|  |  |
| --- | --- |
|  |  |
| **Computer Science Course Work** |  |
|  |  |
|  | 2022Dino Game |
|  | Leo GratwickSurbiton High School |

**Contents:**

Analysis…………………………………………………………………………………………………………………………………………… 3

Objectives…………………………………………………………………………………………………………………………… 5

Documented design………………………………………………………………………………………………………………………… 7

Algorithms……………………………………………………………………………………………………………………………7

Classes………………………………………………………………………………………………………………………………..11

Screen Design……………………………………………………………………………………………………………………..15

Level Creation……………………………………………………………………………………………………………………..18

Hierarchy chart……………………………………………………………………………………………………………………19

Technical Solution……………………………………………………………………………………………………………………………20

Dino class………………………………………………………………………………………………………………………………..

Background and Screen and level……………………………………………………………………………………………

Other Classes (crystal, chest, heart, health bar, button?)………………………………………………………..

Main game loop………………………………………………………………………………………………………………………

References……………………………………………………………………………………………………………………………..

Testing………………………………………………………………………………………………………………………………………………..

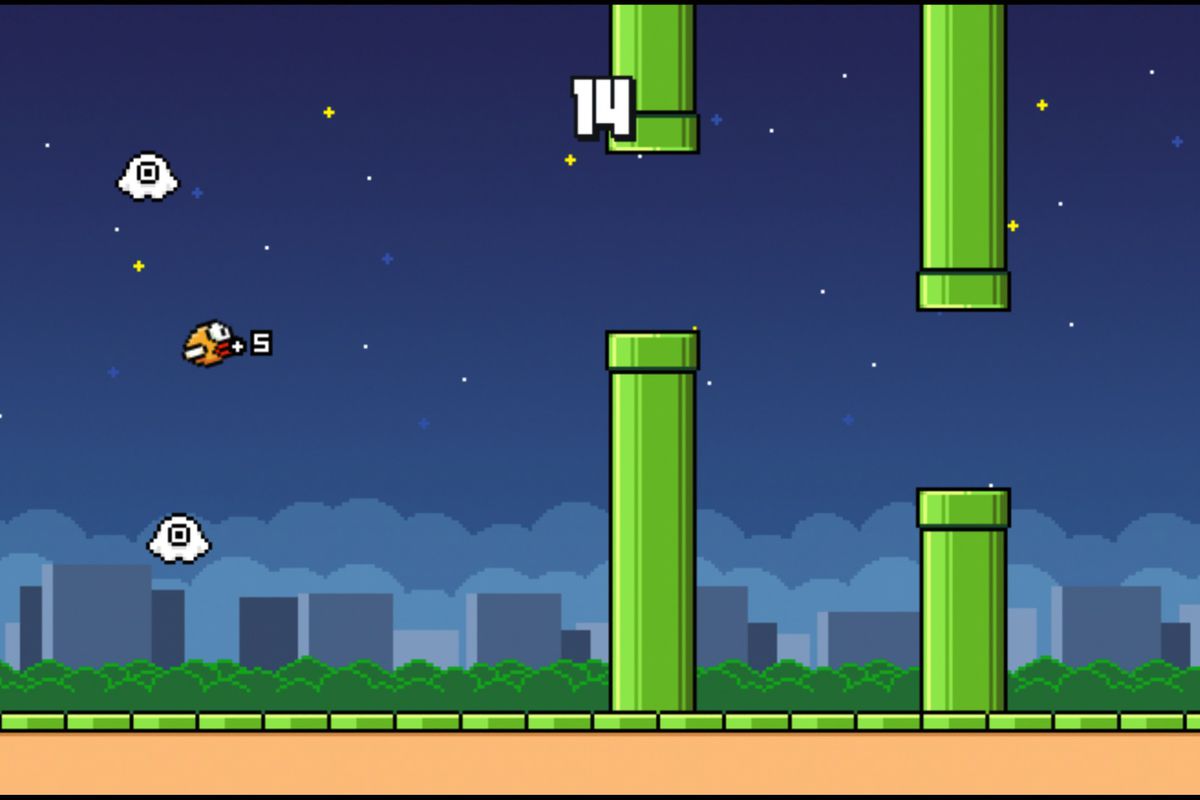
Evaluation……………………………………………………………………………………………………………………………………………

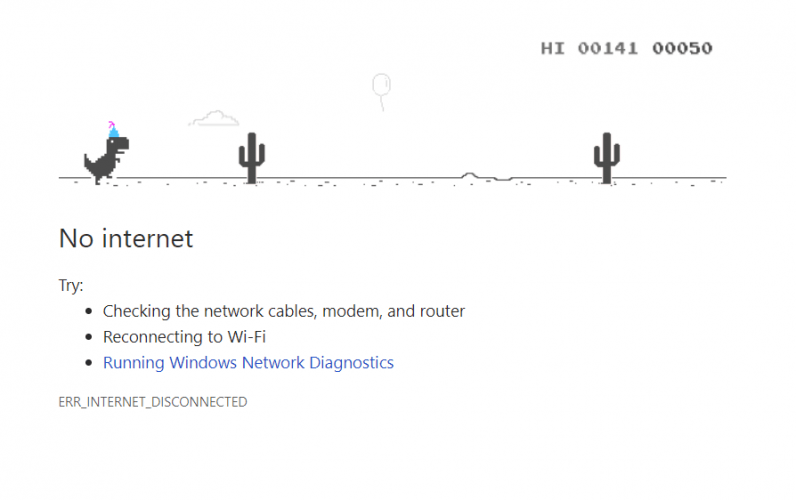
**Analysis**

I enjoy playing games, and I am interested in the programming behind them, so I thought that I would use this opportunity create my own game, that I could play with my friends.

One of my favourite games is Minecraft, which is a sandbox game, based of everything being made of blocks. You can play single player or multi player with your friends in which you can play in survival, where you must collect resources and fight monsters to survive, or creative where you can’t die, and you have access every single block in the game and is commonly used for building things. There are also servers you can join which have mini games and adventure maps and many other things. You can basically do anything you like. Games like this take years to develop and are still releasing updates now. So as interesting it would be to program a game like Minecraft, it would be very unrealistic.

Initially, I wanted to do a game similar to The Dinosaur Game you play when the internet goes off, or Flappy Bird, where you go through randomly generated terrain and try and survive for as long as you can. However this is quite simple and after a while it gets repetitive and boring. So then I moved on to something slightly more complex, Fire boy and Water girl and Mario. I played Fire boy and Water girl quite a bit when I was younger, and I really like the design of the maps and the way that there are different levels and you can see the whole map. Also in Mario I like how you can use tunnels to get to different places and to get to the rooms with the enemies that you can beat, I also like the maps in Mario. I will also take inspiration from some of the features of Minecraft, such as crafting.







I decided that the target audience for my game would be people around my age, as I want to make a game that me and my friends would enjoy playing. Later on I will ask some of my friends about some of the things that they like and dislike about the games that I am taking inspiration from.

In basic Minecraft, you spawn in a new randomly generated world, and you need to break some of the blocks from a tree, turn them into planks and then create a crafting table, which you can use to make tools such as pickaxes, axes, and many other things when you have the resources. I really like the idea of being able to craft objects, with things that you have collected so I will try and incorporate that into my game, although it won’t nearly be a complicated as Minecraft crafting, it will probably be something like combining an item with a weapon to make it stronger. Another thing I really like about Minecraft is the design of the health bar and the inventory at the bottom of the screen, so I might use something similar in the graphics in my game. One other thing I like about Minecraft is that you can find chests with useful items in them, I want to incorporate a feature like this into my game, where if you go into the right room you can find chests with items that you can use to upgrade weapons or armour using crafting.

My game will be a platform game with around 5 levels, each having 3-5 enemies of different levels. All enemies will drop items which you can use to upgrade your attack or defence statistics. Each level will also have hidden chests with items that you can use to upgrade your attack or defence levels.





**Objectives:**

Player

The player can go forwards backwards and jump using A, D, Space and sprint with S 🗹

Player can deal damage to enemies using right click, however there should be a cooldown after using an attack 🗹

There should be walking, jumping and attacking animations for the player’s movement 🗹

If the player is killed, they should respawn at the start of the level

The player has a set amount of health depending on what armour they have 🗹

At the start of each level the player should spawn with the lowest grade attack and defence

The players health, attack and defence levels and number of gems should be displayed on the screen 🗹

Enemies

All enemies should be able to fight and deal damage to the player

All enemies should have a set amount of health and be able to be killed by the player 🗹

All enemies should drop item(s) after being killed 🗹

Enemies should not respawn after being defeated, unless the player dies 🗹

Crafting

The player should be able to craft, to upgrade attack and defence by using 1 2 3 4 5 6, by using drops from defeating enemies 🗹 or items found in chests

The player should only be able to increase a stat if they have the correct crystal to do so 🗹

Attack and Defence

Attack and defence have levels 1-10 🗹

The level of the attack determines how much damage it does to enemies and the level of defence determines how much damage the payer will take when hit by an enemy 🗹

Chests and Drops

Chests will spawn in set locations of the map in each level🗹

Each chest will contain 1-2 random crystals or hearts🗹

Each enemy will drop a set item, the type depending on the health of the enemy 🗹

There are 4 different items: green, red and blue magic crystals and hearts 🗹

Green magic crystals will increase the level of items by 1 level, red by 2 levels and blue by 3 levels🗹

Hearts restore the players health 🗹

Levels

There should be 5+ levels

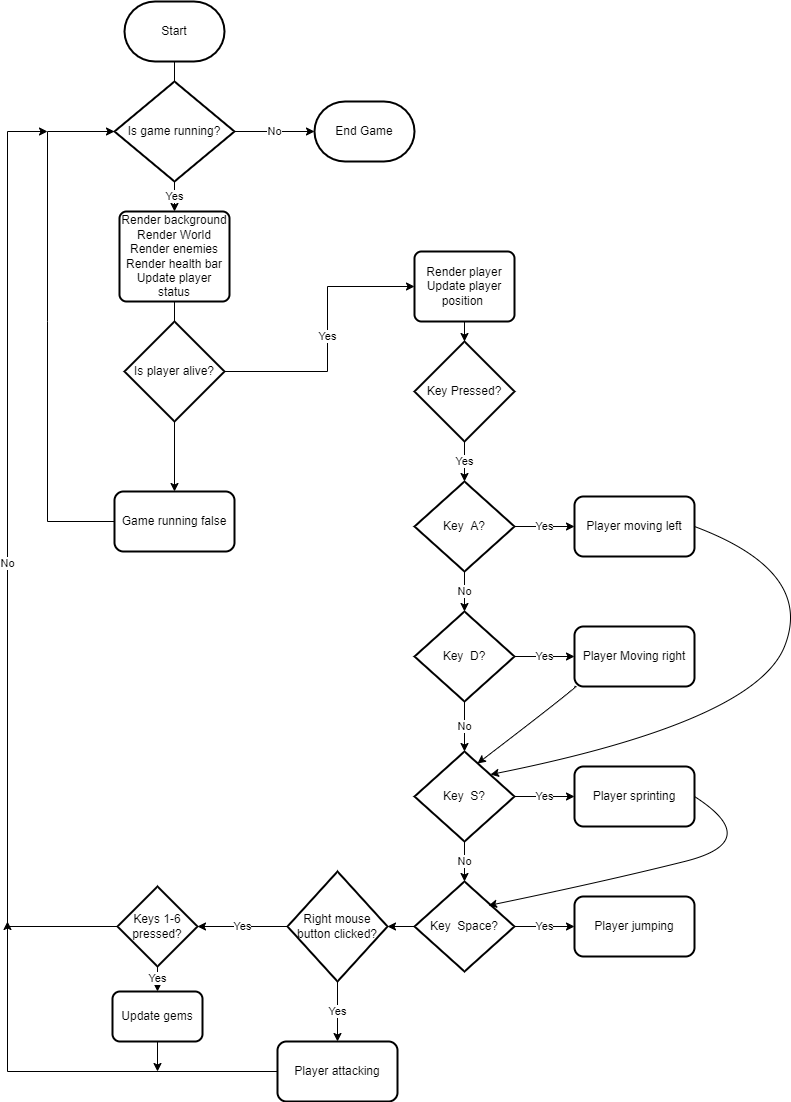
Each level should have a different map

Each level should have a different boss

Each level should be progressively harder to clomplete

**Documented design**

Algorithms

For my project to work I will need to use a wide variety of algorithms, from a main game loop, to player movement algorithms

This is a flowchart I made for the main loop of the game that runs while the game is open, it renders the background and the level, then it checks if the player is alive and based on the keys pressed it updates the player class, it also renders and updates the enemies, the gems and everything else in the game.

Diagram

Description automatically generatedThis flowchart shows a basic algorithm for player movement, where player moving left, right, jumping and sprinting are variables passed in from the main game loop where the player presses keys to control the player. The layer velocity is set beforehand and is how far the player moves per frame, whether it is negative or no depends on the direction of movement of the player. If the player is on the ground the y coordinate wont change so the player won’t fall through the ground, however if the player isn’t on the ground gravity need to be applied to the y coordinate, so the variable gravity should be a negative constant so that it acts downwards. When the player jumps the y coordinate is increased by a constant once in this case 12, but this could be changed depending on how high I want the player to jump. If the player is sprinting the change in the x coordinate doubles which doubles the speed of the player. Lasty the change in x (dx) and the change in y (dy) are added to the current x and y coordinates of the player, which is where the player will be rendered in the next game loop.

Diagram

Description automatically generatedThis method of the player class will be for handling collisions, of everything in the game with the player. Collisions with enemies are checked, if the player is attacking it does damage to the enemy depending on the attack level, however if the player is not attacking the enemy does damage to the player depending on the level of the enemy and the defence of the player

Diagram

Description automatically generated with medium confidenceThis algorithm will be a method of the chest class, it checks that the player is touching the chest and then the player needs to left click it to open the chest, when the chest is open, it generates a random amount (1-3) of random loot items and then finds coordinates that aren’t inside the ground to place them at.

Classes

My game will be based around object oriented programming and calling all the methods from the classes in the main loop to update and render the images of everything in the game including the player, the enemies, the chests, the level and all the other things that will be in the game. Most of the classes I will create will have images that will be saved in the same folder as the main programme, and also have coordinates to render them at. Classes such as chests and gems will be put into an array every time an instance of them is created which will make it easier to call the update and draw methods in the main loop as I can just use a loop an iterate and call the methods of all the objects. This will also make it easier to remove things such as crystals and hearts after they have been picked up ad I will be able to just remove them from the list and they will no longer be rendered or updated

Text

Description automatically generated with medium confidence

Shape

Description automatically generated with low confidence

Text

Description automatically generated

Text

Description automatically generated

A picture containing text

Description automatically generated

Text, shape

Description automatically generated with medium confidence

Screen designs

For my game I will need images for everything including the player, enemies, background ect. I went online to some free sprite websites[[1]](#footnote-2), and pixel art websites to find some sprites for my game.

I used an online sprite cutting website[[2]](#footnote-3) to split the long sprite sheets like the one below into separate images for the animations, also for some sprites I needed to remove the background[[3]](#footnote-4).







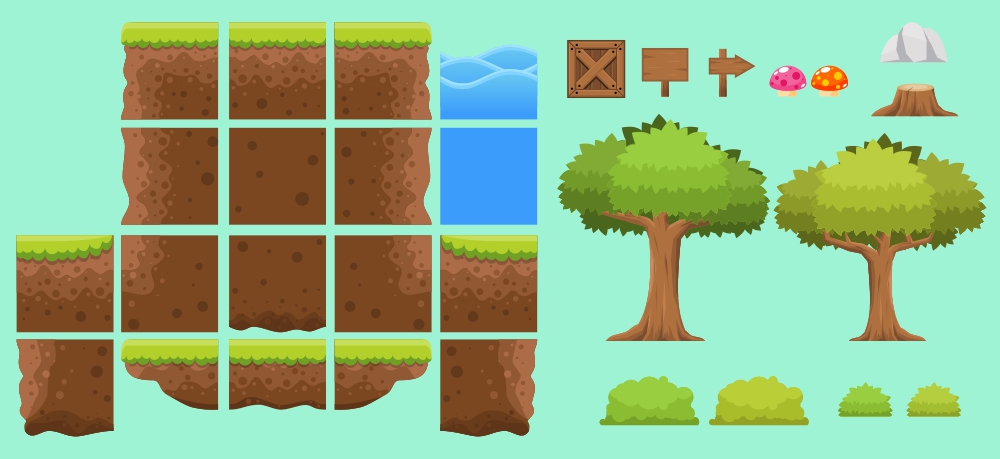


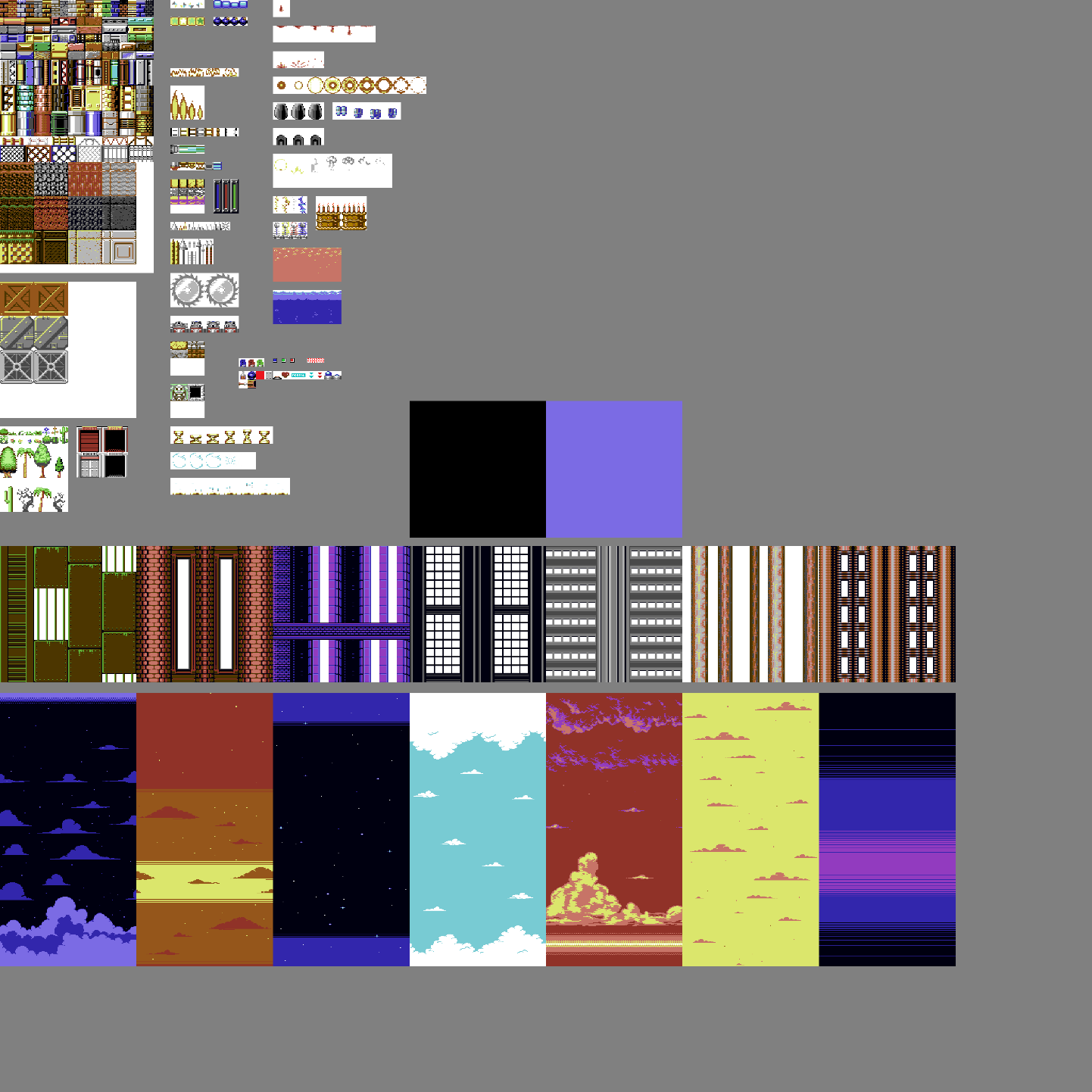


Above is an example of a sprite sheet compared to a single sprite that has been cut from the sprite sheet

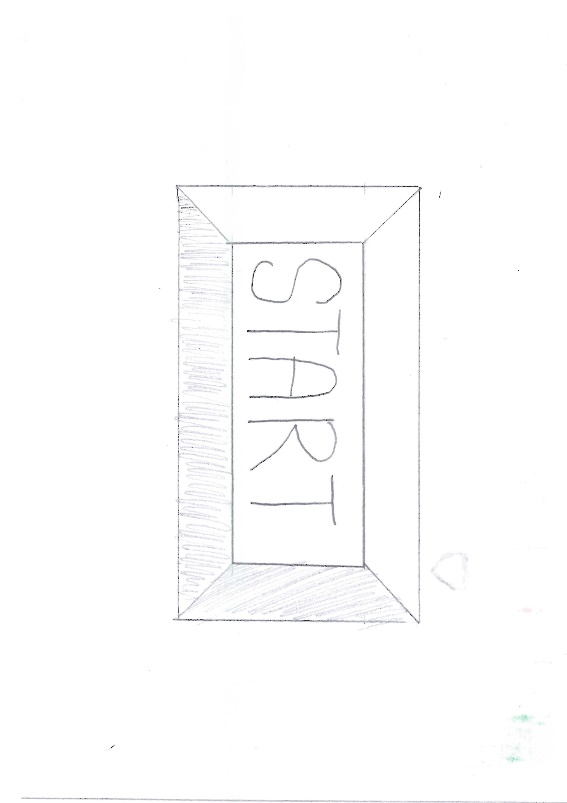
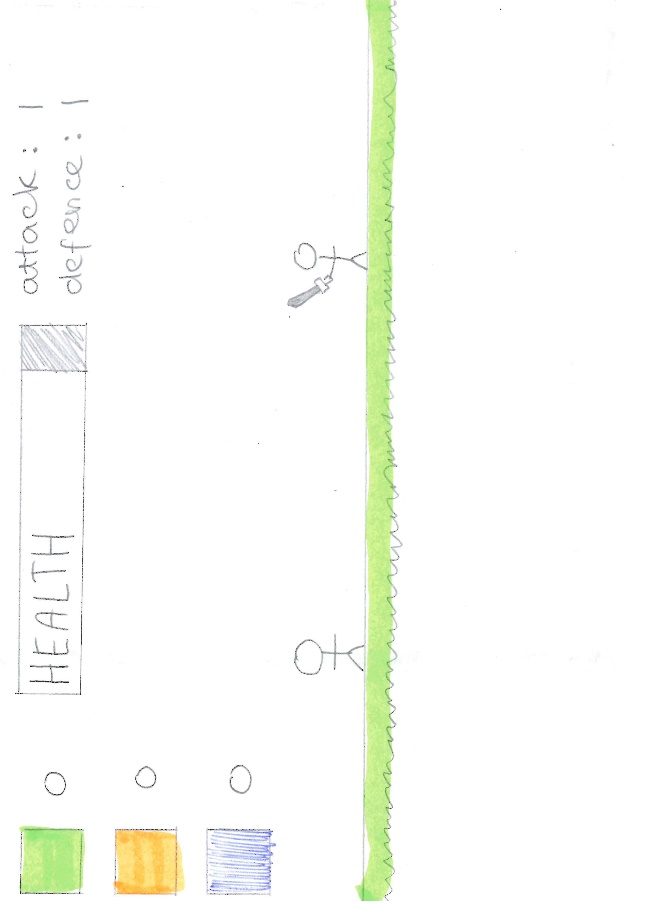
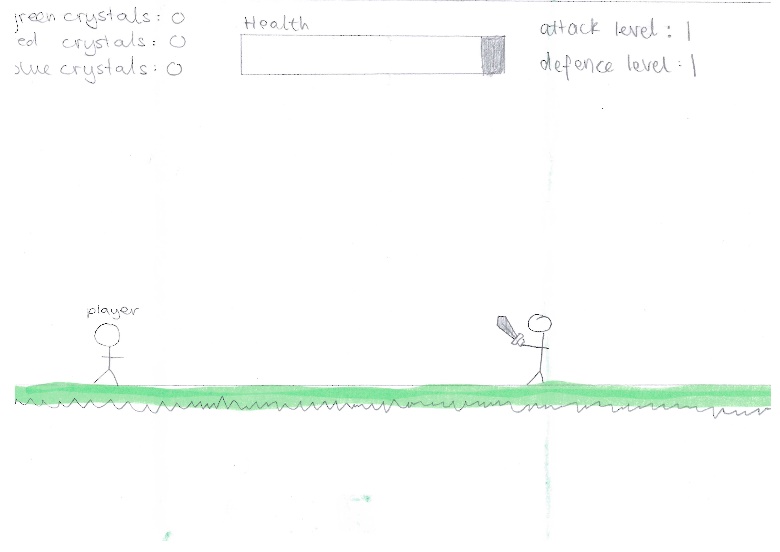
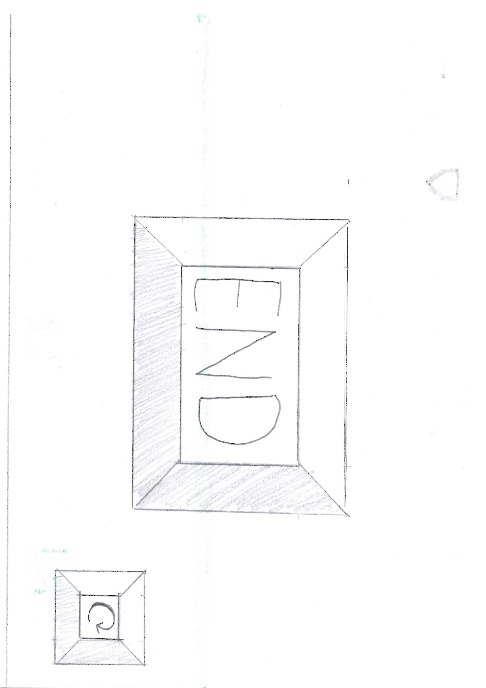
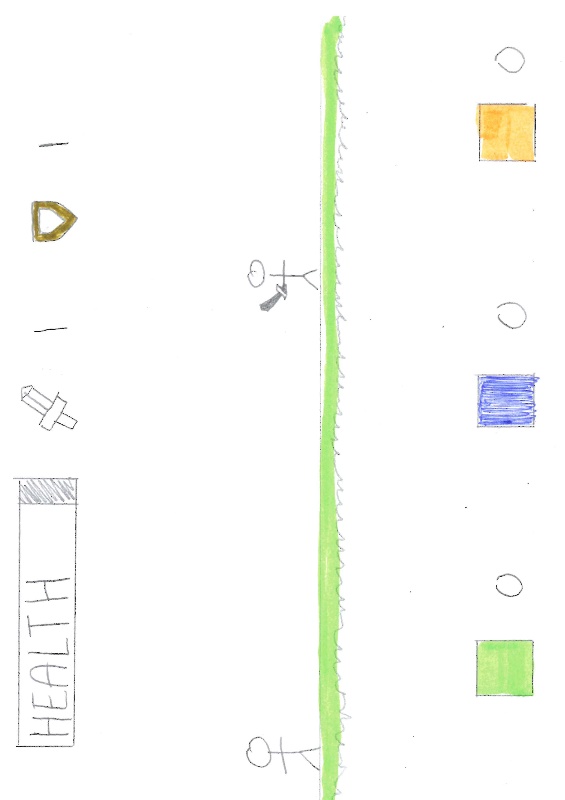
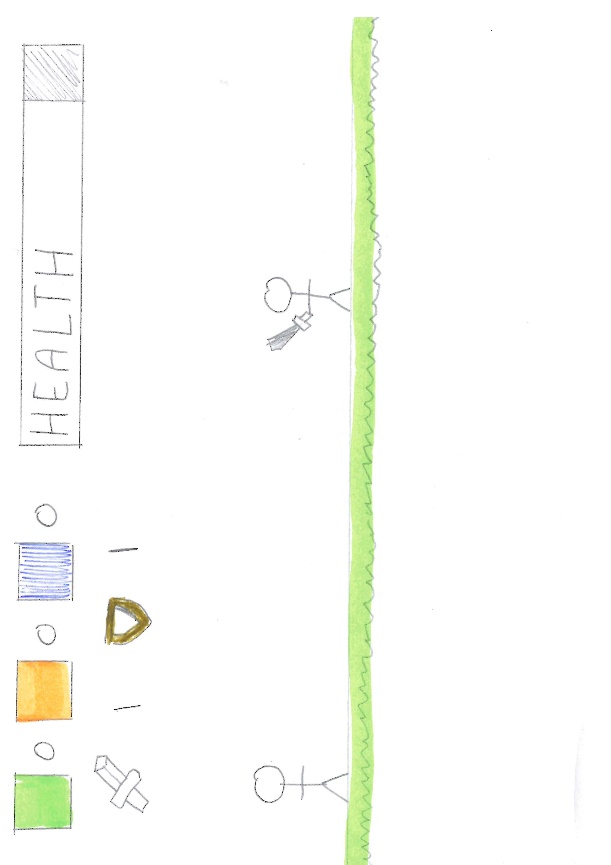
I also used a website[[4]](#footnote-5) that pixelated images for some of the more detailed sprites so they would have the same amount of detail as all the other sprites







This image I found has may different types of sprites, however I like the backgrounds at the bottom of the sprite sheet so I will crop and use those as the background for my game

After I was finished researching sprites, I thought about the screen design and drew a few designs of where everything on the screen would be including, the health bar, the number of each gem that the player currently has and the attack and defence levels of the player. I also screen designed the start, end and reset screens.

Diagram

Description automatically generatedHierarchy chart

Level Creation

I also thought about how I would create each level of my game and I decided that the best idea was to have the game be based on tiles as then I could store levels much more easily as a 2-dimensional array, where each array inside the main array represents a row of tiles in the game. Each item in each row is a number which will represent what tile goes in that position, the numbers will include representations for the different aspects of the ground, chests, enemies, and players. In the class world, there will be a subroutine that processes all the data in this 2-dimensional array and creates an instance of each class in the correct location on the screen depending on where in the array it is. This subroutine will also make an array of all the ground tiles and handle the rendering of them in its draw function.

For example: Where, sky = 0, grass = 1, earth = 2 and player = 3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 3 | 0 | 1 | 2 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

Level = [[0,0,0,0,0,0,0,0,0,0],

[0,0,0,0,0,0,0,0,0,0],

[0,0,0,0,0,0,1,0,0,0],

[0,0,0,3,0,1,2,1,1,1],

[1,1,1,1,1,2,2,2,2,2],

[2,2,2,2,2,2,2,2,2,2],

[2,2,2,2,2,2,2,2,2,2]]

I hope to have at least 4 levels by the time I finish programming the game, which become progressively harder the higher the number of the level, also the game should also automatically load the next level once the player completes the previous level.

**Technical Solution**

For my technical solution I installed a set of Python modules, called Pygame[[5]](#footnote-6), which are designed for creating video games in Python. These include things like computer graphics, so I can load in images and use them as sprites and also, I am able to render images and shapes in a window, and sound libraries, so I can add sound effects to the game, that are crucial to creating a video game.

Dino class:

Movement + player ground collisions

Player animation

Picking up crystals

Player enemy collisions

Enemy ai

Enemy drops

Background and Screen and level:

Rendering background, writing on screen

World sort data

Creating entity groups

Screen scroll

Loading images into an array

Other Classes (crystal, chest, heart, health bar, button):

Player gem collision

Chest generation

Chest player collision

Crystal player collisions

hearts

Health Bar

Button

Main game loop:

User Input from keyboard (key down / key up) and mouse clicks

Calling methods from classes

Updating player actions

Closing out of game

References:

To help me get familiar with the Pygame module that I was using I looked up a few tutorials and took inspiration from them, these helped me get familiar with the specific syntax that Pygame adds into Python and also helped me with some of the ideas for the classes in my project.

<https://coderslegacy.com/python/pygame-rpg-game-tutorial/>

<https://www.youtube.com/playlist?list=PLjcN1EyupaQm20hlUE11y9y8EY2aXLpnv>

https://github.com/russs123/Shooter

<http://codingwithruss.com/pygame/shooter/music.html>

https://www.pygame.org/docs/ref/time.html

**Testing**

Player

a) The player can go forwards backwards and jump using A, D, Space and sprint with S , check all controls are correct

b) Player can deal damage to enemies using right click, however there should be a cooldown after

using an attack, no spam clicking

e) If the player is killed, they should respawn at the start of the level

h) The players health, attack and defence levels and number of gems should be displayed on the screen, check this updates, close check health bar is correct

Enemies

a) All enemies should be able to fight and deal damage to the player, check enemy’s do damage different based on level of enemy

b) All enemies should have a set amount of health and be able to be killed by the player, check can kill enemies

c) All enemies should drop item(s) after being killed, check right drops for right level of enemy

d) Enemies should not respawn after being defeated, unless the player dies, check that the enemy disappears when killed.

Crafting

a) The player should be able to craft, to upgrade attack and defence by using 1 2 3 4 5 6, check the defence and attack stats increase and gems decrease

check if attack and defence stats work, attack more damage, defence less damage taken

b) The player should only be able to increase a stat if they have the correct crystal to do so,check can’t increase stats with 0 gems

Attack and Defence

a) Attack and defence have levels 1-10, check can go higher than level 10

Chests and Drops

a) Chests will spawn in set locations of the map in each level, heck chests spawn and open

Check can only be opened once and the image changes to an open chest

b) Each chest will contain 1-2 random crystals or hearts, check right amount

e) Green magic crystals will increase the level of items by 1 level, red by 2 levels and blue by 3

levels, increase by correct levels

f) Hearts restore the players health, check this at health bar

1. <https://arks.itch.io/dino-characters>, <https://www.spriters-resource.com/>, https://opengameart.org/content/2d-tiles-0 [↑](#footnote-ref-2)
2. https://ezgif.com/sprite-cutter [↑](#footnote-ref-3)
3. https://www.photoroom.com/background-remover/ [↑](#footnote-ref-4)
4. https://pinetools.com/pixelate-effect-image [↑](#footnote-ref-5)
5. https://github.com/pygame/pygame [↑](#footnote-ref-6)