Game Journal

Pre Project Work

Before the project was actually assigned I had already started work on my game, I got as far as a making a primitive version of my current game with the help of online tutorials. This version only featured the single green ship and the first level enemies, the music and sound effects, and the start and game over screens. The hardest part of it was getting the explosion class to work as I did not understand the use of the time class method get\_ticks () but I not now it is used to make the explosions animate at a consistent frame rate and eventually managed to get them working. This version was also written without a game class only the classes for the player, enemies, and explosions.

April 29

Now knowing more about the requirements of the project I spent about an hour rewriting original version of my game using a game class. I though this would be difficult at first but using the example of the snake game from in class I mange to get it do easily.

April 30

I spent my spare and some time after school to start working on the first level boss. My plan for it behavior was for the boss to move back and fourth at the top of the screen and shoot three projectiles at the player that would travel in a sort of cone shape from the boss at slightly varying angles, I managed to get as far as creating the bosses class and getting it to spawn and move but the shooting at angles was still something I wasn’t sure of how to do so for the time being it just spawns and starts moving back and forth when the player reaches one thousand points.

May 1

Spent about half an hour creating a new class for the boss’s shots and coming up with a way for it to shot at an angle. The solution I came up with was to create two variables for the shot’s speed one for the x and y coordinates each, to get the angles to vary for each shot but keep them traveling relatively towards the player I keep the y component of the speed constant at five and made the value of the x component random in a range of negative five to five, this is the solution I am currently going with and so far I am happy with it.

May 2

Used my spare again to start working on my first power-up the gatling gun. This turned out to be pretty easy as all I had to do was create a class that would randomly spawn the the power-up icon itself anywhere on the screen and make it so that when it and the player collided the delay on firing that I had set on the player would decrease. I spent the rest of the spare altering the player class to allow the player to move all over the screen and not just left and right at the bottom

May 3

Having decided to include multiped levels I spent the python work period on creating a secondary python file that would contain the code for level 1 which I would import into the main game file. This was a bit frustrating seeing as I now and to rewrite most of the code in main game file pertaining to interactions between the enemies and boss so that the two files would interact appropriately but after a lot of debugging, I managed to get it work.

May 5

Spent two hours writing the code of the second level, I got as far as making the enemies and boss spawn but the boss so far seems pretty easy , it’s not connected to level one at all, and I haven’t found new images for the enemies and boss yet and I am just the ones from level one for now.

May 6

Spent my spare creating a function for the player to select a ship when they start the game by creating a list of sprites and letting the player cycle through them. So far it works but I again still haven’t gone looking for new images yet and am just using the default green ship and the enemy ship sprite.

May 7

Spent the period creating the high score system which uses three functions one t display the high scores and names when the game ends and one that takes the player’s score from when the game ended and places it among the top fire scores in the score file if the player got a high enough scores, and one that lets the user enter their name when the game ends by creating a string and concatenating every letter the user enters to it so that it will be displayed along side their score if they make it into the top sores.

May 9

During the period and lunch I worked on making the two levels connect so that when level one ends level two starts up. Originally I had it so that both where instantiated in when the game class was instantiated and then setting the boolian variable that control the second level to false, however that resulted in a single wave of the level two enemies appearing at the start of level one so instead I created a variable that keeps track of what level the player is currently one and the game class only checks the status of that level.

May 10

Used the period to create a function that would display the current level the player was on at the start of each level and freeze the fame for a few second using the wait() time class method. I also tweaked the health and spawn rate of the enemies in level 2 to make it more balanced I made the spawn rate one second as opposed to two and lowered their health from six to four.

May 13

Used the period to finished tweaking level two. The boss and enemies are finally at a point that I think they are fair but still relatively challenging. I also created the outline for level three giving it the structures for the boss and enemy classes as well as its own level class.

May 14

Used the period to start working on the second power-up the spread shot which I expected to be the most challenging of the three. This first version simply creates eight instances of the shot class and spends them each in different directions by creating variables for the x and y components of the shot’s speed like with the boss in level one but for all eight shots. This version works but doesn’t feel efficient and lags the game when fired.

May 15

Used the period to try a new method of creating the spread shot which was to add a nested list containing the x and y components of each shot, this works and feels more efficient than just having sixteen variables but it did still lag the game. It was brought to my attention that the reason for the lag was likely the fact that I was resizing the image for the shot each time so to fix this I resized the source image itself in Photoshop so I wouldn’t have to resize it within the program.

May 16

Used this period to create actual images for the menus in Photoshop instead of drawing the text and rectangles on to the screen within the code. I also finally created an instruction screen that can be accessed at the beginning of the game that tells the player about the controls.

May 17

Used this period to start actual work on the third level starting with the normal enemies. These enemies follow the player at shoot towards them, because of this unlike the enemies from the other two levels these enemies require the player class to be passed in meaning that the first enemy created when the level is initialized needs the player class to be in the init function. This means that the current way I have been handling level switching will have to change as the variable s being passed into the run logic function are not the same across all levels so I’ll either have to come up with a new method or add some if else statements.

May19

Spent about an hour writing the code for the boss of level three. The challenge for this boss was getting it to shoot towards the player while also moving around the entire screen. Originally I had it so that the x and y speeds of the boss’s shots where determined by the difference in the location of the player and boss, this resulted in the shots either moving too fast or slow depending on the distance. The solution I currently have is that the speeds are within a random range that randomly generates the x and y speeds of the shoots, to keep the boss shooting at the player I made it so the range of the speed changes based the boss and player location in relation to each other.

May 21

Spent the period rewriting the code for switching levels. Due to the third level requiring different things to be passed into both its init and run logic functions I can no longer have just a single line for the running the logic of each level like I had before, instead I now have if else statements that check whether each level has ended and if not runs the logic for that level.

May 22

Spent the period transferring the explosions class, the explosion function, and the load image function to a new file since these elements are used in each level file and until low I simply had a copy of the functions and class in each file. This was annoying as I had the comb through each file looking for any instance in which I used any of these function or class.

May 23

Spent the period writing the code for my third power up the charged shot which goes through enemies. To do this I had to add a new variable to each power-up which determines whether it does goes through enemies or not , I also wanted to change the sprite used for the shot for this power up so I had to add an image variable the original shot class, and in the process of all this I decided it make the power-ups each do different amounts of damage for the sake of balancing them so I added a damage variable to each of them. Because of all these changes I now have to pass two new variables into the run logic function of each level telling them whether the player’s shots go through enemies’ and how much damages they do.

May 24

Spent the period trying to figure a way to fix the problem with the third level boss, so far each attempt to get the boss to shoot at specific angles has not worked so I believe I have no other choice but to do some research and learn how to do vectors in python☹.

May 27

Spent the period trying to figure out how to do vectors in python and get the final boss to shoot at the player. I started by just using the x and y speeds and trying to avoid angles like it managed to do with the first level boss and third level enemies but I eventually realized this was not going to work and I was going to have to learn how to work with angles in python.

May 28

Spent the period and lunch researching and working with angles. I had to learn how to convert radians to degrees and work around how python does angles in general with it only going to 180 degrees and -180 degrees and not the full 360. Eventually I got it working and the boss now shoots at the player.

May 29

I decided that with the final boss now working with angles I should try and implement one of my hope that was related and try and make it so the player could aim shots with the mouse. Making the player rotate with the mouse was a little difficult as the player image kept changing size as it rotated but after resizing the player itself in Photoshop is worked.

May 30

Today I spent the t period working on making the players shots go at the angle the player was facing this was not too difficult as all I had to do was copy the code I use for the third level boss and integrate it into the shot class. The hard part was making the spread shot power up work with these new controls. I settled on making an if statement and for loop completely dedicated to this and adding a list of angles to the spread shots class that would each be passed in to the shoot class and removed the shot speeds I had in each power-up beforehand.

May 31

Spent the period commenting on my game.

June 3

Spent the period finishing commenting on my game.

June 4

Spent period looking over my game and correcting mistakes in coding style as well as working on balancing the enemies and boss.

June 5

Spent the period still working and balance and editing sprite having finally found sprite to use for the third level enemies and boss, and the charge shot.

June 6

Spent the period having people test my game.

June 9

Spent an hour doing more testing and polishing and adding the win screen

June 10

Spent the period doing final touch up and handed it in, scared as all heck.