with all of the proper assets including textures, animations, physics mechanics, and C# scripts to work in tandem with the leaderboard.

2. Leaderboard deployment: https://short-fuze.herokuapp.com/

   The leaderboard was deployed through Heroku. With Peter taking the lead on this section, we were able to get a functioning database that would receive users and score updates from the game and add them on the server side. The leaderboard is presented using NodeJS/EJS form so that the user has an easier experience interacting with other users and creating a competitive environment. Peter's server side code lets the leaderboard update after each game so the user can see their improvement in real time.