TP1: Project Proposal

Patrick Lucas
ID: pllucas

Project Description:

“Jetpack Joyride” - My project will be a slightly different version of the original Jetpack Joyride game. The main goal of this game is to travel as far as you can by surviving as long as you can. The main tool for survival is the character’s jetpack, which can boost the character up–or allow the character to fall when not used. The character must go up and down, dodging a moving screen of randomly generated zappers while collecting coins along the way. Periodically throughout the game there will be missiles that give a warning and then shoot across the screen, and there will be red lasers that span the whole screen and move up and down turning on and off (just like in the original game). Finally, the game speed will gradually increase, making it harder to survive as you go.

Once all of these basic components of the game are programmed, there are several things that I will consider adding to the game. First, I would like to add a few gadgets to the game. This could include magnetism, which would pull coins toward the character, improved jetpack and fall speed, a second life chance, and several others (there’s tons of possibilities). I would also like to add a game menu before the game, and a shop where you can purchase the gadgets and different jetpack skins using the in-game coins. Finally, I could try to add vehicles like in the original game. This all depends on how long it takes me to finish the basic components of the game.

Similar Projects:

- “Barry; Jetpacks; and Friends” - Aashav Mehta

My project will be very similar to Aashav’s project because we are both modeling our projects after the same game. I will probably make mine more similar to the original game and not include the multiplayer version in their project. I like the use of a coin radius to simulate a magnet, and I think I will also use this concept in my project. However, I think I will make my gadgets something that you select before you play–like in the original game–rather than having collectable power ups along the way, like Aashav has.

- “jetpackman” - David Zhu

I think this project does a very good job of simulating the jetpack boost and gravity effects. I would like to incorporate similar effects to my game so that my character moves smoothly and the visual effect of the jetpack being on and off is very clear. This is something that I think the “Barry; Jetpacks; and Friends” game could have improved upon.

Structural Plan:

I plan on creating classes for all of my obstacles. This will allow me to create zappers of different lengths/angles, missiles with different speeds, and different types of lasers. I could create a jetpack class so that I can modify different skins of jetpacks as well as the fuel that they use. I may also choose to make a class for gadgets who's gadget attribute will change depending on what gadget it is. I will make functions for the moving app background and call it in redrawAll when the game is being played. There will be a function for the coins, and they will be drawn in a 2D list of a random length and random height between 1 and 3 rows. The fill of the coins will change to None if intersected with the character. I will create a function for a death animation and set it to True when the character is hit by an obstacle–as well as a gameOver screen. When the game is not being played, I will have a function for the app background in the menu and the shop. I will utilize onMousePress to navigate the menu and shop. onKeyPress will move the character up and down on the screen and be limited by the app.floor and app.ceiling.. onStep will move the screen to the left with a gradually increasing speed(dx). I will also probably use onStep to move the character up and down based off of the jetpack ON/OFF status. Finally, I will probably import images that are from the real game so that the project looks good.

<u>Algorithmic Plan:</u>

I think one of the most difficult parts of this project will be simulating gravity as well as the lag after the jetpack fuel is released (like how the character floats before it falls back down). I will have to create a character dy that will change the vertical position of my character by that amount in onStep, but I will also need to create a d^2y that changes my dy based on how many onStep calls there have been. This is my general approach for this problem, but I may find that there are more effective ways of creating in game gravity. I could also use this concept in my death animation in the x direction so that the app slows down when the character is hit. Regardless, I think this will be difficult, but if done correctly, it will make the game look much better.

I also think it will be hard to come up with a good random generation of my zappers, missiles, and lasers, as well as the timing of when the missiles and lasers will come in. I need to make sure that the game isn't impossible, but it also has to be challenging. Lastly, it may be difficult to create multiple gadgets.

<u>Timeline Plan:</u>

<u>Prior to TP1:</u>

- Classes for zappers, missiles, and lasers
- Moving background working with onStep and my character moving up and down with onKeyPress
- Character bounded by the ceiling and floor
- Function for coins and drawing packs of coins using a list

<u>TP1 to TP2:</u>

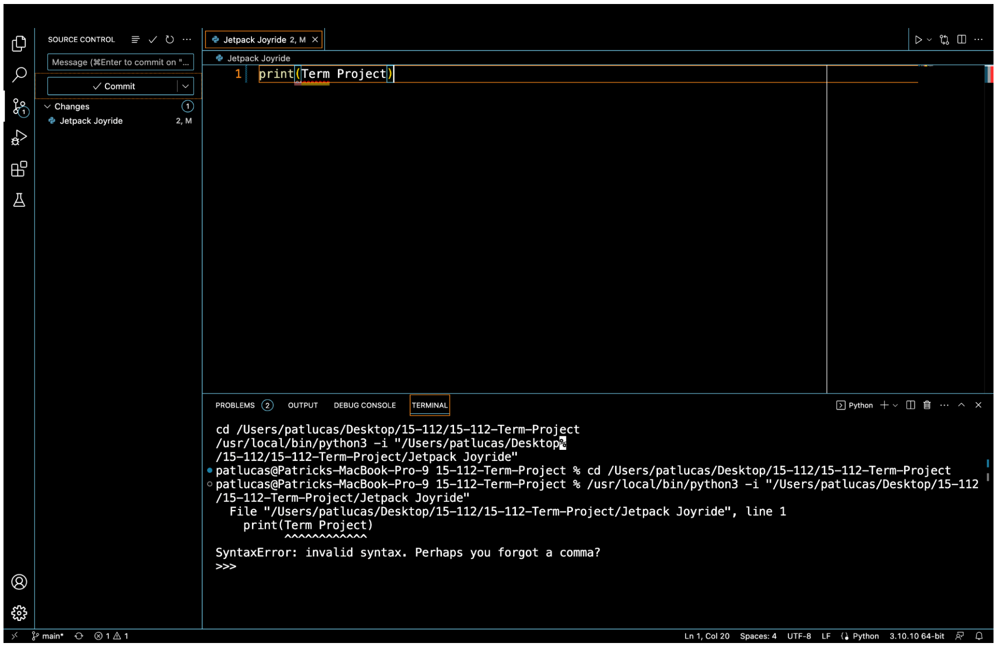
- Smooth gravity and floating effect
- At least 2 gadgets
- Character and object interactions and death animations
- Distance Tracker and Coin counter

TP2 to TP3:

- More Gadgets
- Menu and Shop
- Jetpack Skins
- Vehicles?
- Blaster?

I want to have a lot done before TP1 because I have two tests the week between TP1 and TP2.

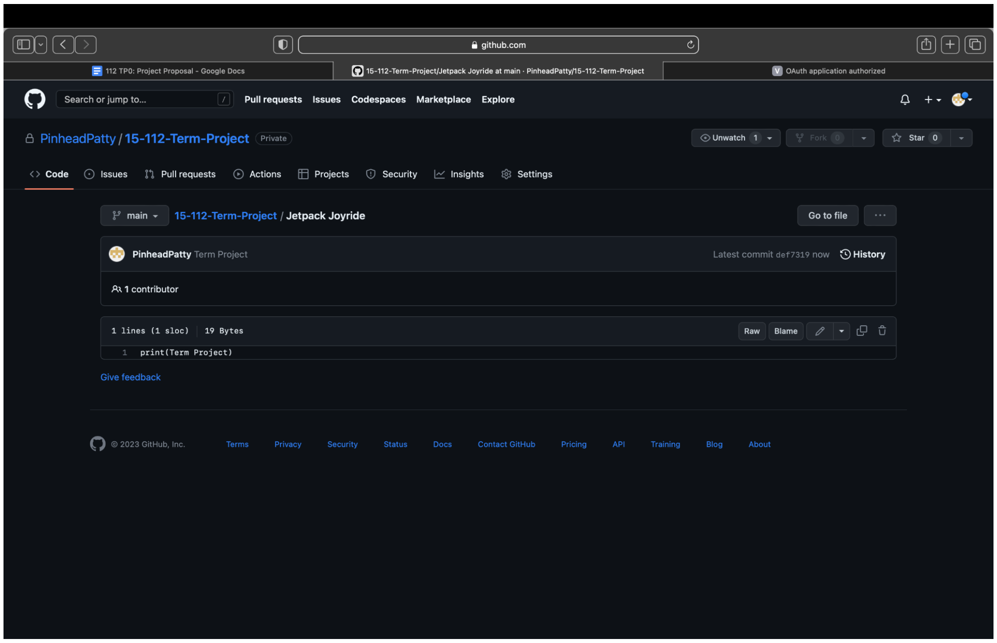
Version Control Plan



```
print(Term Project)
```

```
cd /Users/patlucas/Desktop/15-112/15-112-Term-Project
/usr/local/bin/python3 -i "/Users/patlucas/Desktop%
/15-112/15-112-Term-Project/Jetpack Joyride"
patlucas@Patricks-MacBook-Pro-9 15-112-Term-Project % cd /Users/patlucas/Desktop/15-112/15-112-Term-Project
patlucas@Patricks-MacBook-Pro-9 15-112-Term-Project % /usr/local/bin/python3 -i "/Users/patlucas/Desktop/15-112
/15-112-Term-Project/Jetpack Joyride"
  File "/Users/patlucas/Desktop/15-112/15-112-Term-Project/Jetpack Joyride", line 1
    print(Term Project)
          ^^^^^^^^^^^^

SyntaxError: invalid syntax. Perhaps you forgot a comma?
>>>
```



PinheadPatty Term Project Latest commit def7319 now

1 contributor

1 lines (1 sloc) | 19 Bytes

```
print(Term Project)
```

Anytime that I make a major change:

- I can run the code
- Save the file
- Type in the message bar the change that I made
- Click connect and then sync to back up my file in github

Module List:

N/A

TP1 UPDATE:

- **Classes for zappers and missiles done**
- **Zappers move onStep and character moving onKeyPress**
- **Character bounded by ceiling and floor**
- **It should be noted that everything I made is sort of just objects representing character and object hitboxes. The real game images will have to be assigned over top of the basic game.**

Missing: functions for coins and drawing packs of coins, lasers

Other Additions:

- A fairly smooth working gravity effect. It looks pretty good
- Collision detection for both zappers and missiles
- Distance Tracker
- Implemented slow and fast missiles, where slow missiles track your character for half of the screen

Overall I think I did a fairly decent job with the time that I had. However, I didn't get done quite as much as I wanted. I'm going to have to try to get most of, if not all of the game elements done by TP2 so that I can spend a lot of time doing user interface stuff.

TP2 UPDATE:

- **Death deceleration**
- **Basic menu outline**
- **Gadget button class**
- **code is more organized**
- **coinLists done**
- **Increasing speed over time**
- **A lot of the inGame graphics are cleaner**
- **Menu setup is getting there**

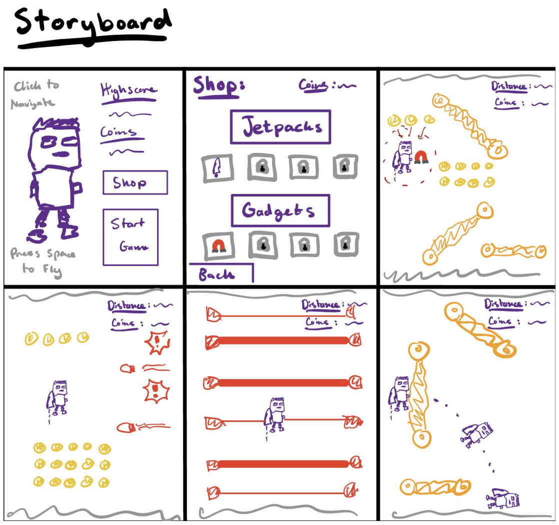
I need to work on getting my graphics and user interface looking nice. I also need a more complex obstacle algorithm and better timing with the obstacles in general, especially the lasers.

TP3 UPDATE:

- **All graphics and images incorporated as well as menu images**
- **gadgetButton toggles working with gadgets and jetpacks**
- **Rectangular collisions with zappers (based off a square hitbox)**
- **Increasing spawning with time (#spawned based on app.stepsPerSecond)**
- **Gravity belt gadget**
- **Pathfinder gadget(which uses backtracking) to try to find a survivable path for the character**
- **Added a click to start for initial game**
- **Fixed bugginess in acceleration/app.grounded**
- **Added music**
- **Added menu profile and corresponding graphics in game**

I put in everything that I was looking for in TP3. The game runs pretty smoothly and the graphics look good. I feel really happy with the way the game turned out. I hope you like it!

Storyboard:


Storyboard
Click to Navigate
Highscore
Coins
Shop
Start Game
Press Space to Fly
Shop:
Coins :
Jetpacks
Gadgets
Back
Distance :
Coins :
Distance :
Coins :
Distance :
Coins :
Distance :
Coins :