Donny Le

TP1 Deliverable

Project Plan

The name of the term project will be called “Grappling Garry Plus”. It will be based on the mobile game, Grappling Garry, by the game company, Double Jump. The goal of the game is to climb up into the sky by grappling onto drones whilst avoiding bombs. The player loses when they touch a bomb but gains points by touching drones. When a drone is touched, the drone falls and the player cannot grapple or interact with it. However, if the player goes within a short distance of a drone, they will receive something called a “Near Miss”. If the player receives a “Near Miss”, their score multiplier increases by one. The game is a 2D vertical scroller and does not have levels so the player's main goal will be beating their previous scores. In addition, the base game features different drones the player can grapple onto that will give them special power-ups. Most of the term project will share the same core mechanics as Grappling Gary but with certain aspects of the game modified/improved.

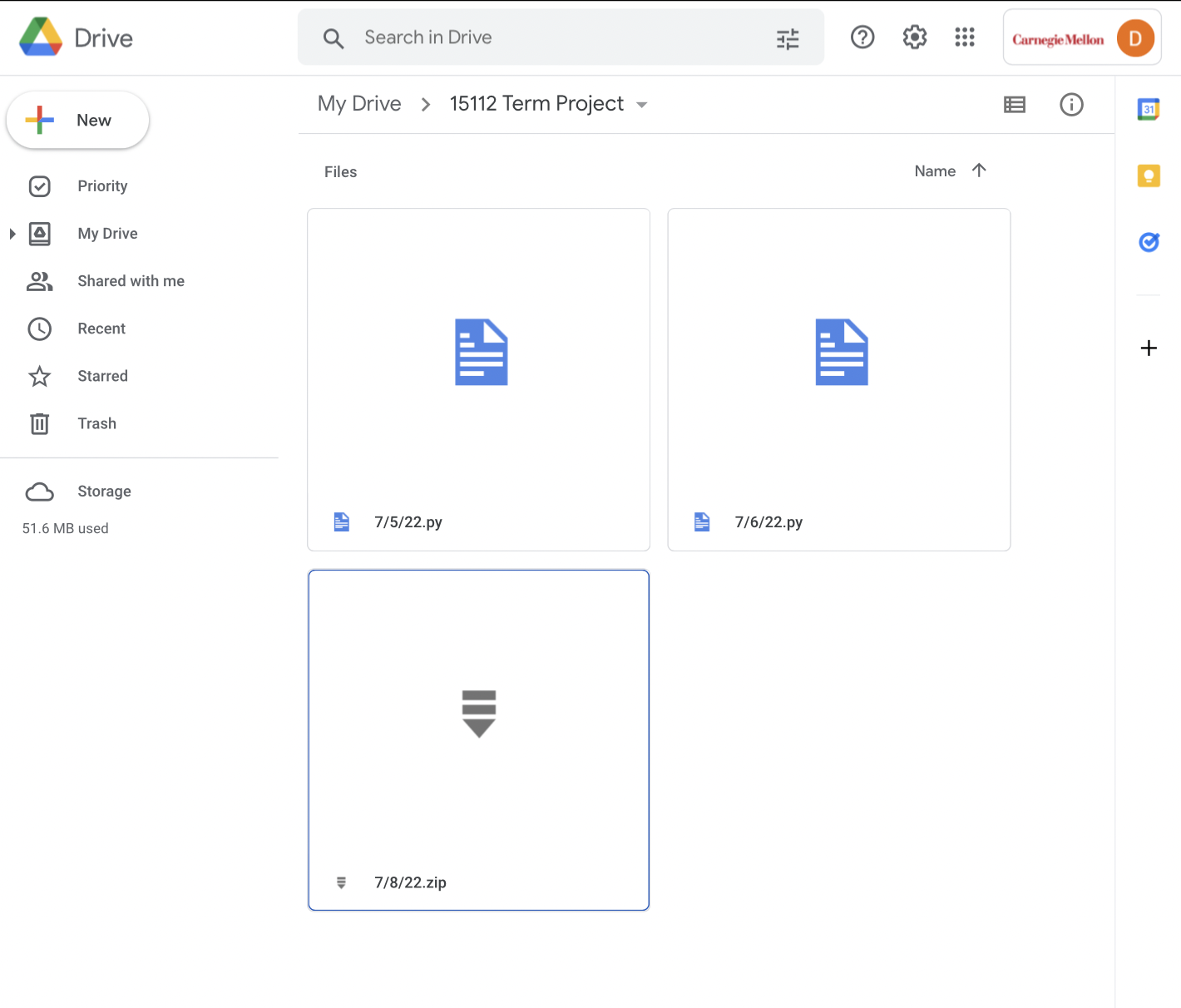
The code of Grappling Gary will consist of (at least) three different files. One file (named as “objects” currently) will include all the objects in the game. The parent class, gameObjects, will have the basic traits of all the objects in the game. The functions in this parent class include but are not limited to: moving the object, checking if the objects intersect, and a function to draw the hitbox of the object. It also stores the size and position of the objects in the game. There will be at least 3 subclasses of this parent class: a Bomb subclass, a Character subclass, and a Drone subclass. To make declaring objects easier, the hitboxes of each object will be declared inside their respective \_\_init\_\_ functions. The Drone class would also have more subclasses for the different power-ups. The second file will be called the main which includes most of the code for the game rules and features. It will utilize top-down design and helper functions. The code will be organized with all the draw helper functions at the end, the helper functions and most of the logic in the middle, and the user interaction part in the beginning. Once different menus/difficulties are created, they will be further organized into different modes. The third file (may be combined into main) would include the layer generating algorithm which will be explained next.

The most complex part of the project will be the algorithm to generate new layers (a row of bombs/columns). It will use different probabilities to generate layers at different levels of difficulty. As the player moves higher in points, their chances of receiving a harder layer would increase. However, the algorithm should also give a player a mini-break at different times during the game by granting them easy stages to “reward” them for completing harder layers. In addition, all the layouts of the bombs and drones will not be hard coded and generated randomly without making the layer impossible to move past.

Another area of complexity is the movement of the character throughout the grappling motion of the app. When the player clicks on a drone, the player is given immediate velocity and moves directly to the drone. A grapple hook image is drawn from the player to the drone. This was done by calculating the slope and finding the angle between the drone and the player. The angle was also used to find the x and y velocity vectors of the player. However, when calculating the slope, the direction of the player must also be taken into account since a positive slope can either represent going into the positive direction up and down or the negative direction up and down. When the character hits a drone, it slows down to a certain velocity and then gravity acts on it. The vertical scrolling aspect also has to be implemented with the player’s movement and only scrolls with the player when the character is above a certain threshold.

The code will be backed up every day/session using Google Drive. Here is an image of the folder:

1



TP2 Updates:

* The plan to have three different files is still active.
* The layer generating has been completed but further optimization may still be needed.
* For further features, I plan to implement different powerups which include a missile tracking drone that locks onto bombs and flies in a sinusodial manner.
* The highscore system needs to be further optimized since it stores every score. If I decide not to change it, I could have an all time stats section
* I still need to implement the textures and make everything look visually appealing
* There may not be enough time but I would like to try to implement a path-finding algorithm (or something similar)
* The button object is also subject to change
* The number controls may also need tweaking to be usable (works but it is not very effective).