Elephant Survival Game

Project Description

The term project is tentatively[1] called "Elephant Survival Game." The project is designed to game-ify elephants by having the player act in the role of an elephant in a survival-style game. Following a scroll-through intro where the user can learn a bit about elephants, they start as a baby elephant and grow to become an adult. Their goal is to rise through the ranks of the elephant herd to finally become matriarch of the herd.[2]

There are three general life stages for the player elephant that correspond roughly to easy, medium, and hard levels of game play: baby, adult, and elder life stages. In all stages, the player is trying to make sure that their elephant gets enough water, enough food, and enough sleep. As a baby, the player's main goal is to keep near the mother and to build friendships with the other elephants. As an adult, the player's main goal is to continue to build friendships, stay close to the herd, and to deal with deaths within the herd. When they reach an elder, the player becomes matriarch of the herd. This means that the herd defers to the player. Their goal now is to make decisions for the survival of the whole herd.

Similar Projects

This project is supposed to be somewhat similar to games like *Minecraft*, *The Sims*, and *Tunnel Town*. Like Survival Mode in the video game *Minecraft*, the player is aiming to survive in their surroundings. They must meet basic needs like finding water, getting food, and sleeping. Similar to the interactive component of life-simulation game *The Sims*, the player is supposed to interact with other members of the herd and make friendships and make decisions that impact the herd as a whole. Like *Tunnel Town*, an animated game where you build a burrow of rabbits, this project turns elephants into a game. It also will, like *Tunnel Town*, have levels.

The difference between these similar projects and my project is that my project is a combination of the three. It will have survival elements like *Minecraft*, social elements like *The Sims*, and the modification of a real animal like *Tunnel Town*.

---

[1] The hope is to come up with something slightly more catchy than this before submission.
[2] This is not entirely reflective of true elephant social hierarchy, but for the purposes of the game, I have made some creative choices.

Structural Plan

The final project will be broken largely into three modes: opening screen, game mode, and help mode.

Opening Screen Mode

This will be a fairly simple single file broken into functions that load the start screen and a few screens with information about elephants and information on how to play the game. There will be functions that implement a skip-tutorial button that jumps straight to game-mode, a button that goes to a tutorial, and a button that goes to a page with information about elephants.

Help Mode

This will be another fairly simple single file broken into functions that load the help screen when a key-board input is pressed in the Game Mode.

Game Mode

This will be broken into roughly five files:

1. Survival Classes/Implementation
    a. This will include the classes for watering holes (the source of water), grasses and tree clusters (food sources), and a day/night implementation (implementation of sleep).
    b. This will also include functions that dictate the player interaction with each of these classes.
2. Baby Elephant Implementation
    a. This will include functions that dictate the player keeping near the designated mother elephant and social interactions with other baby elephants within the herd.
3. Adult Elephant Implementation
    a. This will include functions that dictate the player continuing to build friendships with other elephants in the herd, staying close to the herd, and the possibility of elephants dying within the herd.
4. Elder Elephant Implementation
    a. This will include functions that dictate the player becoming the matriarch of the herd and thus having to keep the whole herd safe.
5. Player Elephant Implementation
    a. This will include the functions that dictate the player's actions in general, including loading the sprite representing it and the keyboard actions that will control it. It will import functions from files 2-4 to help implement this.

Main.py

This will be the main file that loads the app and imports all the various files that dictate gameplay. It will include appStarted, keyPressed, redrawAll, timerFired, and other functions as necessary.

Test.py

This will include all of the test cases for the project.

Algorithmic Plan

There are two difficult parts of the project to implement will be:

1. Elephant Interaction with the herd:
    a. The herd will be created using a general elephant class that has somewhat randomized elements to add a level of complexity to the game.
    b. Social interaction between the player and the herd will be done on an elephant-by-elephant basis where certain relationship connections (i.e. making friends) will be determined by user interaction.
        i. The basic interaction to make a "friend elephant" will be to move to the elephant and share food.
2. Survival elements:
    a. The watering holes (i.e. the source of water) and grasses and tree clusters (i.e. food sources) will be built using separate general classes that include a somewhat randomized quantity that elephants in the herd will take from. They will be somewhat randomly generated within the world of the game and the player will be able to travel certain distances depending on how thirsty and hungry they are to reach them.
    b. The day/night implementation (i.e. implementation of sleep) will be created using a function called in timerFired that will happen at even intervals. The player will be able to sleep during this time frame to regain energy. They will not be able to sleep during "day" periods.

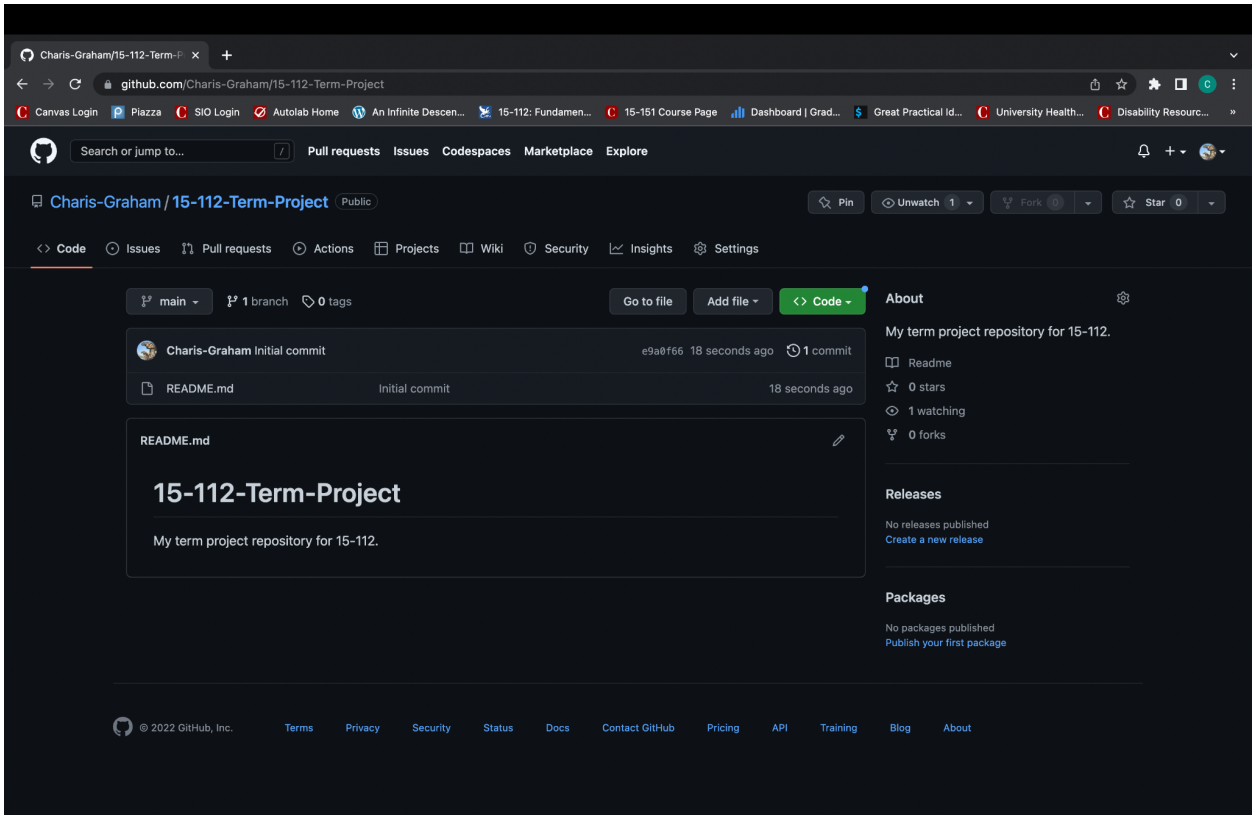
Timeline Plan

- Wednesday, November 16
    - Have a basic framework for the project with a starting page, a help page, and a main game play stage.
- Friday, November 18
    - Implement the elephant icon and have it respond to keyboard inputs as well as change with certain time frames.
- Saturday, November 19

    - Create the classes for watering holes and food resources
- Sunday, November 20 @ 5pm: TP1
    - Have a finalized plan for the project with 400-600 lines of working code.
- Tuesday, November 22
    - Implement combined gameplay aspects of elephants needing to survive by getting to watering holes and food resources and sleeping all working within limited randomized parameters.
- Friday, November 27
    - Implement elephant social aspects so that the player can interact with the elephant herd.
- Wednesday, November 30 @ 5pm: TP2
    - Have a Minimum Viable Project (MVP) with the majority of the project's main code finished.
- Friday, 2 December
    - Add background music to the game
- Sunday, 4 December
    - Create the video demo, make the README file, and finalize the design files.
- Wednesday, December 7 @ 5pm: TP3
    - Final project due, including a README.txt file, a video demo, and all design files related to the project.

Version Control Plan

I will be backing up my code on GitHub in a repository called 15-112-Term-Project.



Module List

I am planning on using no outside additional modules, hardware, or technologies, at least before MVP.

TP1 Update

I have changed how I am organizing my files a bit. I have moved the watering hole, elephant, and tree classes into one file called classes.py. I have also made another new file with commands by the player for movement called elephantPlayer.py. Finally, I have added another new file called graphics.py where I have created the background for the game play.

TP2 Update

I have moved functions into specific files for the start screen, help screen, and game modes. I have also implemented game logic in the game mode file. I have also removed test.py as I have not seemed to need it so far.