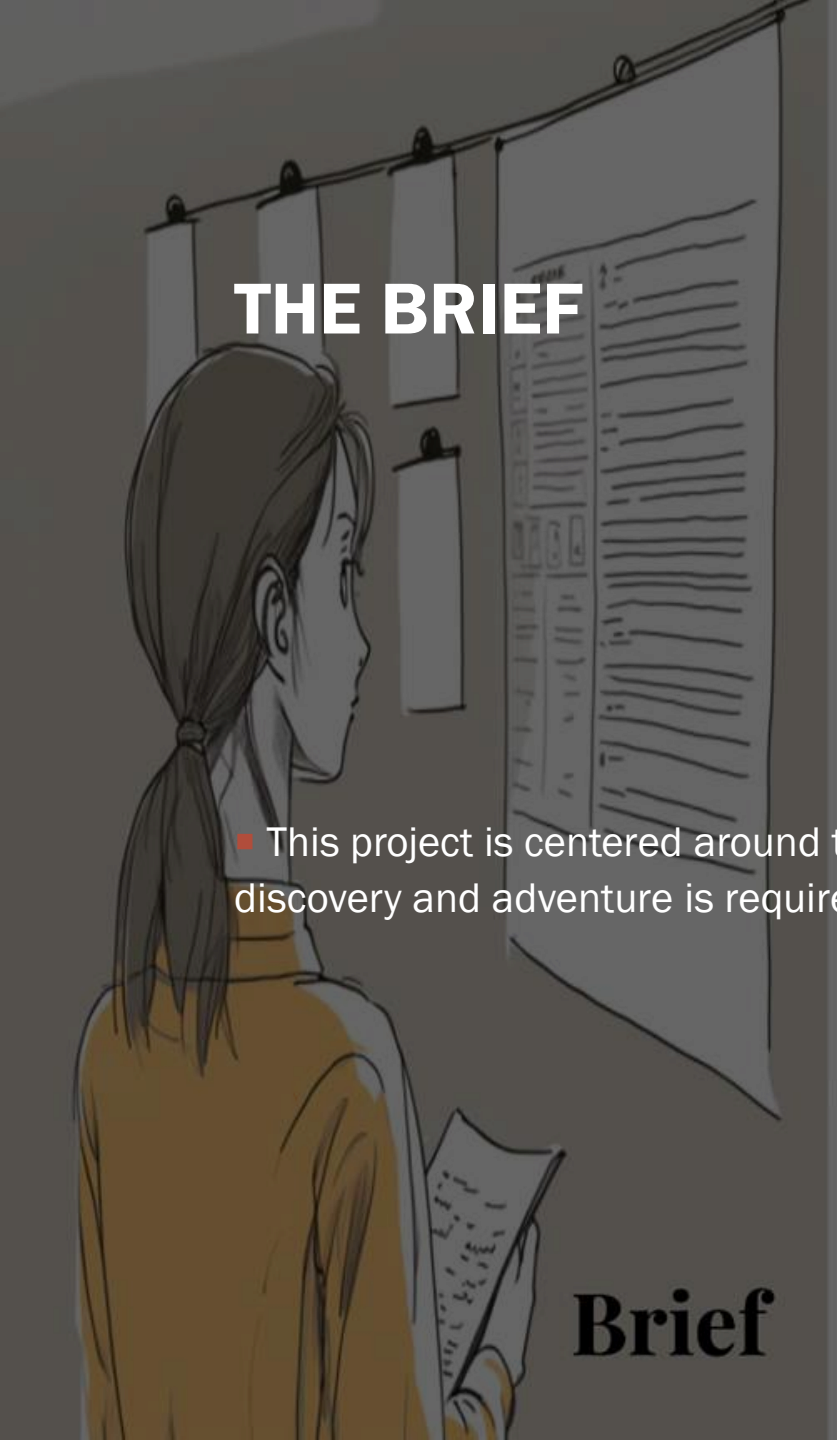
A2 SHOWCASE – DIG IT, DOUG! DEV LOG

BY CHARLIE COLL

THE BRIEF



Brief

Client
FCoT Games Studio

Context
FCoT Games, a new start up initiative, hopes to help students gain real world experience in games development to help them either get a job in industry or into higher education. We want to create a series of games that promote the idea of discovery and adventure

Requirements
Create a playable 2D or 3D game that promote the idea of discovery and adventure

Audience
Your audience is future students at FCoT (ages 16 – 20)

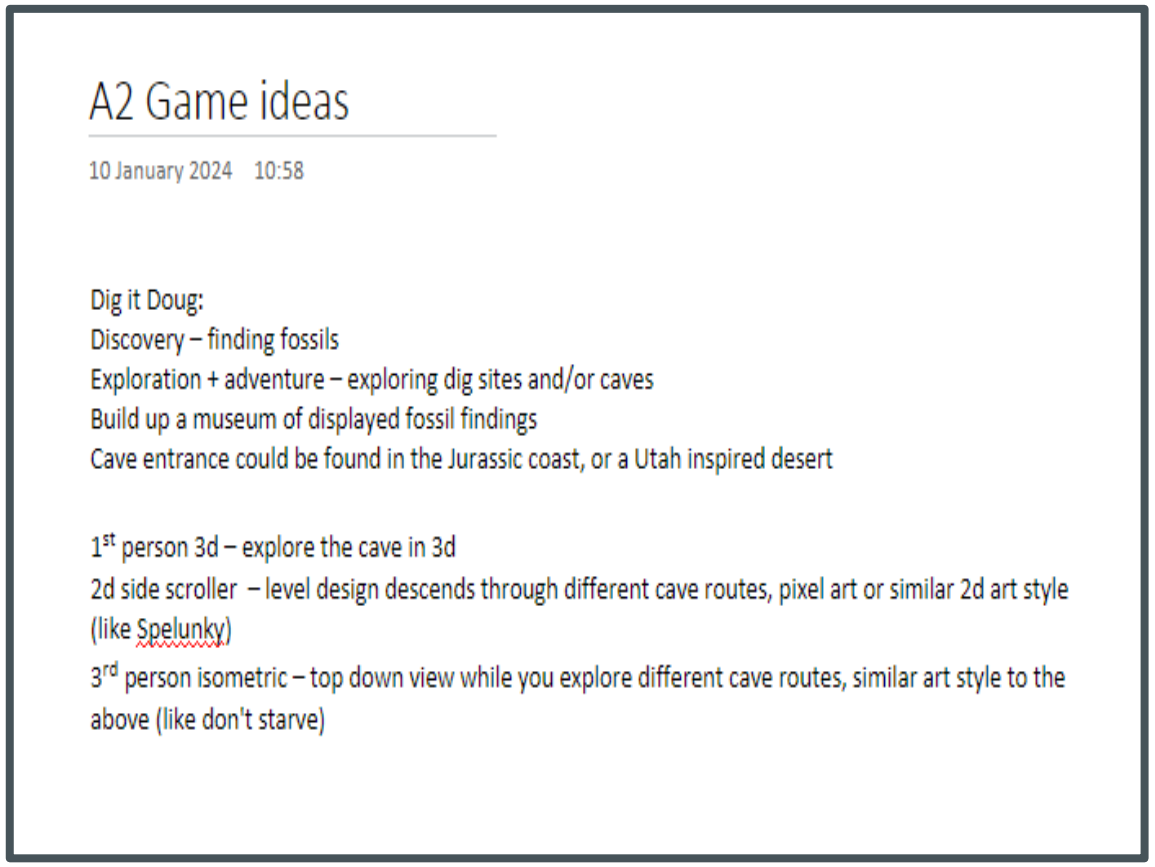
Deadline
01/05/2024

Resubmission deadline
28/05/2024

Assessor
Games lecturers.

- This project is centered around the brief that was issued, which states that a playable 2D or 3D game that promotes discovery and adventure is required – with a target audience of ages 16-20.

THE VERY BEGINNING – INFORMING IDEAS



A2 Game ideas

10 January 2024 10:58

Dig it Doug:
Discovery – finding fossils
Exploration + adventure – exploring dig sites and/or caves
Build up a museum of displayed fossil findings
Cave entrance could be found in the Jurassic coast, or a Utah inspired desert

1st person 3d – explore the cave in 3d
2d side scroller – level design descends through different cave routes, pixel art or similar 2d art style (like Spelunky)
3rd person isometric – top down view while you explore different cave routes, similar art style to the above (like don't starve)

- Prior to all the following official documentations and productions, this project began with pondering around the idea and concept of the game.
- A section from my class notebook is shown on the left, listing some initial ideas for a game centered around the brief that it is required to meet.

TIMELINE – WEEKS

1 2 3 4 5 6 7 8 9 10 11

CONTENTS – WEEK 1

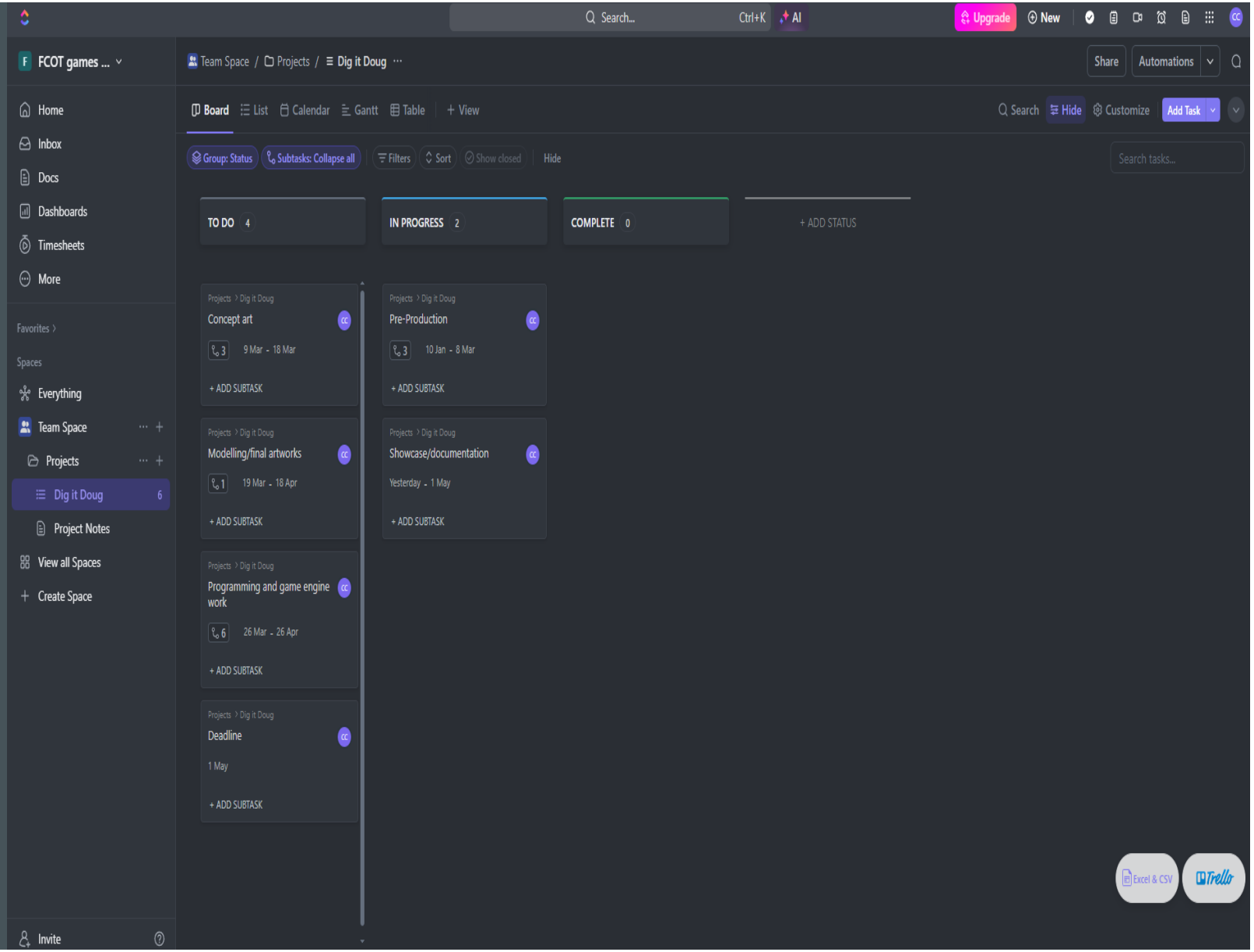
PROJECT MANAGEMENT

COMMUNICATION

ART

PROJECT MANAGEMENT – DAY 1-2

- The project officially commenced with the planning, which began on the week starting with the 19th of February, with the creation of the kanban board shown on the right via the use of click up.
- This organizes the development into four main phases, each of them containing subtasks like creating the game pitch or making a mood board, visualizing the whole process and scope of the project – while also being able to update the status of individual phases and subtasks with the kanban board.

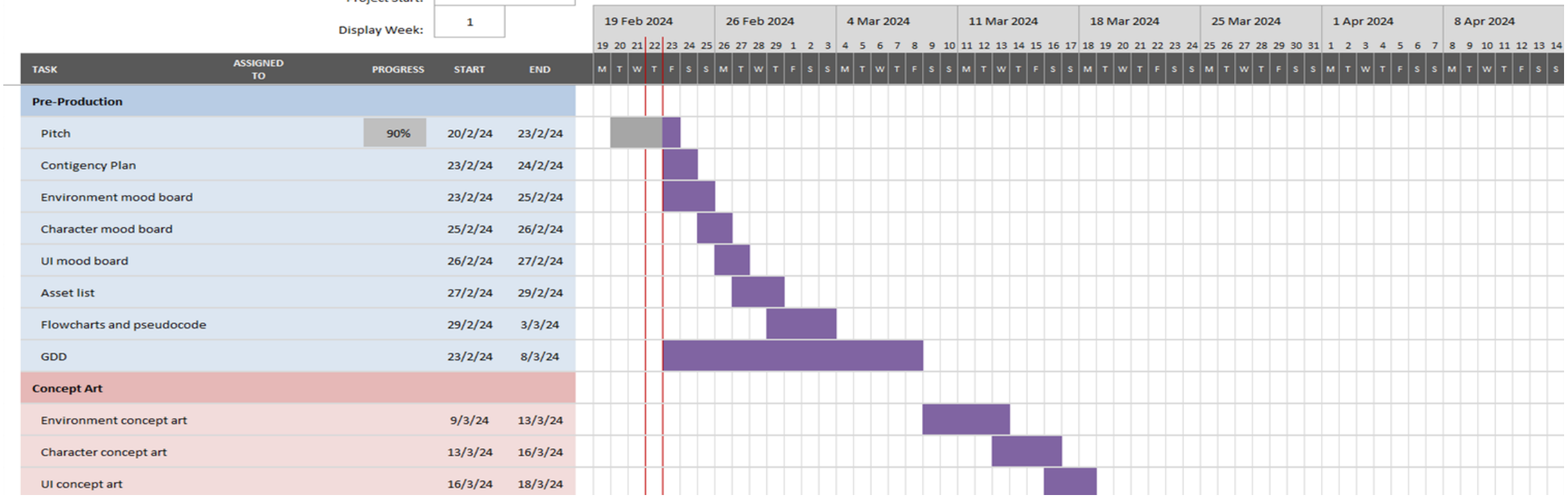


Dig it, Doug!

FCOT studios
Charlie Coll

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: Tue, 20/02/2024
Display Week: 1



TASK	ASSIGNED TO	PROGRESS	START	END
Pre-Production				
Pitch		90%	20/2/24	23/2/24
Contigency Plan			23/2/24	24/2/24
Environment mood board			23/2/24	25/2/24
Character mood board			25/2/24	26/2/24
UI mood board			26/2/24	27/2/24
Asset list			27/2/24	29/2/24
Flowcharts and pseudocode			29/2/24	3/3/24
GDD			23/2/24	8/3/24
Concept Art				
Environment concept art			9/3/24	13/3/24
Character concept art			13/3/24	16/3/24
UI concept art			16/3/24	18/3/24

PROJECT MANAGEMENT – DAY 1-2

- In tandem with click up I'm using excel to create this Gantt chart, which does the essential and useful job of visualizing the project's development chronologically into blocks that represent individual tasks across the dates that they are due in this descending graph – also showing the current day via the column that is highlighted red.

Elevator pitch

Explore the uncharted caves of the Jurassic coast

Discover fossils of prehistoric life you know, you don't know, and you really ought to know

GAME PITCH – DAY 2-5

- The first deadline was the game pitch, where I created a presentation that would act as the background and visual showcase while I verbally delivered the pitch itself from my notes.
- Above are the brief and elevator pitch slides, which were on the display behind me while I talked about the brief I had to fit and the basic summary of what my game is all about.

GAME PITCH – DAY 2-5

- This slide shows the mechanics and gameplay slides from the pitch, outlining the initial gameplay loop and mechanics of the game.

Mechanics

- Tight squeezes - QTE
- Fossil bag - tearing
- Chisel – fossil size
- Pickaxe
- Ropes and ladders

Gameplay

- Navigate cave systems and openings
- Find and retrieve fossils
- Return the findings to your dig site
- View your discoveries in your logbook
- Publish your logbook

INSPIRATIONS AND GOALS FOR AN EDUCATIONAL EXPERIENCE – DAY 5

- This sequence from an episode of “Sea monsters, a walking with dinosaurs trilogy” has the effect that I would like to achieve with this game, specifically with the journal, as it's an educational experience for the audience.
- Like when this episode transitions to a new sea and tells you what it is, how long ago it was, and what lived in it, I would like the journal in my game to educate the player about the fossils they discover.

Trigger	Action	Goal
Digital losses of work	Retrieve the latest version of the backed-up work – either incremental or from the full backup	To get to the most recent version of the lost work and get back to business as soon as possible
Some other obligation or event comes up that overrides and replaces scheduled dev time	Split up the lost dev time into new additional sessions throughout the week before or after	To make sure the task scheduled when the obligation arose is still completed in due time
Minor illness	Follow dev schedule as much as possible and rest otherwise	Recover as soon as possible
Major illness	Get the medical assistance required and inform the college about my ill state	Recover as soon as possible and get authorised absence from the college while receiving support from home
Damage to hardware	Fix or replace hardware and inform the college about time slots to use their computers, get the latest backup from the cloud or undamaged hardware	To have some hardware to work on temporarily while the damaged hardware is fixed

PROJECT MANAGEMENT – DAY 5

- I produced this contingency plan, providing plans of action to prevent any of these unfortunate events triggering significant disruption to the planned schedule during the rest of the project, meaning that I won't have to think about this at the time of a possible trigger and worry about multiple things all at once.

ENVIRONMENT MOOD BOARD – DAY 7

- The planning for the visuals began with this environment mood board, creating a collage of images that will be referenced when working on concept art in a few weeks.
- The images are arranged into different categories: The dig site, within the caves, Durdle door/above ground, video game examples, the lighthouse, the narrow tunnels, the cave entrance, and vertical gaps.
- The reason Durdle door is present so much is because I've personally been there and been inspired by it, the dig site is heavily inspired by one of my favorite films – John Hammond's caravan in Jurassic Park, and the rest of the images are pulled from the Jurassic coast or related searches.

CONTENTS – WEEK 2

ART

PROJECT MANAGEMENT

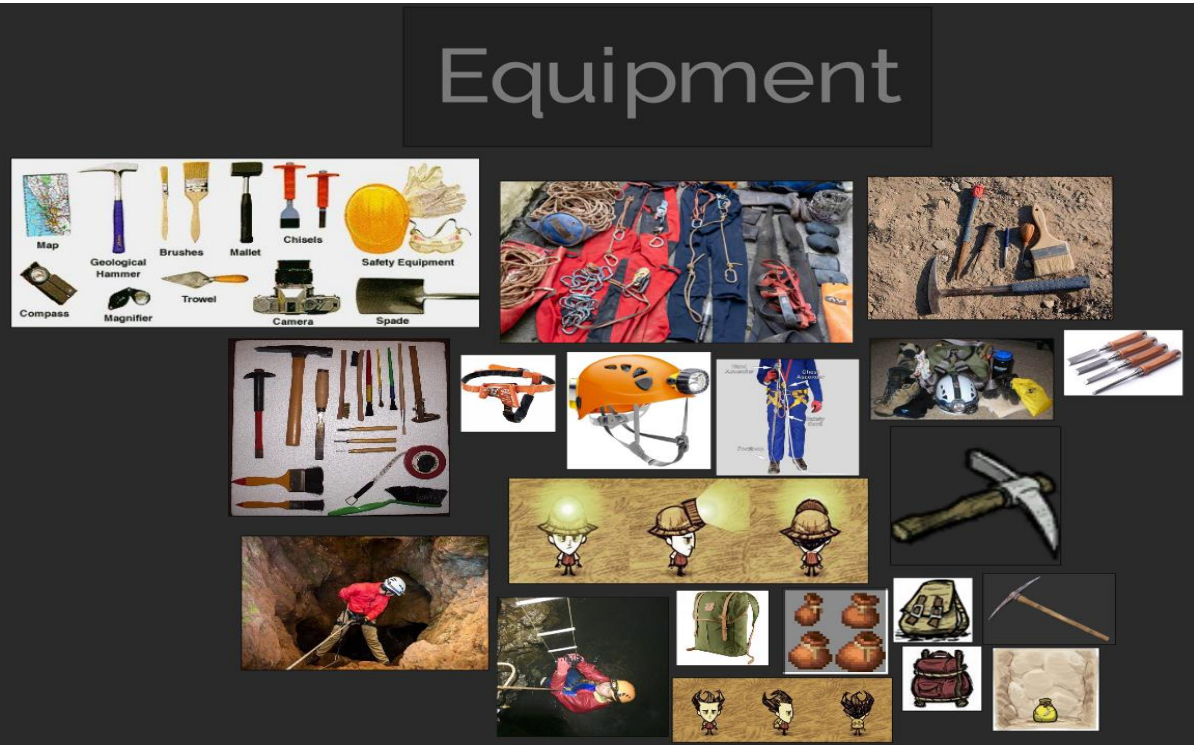
COMMUNICATION

CHARACTER MOOD BOARD – DAY 8

- The next mood board scheduled was for the characters, this board is more general in terms of its arrangement than the environment mood board – featuring Alan Grant and John Hammond from Jurassic Park, Mr. B the gentleman rhymer, Steve Backshall (English naturalist and presenter of Deadly 60, a show I watched growing up), Steve Irwin (Australian zookeeper that I also watched growing up), related game characters, people with a related look, and Nigel Marven (British television presenter and naturalist, featured in walking with dinosaurs specials such as sea monsters and the giant claw – massive inspirations!)

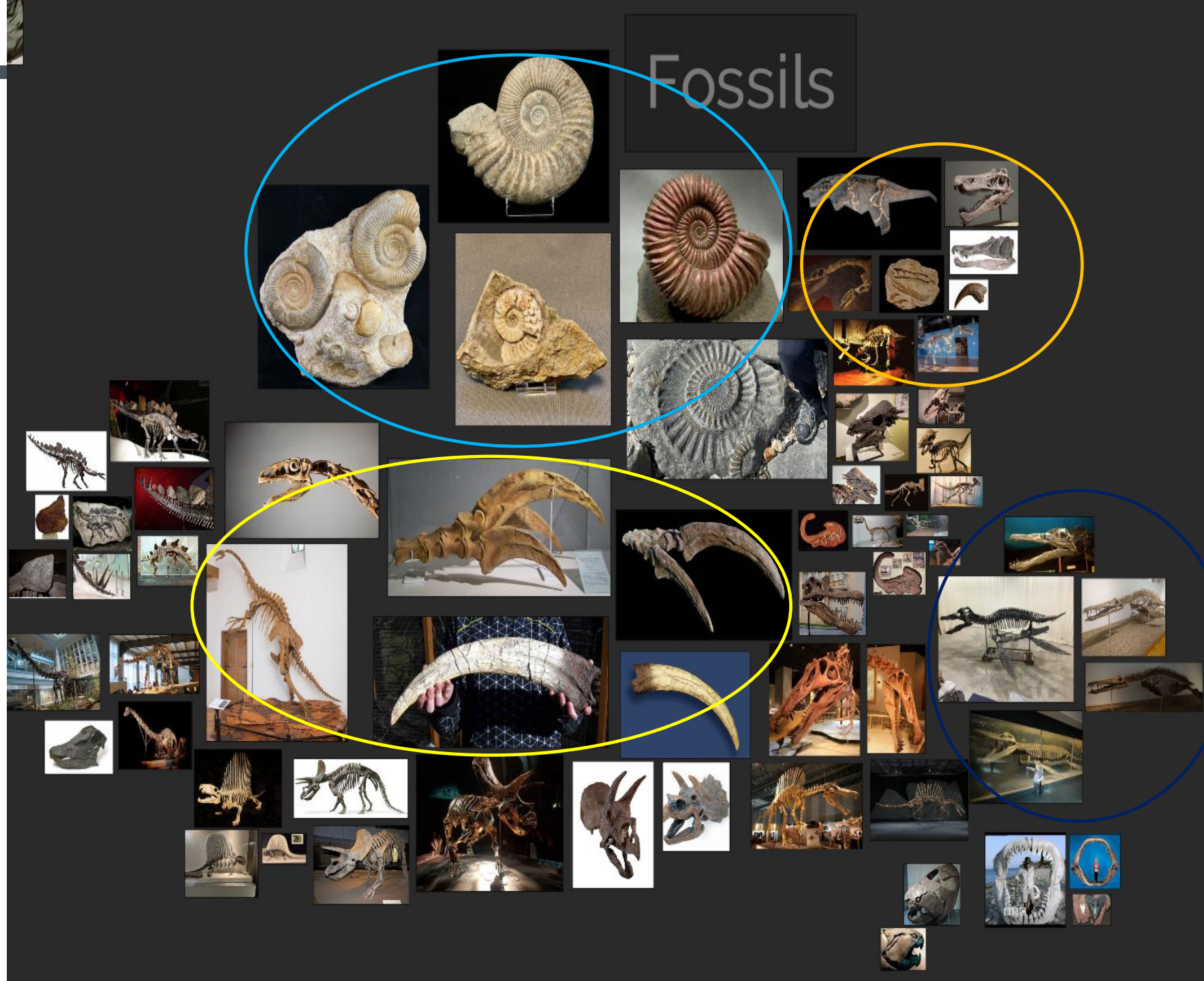
EQUIPMENT + UI MOOD BOARD – DAY 9

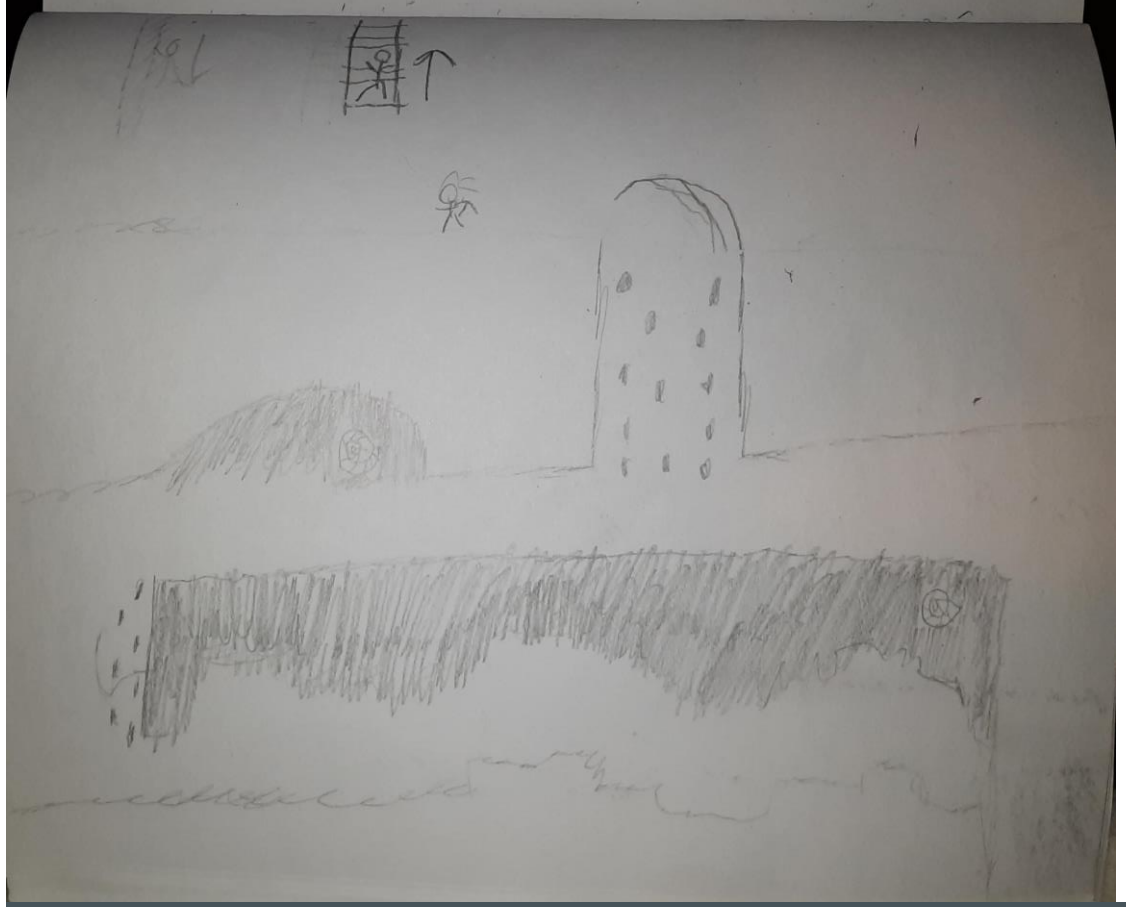
- The following day I created the rest of the mood board I needed – I'd initially planned to only create a UI mood board, but I realized that I would need to also have a mood board for equipment and fossils.
- The UI board contains the journal from Spelunky that will influence the journal in my own game, some UI elements of the game carnivores dinosaur hunter, fossil expedition menus from the game Jurassic world evolution 2, a don't starve menu, and the discoveries menu from No man's sky (all being games I've played that inspire this project)
- The equipment mood board is a collection of paleontology and caving tool images I found on the internet, along with some equipment from related games.

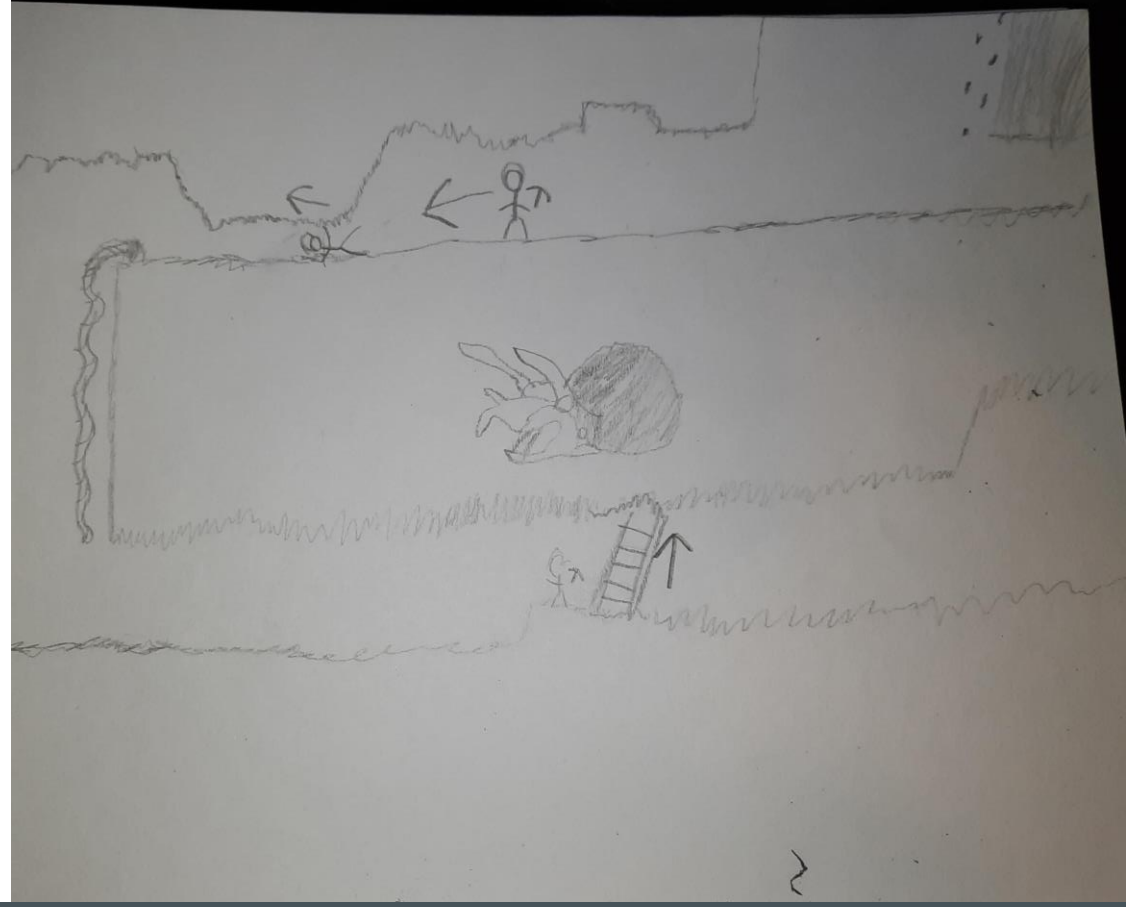


FOSSIL MOOD BOARD – DAY 9

- The final mood board was for the fossils, all of which may or may not be included in the game.
- Four of these are already concluded to be in the game: Ammonite, Baryonx, Therizinosaurus, and Liopleurodon. The ammonite will be the small and introductory fossil, the Baryonyx was discovered in the UK (where the game is based), and the other two are some personal favorites.
- The rest are some other fossils I'd like to add to the game, I'll most likely only have time for a few so the ones that will be implemented will be determined by an upcoming survey.





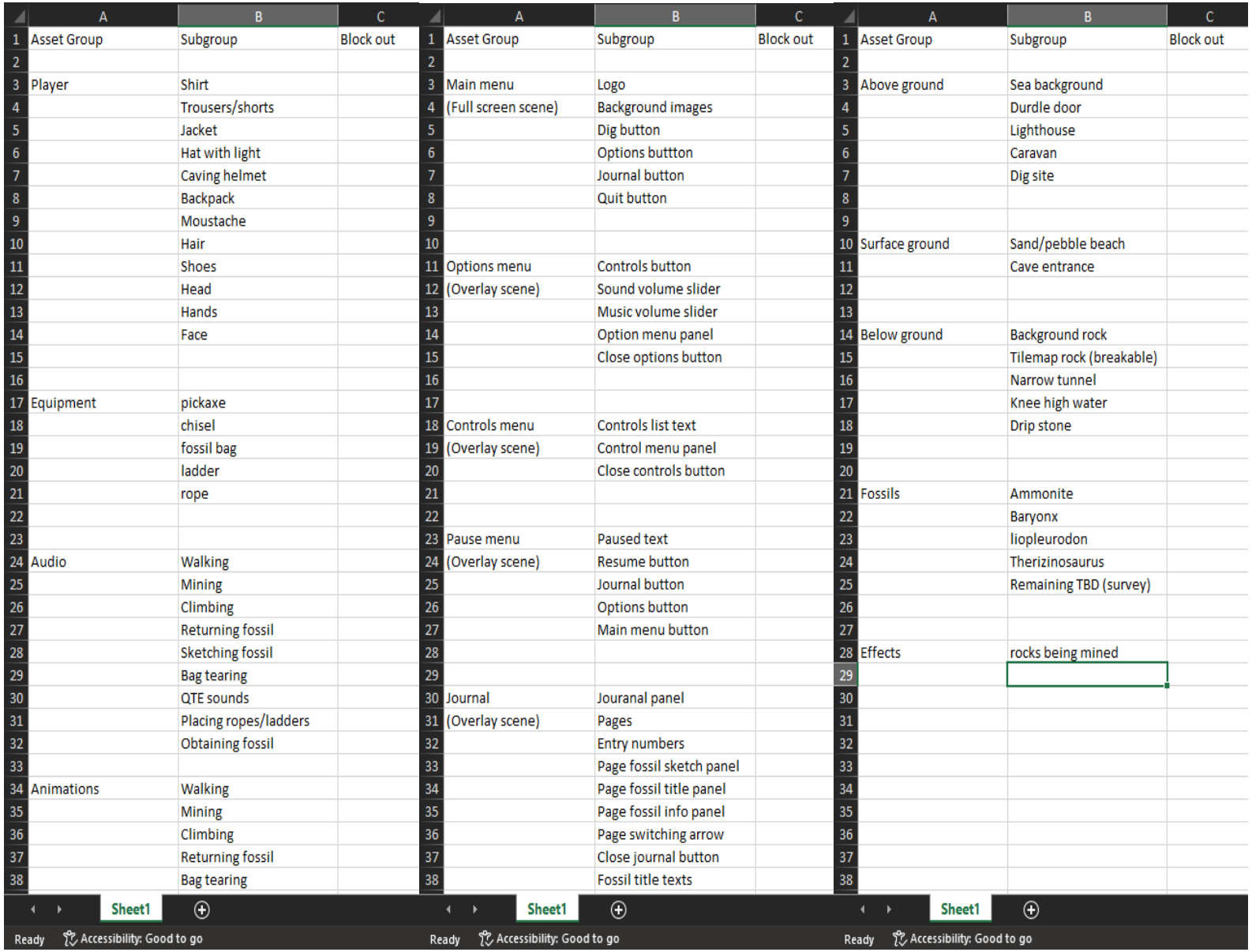


LEVEL DESIGN – DAY 10

I SKETCHED MY LEVEL DESIGN IDEAS DOWN ONTO THESE TWO PAGES, THE FIRST SHOWING THE CAVE ENTRANCE AND THE INITIAL AMMONITE FOSSIL THE PLAYER WILL FIND AND THE SECOND PAGE SHOWING THE DEEPER PARTS OF THE CAVE WITH NARROW TUNNELS AND LARGER FOSSILS, BOTH PAGES INCLUDE ROPES AND LADDERS IN THE LEVEL – WHICH ARE THE VERTICAL METHOS OF LOCOMOTION IN THIS GAME.

ASSET LISTS– DAY 10-11

- I then created my initial asset lists, accounting for every asset that will need to be created or added, ranging from character elements, menu buttons, and environmental features.
- This will certainly have new iterations in the future, as I'll finalize the fossil list once I've completed my survey and most likely add or remove some things due to the scope of the project.



Asset Group	Subgroup	Block out
Player	Shirt	
	Trousers/shorts	
	Jacket	
	Hat with light	
	Caving helmet	
	Backpack	
	Moustache	
	Hair	
	Shoes	
	Head	
	Hands	
	Face	
Equipment	pickaxe	
	chisel	
	fossil bag	
	ladder	
	rope	
Audio	Walking	
	Mining	
	Climbing	
	Returning fossil	
	Sketching fossil	
	Bag tearing	
	QTE sounds	
	Placing ropes/ladders	
	Obtaining fossil	
Animations	Walking	
	Mining	
	Climbing	
	Returning fossil	
	Bag tearing	

Sheet1 Ready Accessibility: Good to go

UI

Asset Group	Subgroup	Block out
Main menu	Logo	
(Full screen scene)	Background images	
	Dig button	
	Options buttton	
	Journal button	
	Quit button	
Options menu	Controls button	
(Overlay scene)	Sound volume slider	
	Music volume slider	
	Option menu panel	
	Close options button	
Controls menu	Controls list text	
(Overlay scene)	Control menu panel	
	Close controls button	
Pause menu	Paused text	
(Overlay scene)	Resume button	
	Journal button	
	Options button	
	Main menu button	
Journal	Jouranal panel	
(Overlay scene)	Pages	
	Entry numbers	
	Page fossil sketch panel	
	Page fossil title panel	
	Page fossil info panel	
	Page switching arrow	
	Close journal button	
	Fossil title texts	

Sheet1 Ready Accessibility: Good to go

Character

Asset Group	Subgroup	Block out
Above ground	Sea background	
	Durdle door	
	Lighthouse	
	Caravan	
	Dig site	
Surface ground	Sand/pebble beach	
	Cave entrance	
Below ground	Background rock	
	Tilemap rock (breakable)	
	Narrow tunnel	
	Knee high water	
	Drip stone	
Fossils	Ammonite	
	Baryonx	
	liopleurodon	
	Therizinosaurus	
	Remaining TBD (survey)	
Effects	rocks being mined	

Sheet1 Ready Accessibility: Good to go

Environment

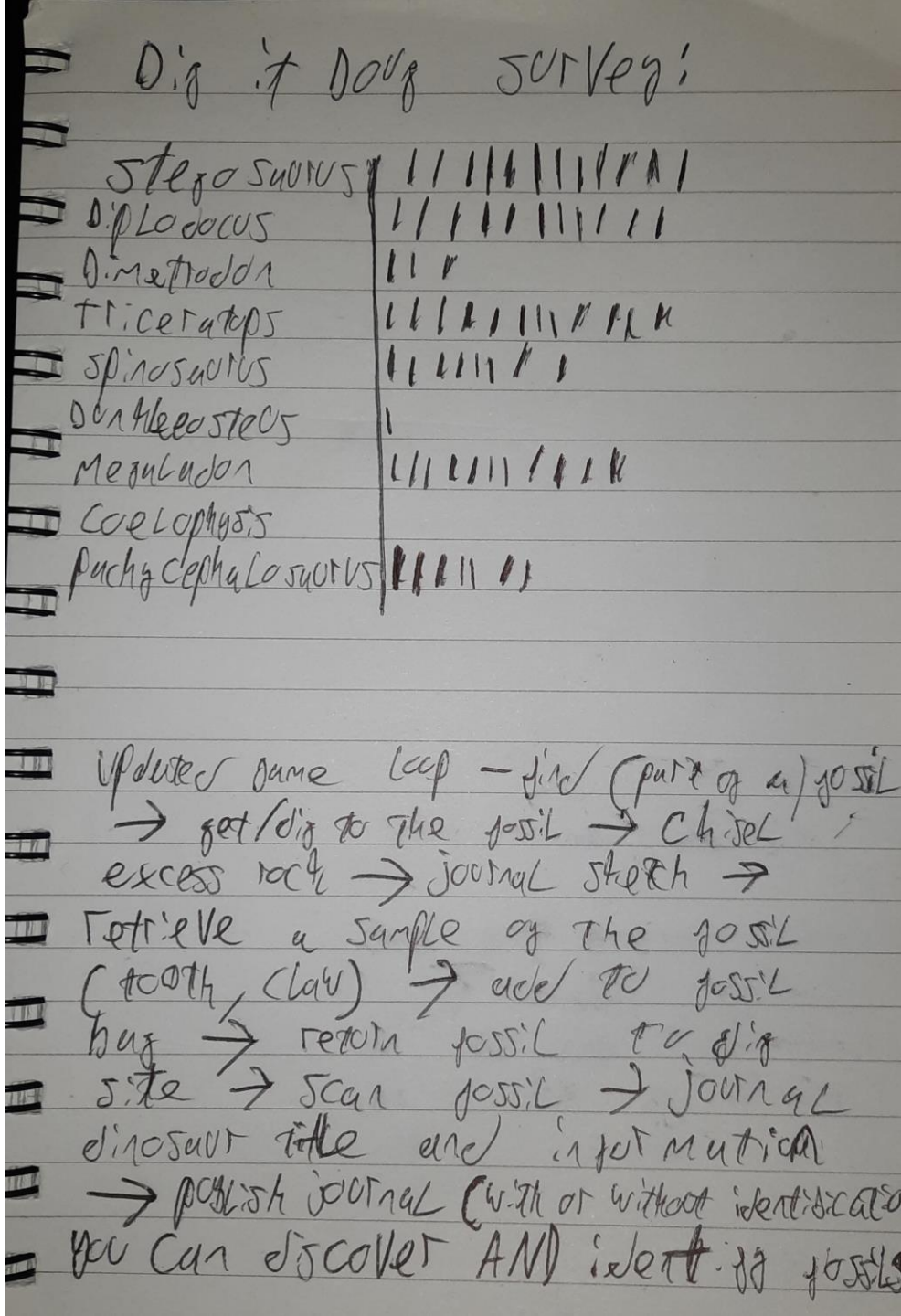
GAME DESIGN DOCUMENT– DAY 11

- I started the game design document, just creating the first page with the elevator pitch and basic gameplay loop, to get the initial momentum going for making the design document so I can continue adding to it soon.

SURVEY AND GAME LOOP – DAY 11-12

- I conducted this survey over these two days, asking my parents, friends, and peers about what prehistoric life they know about from this list – the goal of this is to determine which fossils will be added into the game, as I will add the creatures with the least amount of recognition from the survey, so the chosen fossils are not generally known, and the game can inform the player about new prehistoric life forms that they wouldn't of known about before playing the game (further leaning into the theme of discovery).

- Below is an updated version of the game loop – I made some small changes to it that I believe make more sense and enhance the experience. The difference between this and the loop seen in the pitch is that you obtain the sketch upon finding the fossil, and then you obtain the title of the fossil and information regarding it upon returning a sample of the fossil to the dig site and scanning it – giving depth to the journal and allowing you to discover a fossil with the ability to publish either just the sketch or the title and information along with it, this also allows only taking back fossils that fit into a small bag make sense.



Dig it, Doug!

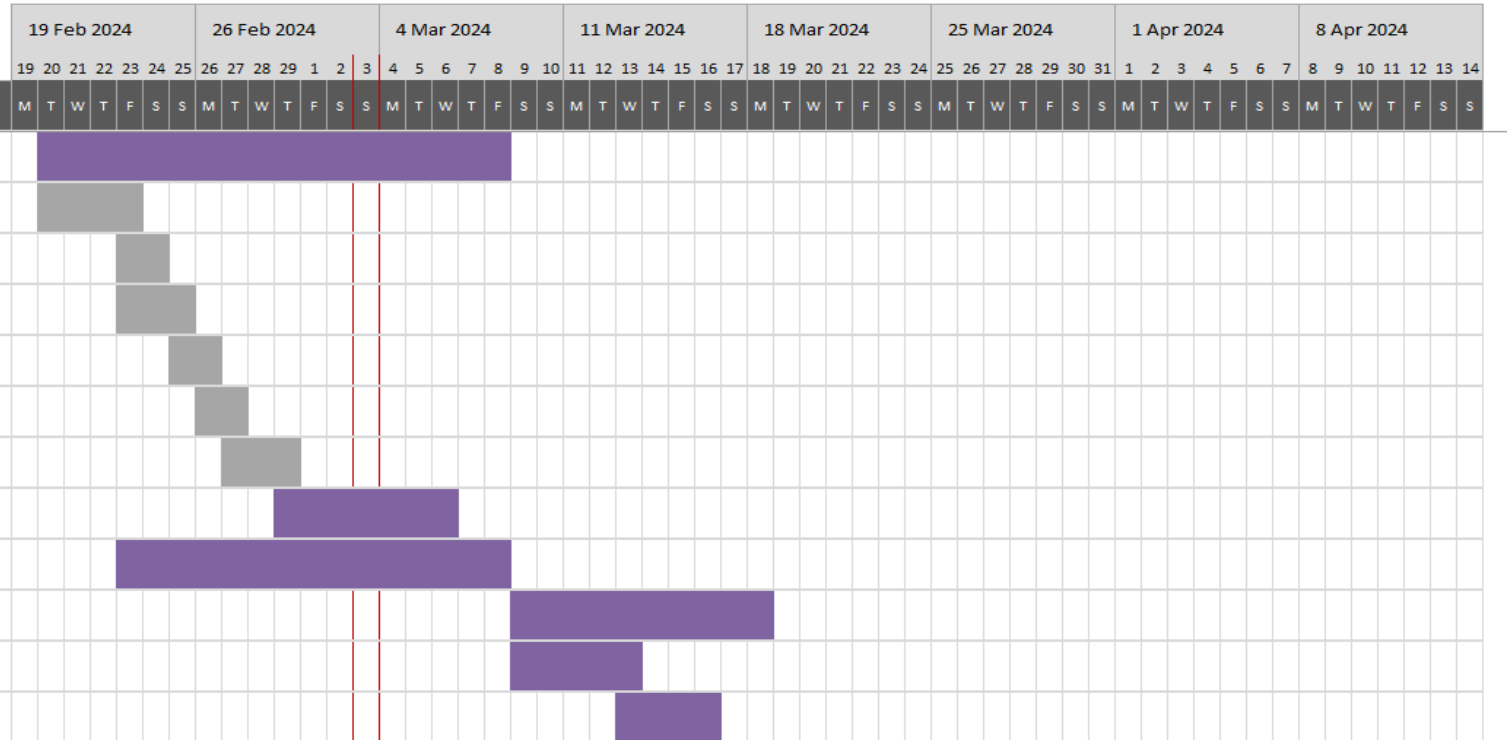
FCOT studios
Charlie Coll

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: Tue, 20/02/2024

Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Pre-Production			20/2/24	8/3/24
Pitch		100%	20/2/24	23/2/24
Contigency Plan		100%	23/2/24	24/2/24
Environment mood board		100%	23/2/24	25/2/24
Character mood board		100%	25/2/24	26/2/24
UI mood board		100%	26/2/24	27/2/24
Asset list		100%	27/2/24	29/2/24
Flowcharts and pseudocode			29/2/24	6/3/24
GDD			23/2/24	8/3/24
Concept Art			9/3/24	18/3/24
Environment concept art			9/3/24	13/3/24
Character concept art			13/3/24	16/3/24



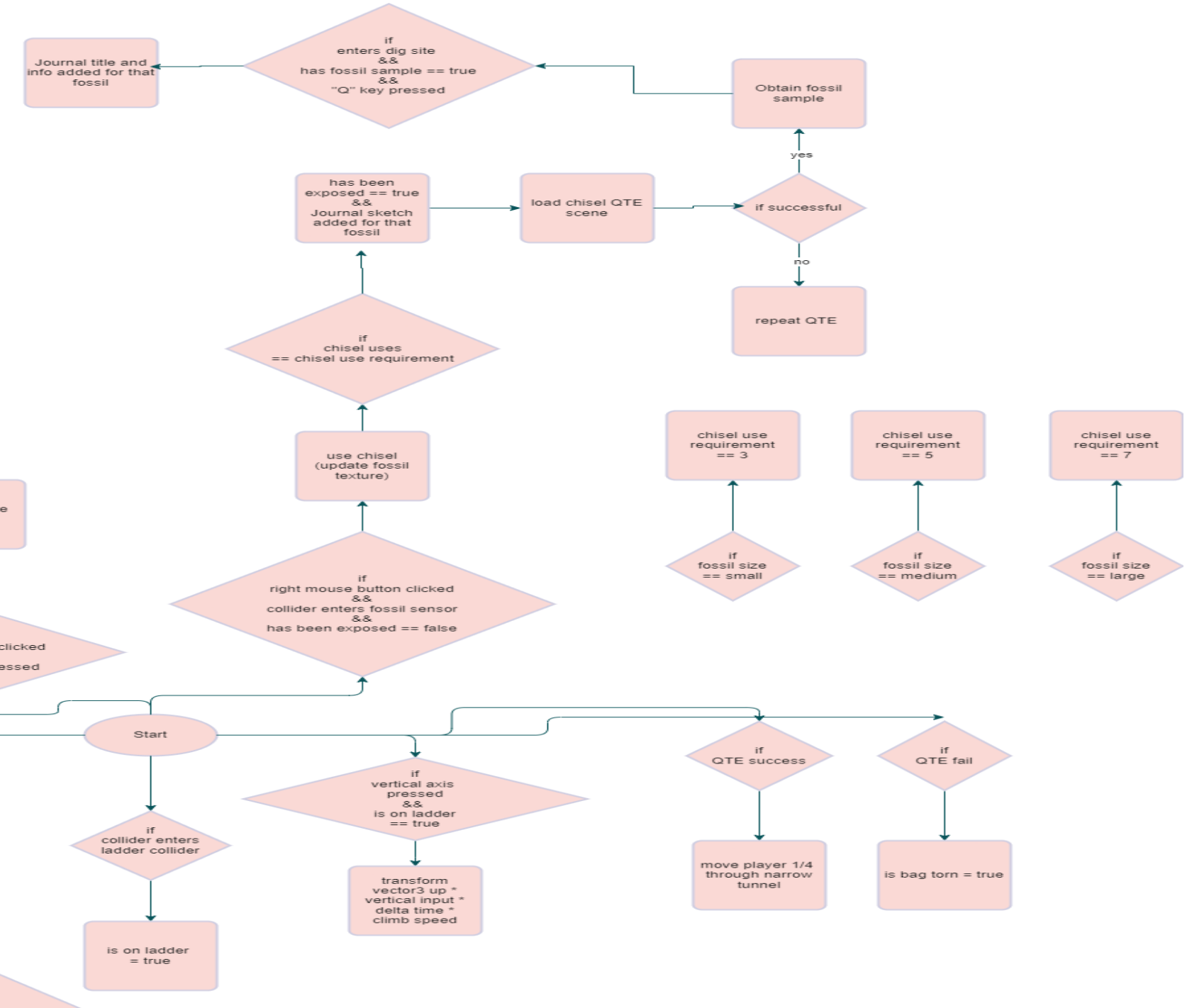
GANTT CHART UPDATE– DAY 14

- I made a small modification to my Gantt chart, slightly extending the due date for the programming planning, as I've been working on getting this dev log up to date while I would have been making flow charts – due to only starting the dev log on day 11.

CONTENTS – WEEK 3

PROJECT MANAGEMENT

MILESTONE



FLOWCHARTS – DAY 15

- This is part of the first iteration of the player controller flow chart, with the logic for everything involving the input from the player, which will be changed due to it currently having three unconnected elements – as I'm not currently sure if these would be in the player controller script or a dedicated fossils script.

GAME DESIGN DOCUMENT – DAY 16

- Now continuing properly with the game design document, I added some visual influences, core mechanics, the core gameplay loop, and the asset lists to the document – with this being a blueprint for the game in mind and imagining if someone else were to be given this then they would be able to make the game exactly to the vison I have for it.

Dig it, Doug!

1.0

Elevator Pitch:

- Explore the uncharted caves of the Jurassic coast.
- Discover fossils of prehistoric life you know, you don't know, and you really ought to know.

Gameplay:

- Navigate cave systems and openings.
- Find and retrieve fossils.
- Return the findings to your dig site.
- View your discoveries in your logbook.
- Publish your logbook.

Visual Influences:

Spelunky for the 2D platforming and below ground elements.

Core Mechanics:

Tight squeezes – get through via successful quick time events.

Fossil bag – tears upon failing a QTE.

Chisel – used to remove small amounts of excess rock around a fossil, the number of times you need to use it depends on the size of the fossil.

Pickaxe – used to break away large amounts of rock that separates the player from fossils.

Ropes + ladders – can be placed to traverse vertical gaps in the caves.

Core Gameplay Loop:

Find fossil – get/dig to the fossil – chisel excess rock – add fossil sketch to journal – retrieve a sample of the fossil (tooth, claw) – add to the fossil bag – return fossil sample to the dig site – scan the fossil – add fossil title and information to journal – publish journal (with or without identification)

You can discover AND identify fossils.

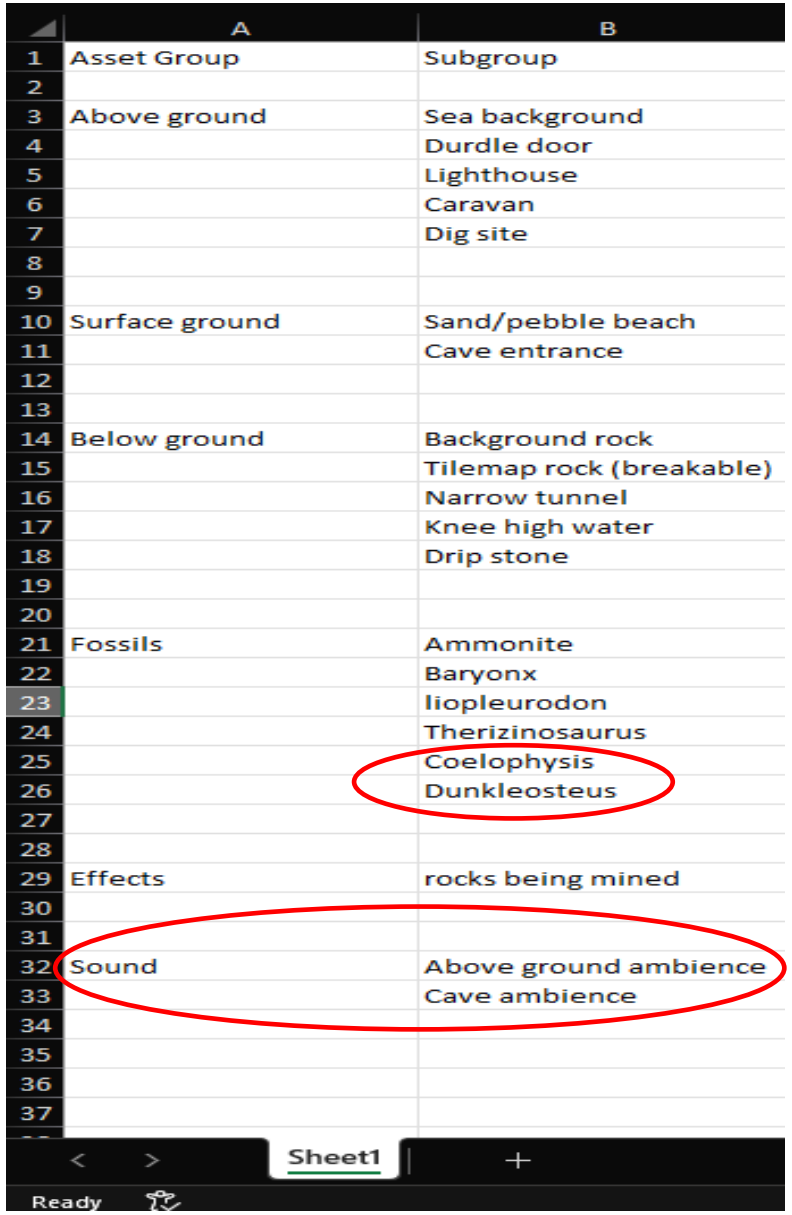
Assets:

Character:

Asset Group	Subgroup
Player	Shirt Trousers/shorts Jacket Hat with light Caving helmet

ASSET LIST UPDATE – DAY 16

- As the survey is now complete, I added the final fossils to the environment asset list – I kept it to only two additional fossils for a reasonable scope. I also added some environment ambience to the asset list, and main menu music to the UI asset list, all under the sound asset group.

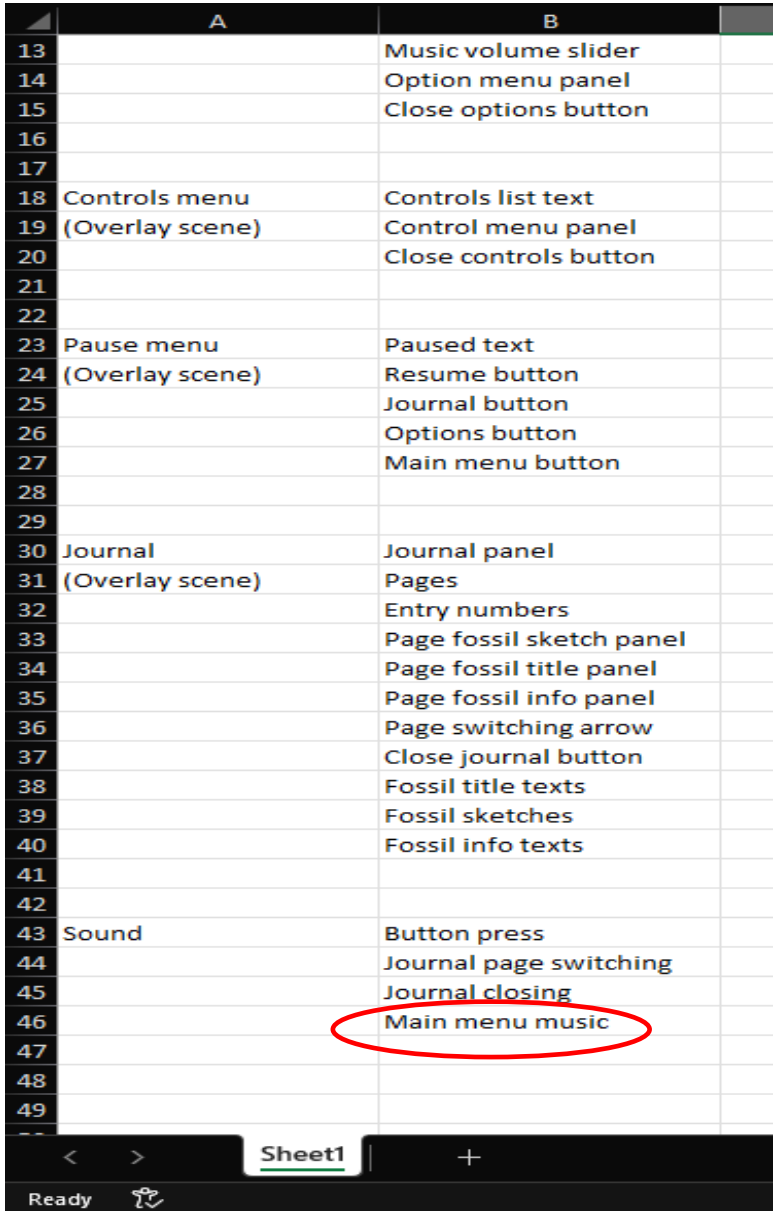


	A	B
1	Asset Group	Subgroup
2		
3	Above ground	Sea background
4		Durdle door
5		Lighthouse
6		Caravan
7		Dig site
8		
9		
10	Surface ground	Sand/pebble beach
11		Cave entrance
12		
13		
14	Below ground	Background rock
15		Tilemap rock (breakable)
16		Narrow tunnel
17		Knee high water
18		Drip stone
19		
20		
21	Fossils	Ammonite
22		Baryonx
23		liopleurodon
24		Therizinosaurus
25		Coelophysis
26		Dunkleosteus
27		
28		
29	Effects	rocks being mined
30		
31		
32	Sound	Above ground ambience
33		Cave ambience
34		
35		
36		
37		

Sheet1

Ready

Environment



	A	B
13		Music volume slider
14		Option menu panel
15		Close options button
16		
17		
18	Controls menu	Controls list text
19	(Overlay scene)	Control menu panel
20		Close controls button
21		
22		
23	Pause menu	Paused text
24	(Overlay scene)	Resume button
25		Journal button
26		Options button
27		Main menu button
28		
29		
30	Journal	Journal panel
31	(Overlay scene)	Pages
32		Entry numbers
33		Page fossil sketch panel
34		Page fossil title panel
35		Page fossil info panel
36		Page switching arrow
37		Close journal button
38		Fossil title texts
39		Fossil sketches
40		Fossil info texts
41		
42		
43	Sound	Button press
44		Journal page switching
45		Journal closing
46		Main menu music
47		
48		
49		

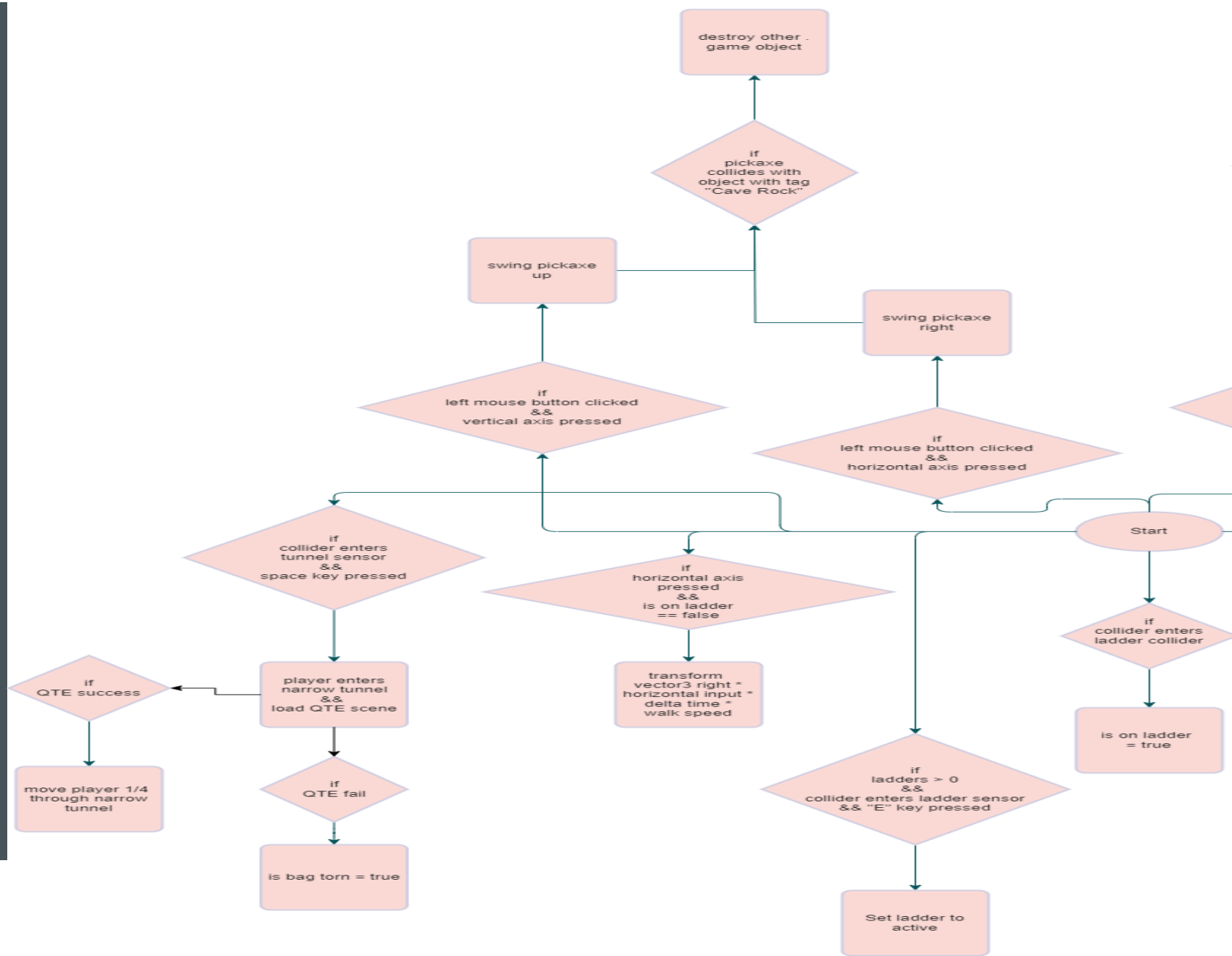
Sheet1

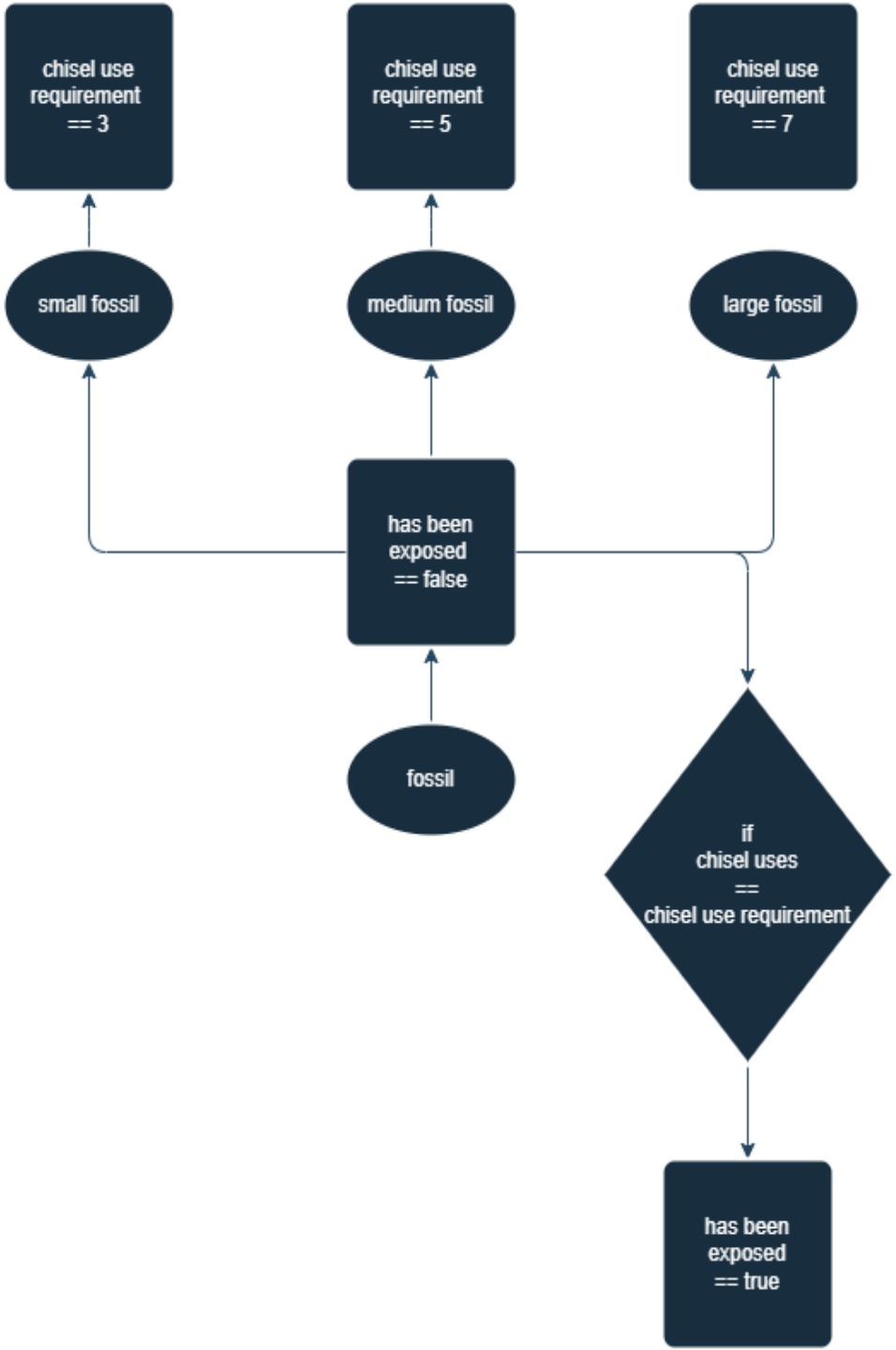
Ready

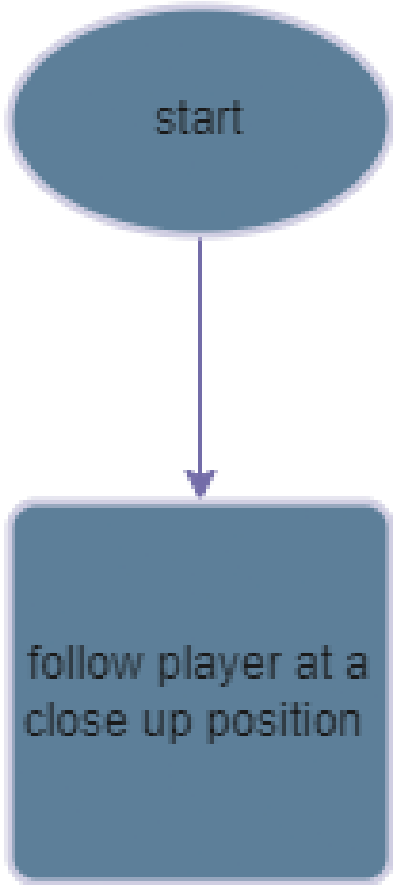
UI

FLOWCHARTS– DAY 16

- I finished the player controller flow chart by removing the detached elements to a different flowchart, as I concluded that their logic was for the fossils only and not suited for the player controller.







FLOWCHARTS – DAY 16

- The dedicated fossils logic involves inheritance, so the fossil size and its chisel use requirement can be in their own child scripts that will be assigned to the differently sized fossils – while still having the other logic passed down from the universal fossil script.

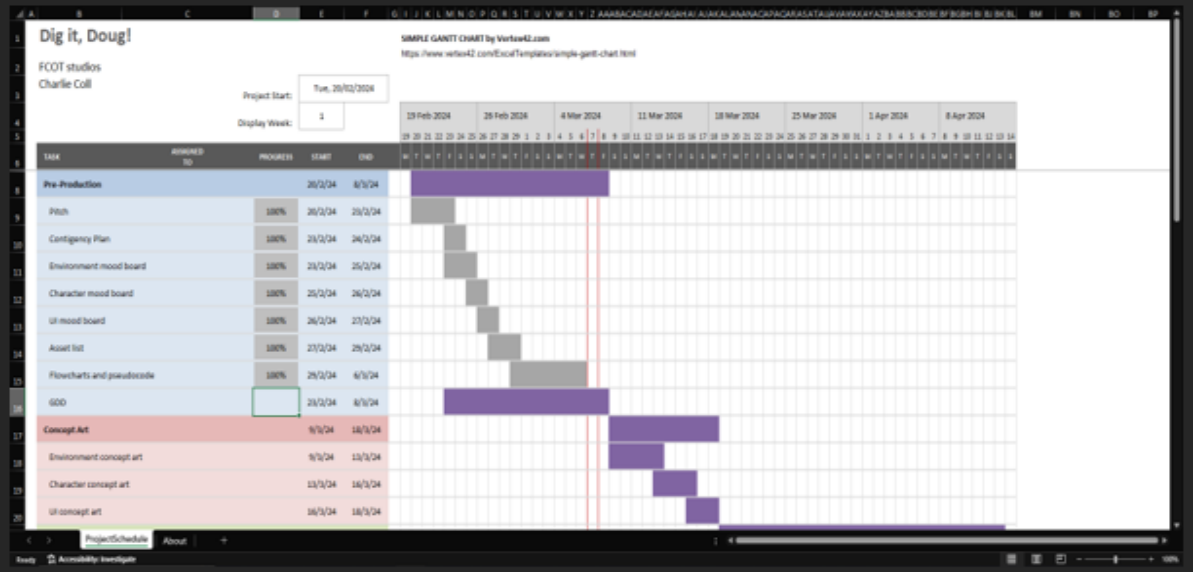
GAME DESIGN DOCUMENT – DAY 17

- I briefly returned to the game design document to add in the rest of the images to the visual influences section – I hadn't added all the images while I initially made the section due to not having the software that allows me to access my mood boards at college, which I use at home.

Jurassic world the game for quick time events.

John Hammond's caravan and Alan Grant's dig site (Jurassic Park) for the player's fossil returning and scanning.

Schedule:



Phase deadlines:

Pre-production – 8/3/24

Concept art – 18/3/24

Final visuals and assets – 18/4/24

Programming and game engine work – 26/4/24

Target Audience:

Ages 16-20, the game will most likely be enjoyed by all ages who have an interest in prehistoric life, dinosaurs, or who want to learn something new.

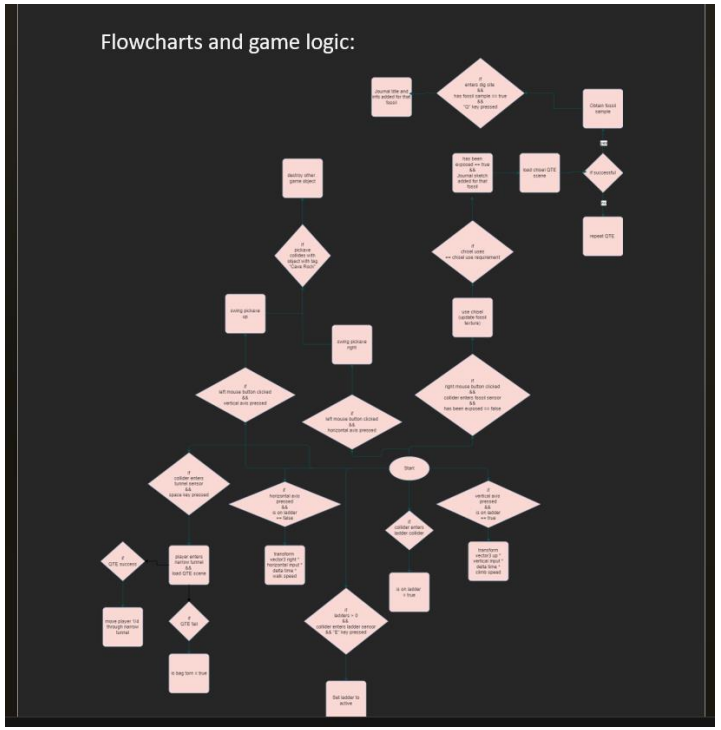
The explorer demographic of players.

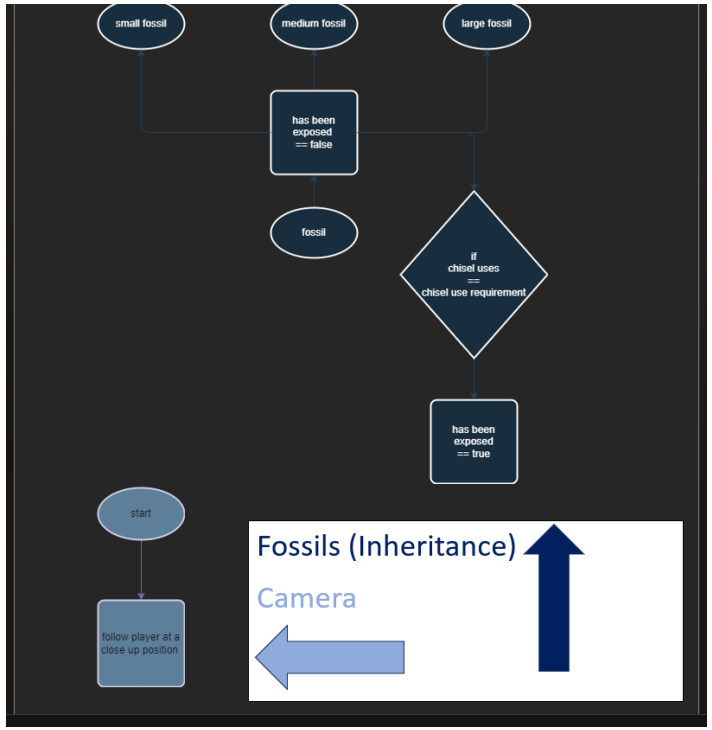
Targeted Platform:

PC.

GAME DESIGN DOCUMENT – DAY 18

- I completed the 1st iteration of the GDD by adding the production schedule and target audience/platform, the future iterations will include the flowcharts I've made and all the new progress I'll be making along the way – like concept art and prototypes of the game – but for now the version of the document with the initial pre-production is complete.



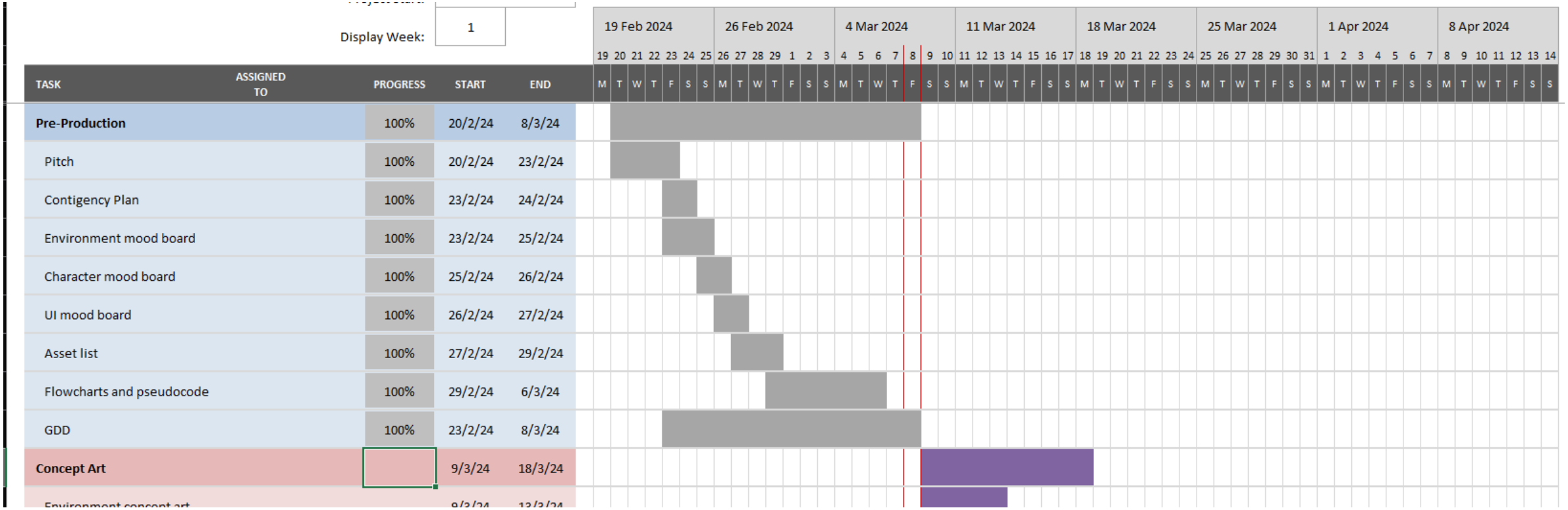


GAME DESIGN DOCUMENT – DAY 18

- I then worked on version 1.1 of the document, adding in the flowcharts and game logic under a new section, this content is rightly part of a new iteration as it covers something that has been previously mentioned in a basic fashion with greater detail.

Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Pre-Production		100%	20/2/24	8/3/24
Pitch		100%	20/2/24	23/2/24
Contigency Plan		100%	23/2/24	24/2/24
Environment mood board		100%	23/2/24	25/2/24
Character mood board		100%	25/2/24	26/2/24
UI mood board		100%	26/2/24	27/2/24
Asset list		100%	27/2/24	29/2/24
Flowcharts and pseudocode		100%	29/2/24	6/3/24
GDD		100%	23/2/24	8/3/24
Concept Art			9/3/24	18/3/24
Environment concept art			9/3/24	13/3/24



MILESTONE – DAY 19

Today I officially marked the pre-production phase with 100% completion, after confirming that the 1st iteration of the GDD contained everything it needed for me to mark it as finished (and therefore conclude the pre-production phase)

GANTT CHART UPDATE – DAY 19

- Before moving onto the next phase, I decided to change the start date of the programming and game engine work phase to begin on the 9th of February alongside the concept art phase – allowing more time for developing the prototypes and the game in general, as the first prototypes don't require the finished assets and can be worked on while I develop my concept arts. I made sure to add the subtasks to the programming phase, making the individual goals clearer for when this phase begins and outlining what must be done during it. I also changed the name of the modelling/final art works phase to "final sprite and asset development", as this is more accurate to what the phase will involve, due to 2D games like the game I'm making requiring sprites instead of models.

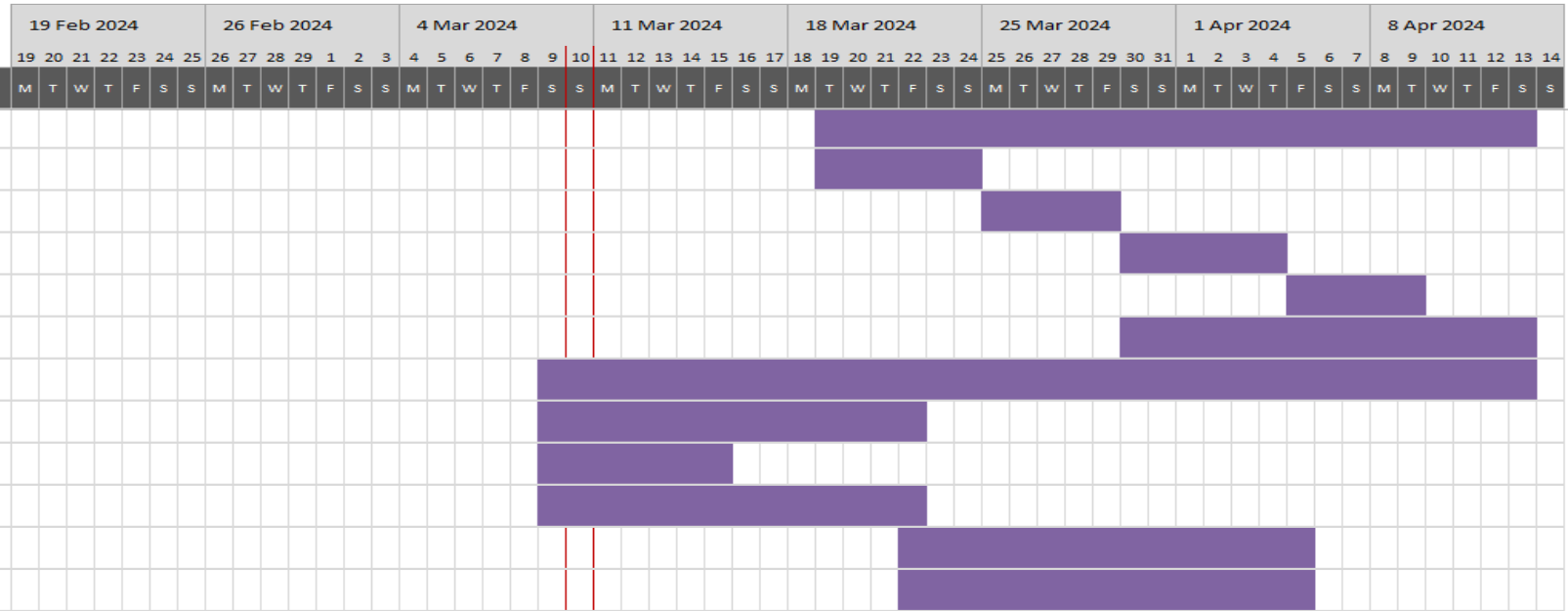
Dig it, Doug!

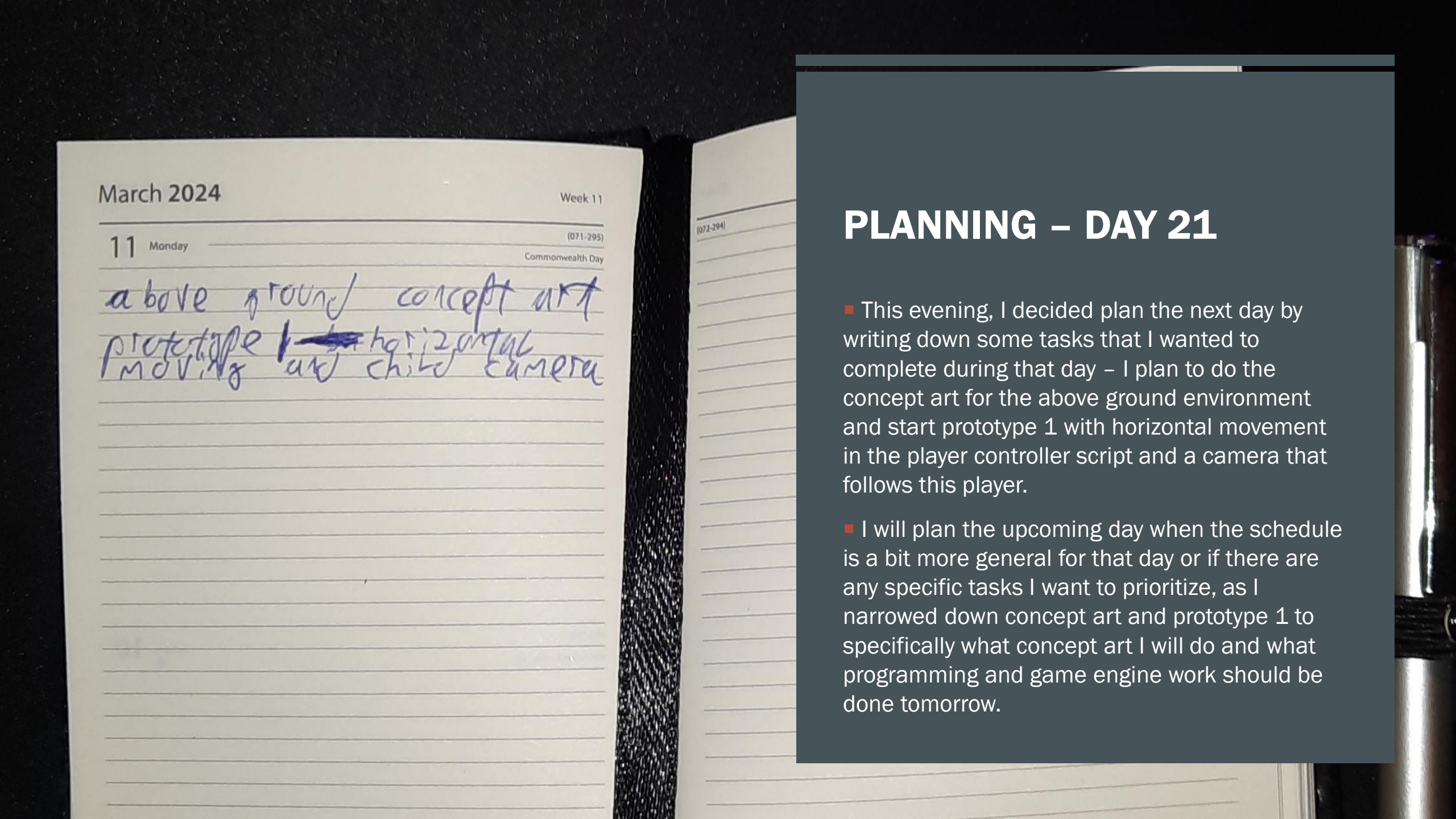
FCOT studios
Charlie Coll

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: Tue, 20/02/2024
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Final Sprite and asset development			19/3/24	18/4/24
Task 1			19/3/24	24/3/24
Task 2			25/3/24	29/3/24
Task 3			30/3/24	4/4/24
Task 4			5/4/24	9/4/24
Task 5			30/3/24	18/4/24
Programming and game engine work			9/3/24	26/4/24
Prototype 1			9/3/24	22/3/24
Camera implementation			9/3/24	15/3/24
Player controller script			9/3/24	22/3/24
Prototype 2			22/3/24	5/4/24
Tile maps			22/3/24	5/4/24





PLANNING – DAY 21

- This evening, I decided plan the next day by writing down some tasks that I wanted to complete during that day – I plan to do the concept art for the above ground environment and start prototype 1 with horizontal movement in the player controller script and a camera that follows this player.
- I will plan the upcoming day when the schedule is a bit more general for that day or if there are any specific tasks I want to prioritize, as I narrowed down concept art and prototype 1 to specifically what concept art I will do and what programming and game engine work should be done tomorrow.

CONTENTS – WEEK 4

ART

PROGRAMMING

PROJECT MANAGEMENT

ENVIRONMENT CONCEPT ART – DAY 22

- My first concept art was for the above ground environment, consisting of many elements from the environment mood board that have been put together into this concept, which I drew digitally with the use of photoshop and my graphics tablet.

PEER FEEDBACK – DAY 22

- Background environment concept art:

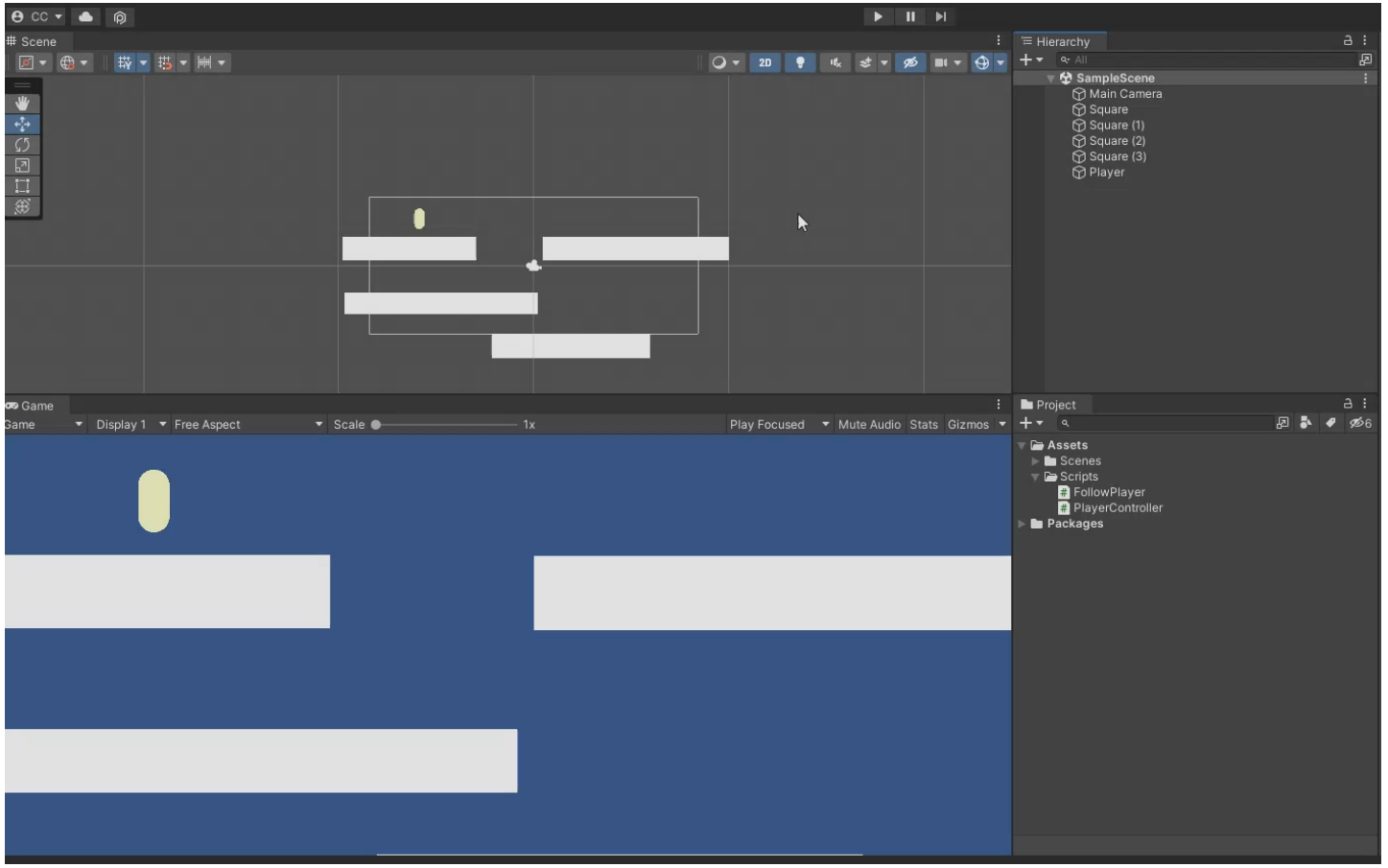
 The ocean is bland and looks like a void

 The light colours work well and fit in with this style of game

 The different elements of the environment make it interesting

PROGRAMMING – DAY 22

- Prototype 1 started with the implementation of the player controller script and the camera onto placeholder player (capsule) – the unity scene and game view is shown in action on the right with this video.



PROGRAMMING – DAY 22

- The player controller script currently just gets the horizontal axis for the input and transforms the position of the player right by this and by a speed variable over time each second.
- I made a new script called “follow player” to add to the camera so it will focus on the player, the camera is set to be at the position of the player plus an offset in the Y and Z axes – raising the camera up slightly on the Y so the levels below the player are not in view, and Infront of the player on the Z instead of being on the same Z as the player and looking out behind the game scene.

FollowPlayer.cs | PlayerController.cs

Miscellaneous Files | PlayerController

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class PlayerController : MonoBehaviour
6  {
7      public float horizontalInput;
8      public float speed = 10.0f;
9      // Start is called before the first frame update
10     void Start()
11     {
12
13     }
14
15     // Update is called once per frame
16     void Update()
17     {
18         horizontalInput = Input.GetAxis("Horizontal");
19         transform.Translate(Vector3.right * horizontalInput * Time.deltaTime * speed);
20     }
21 }
22
```

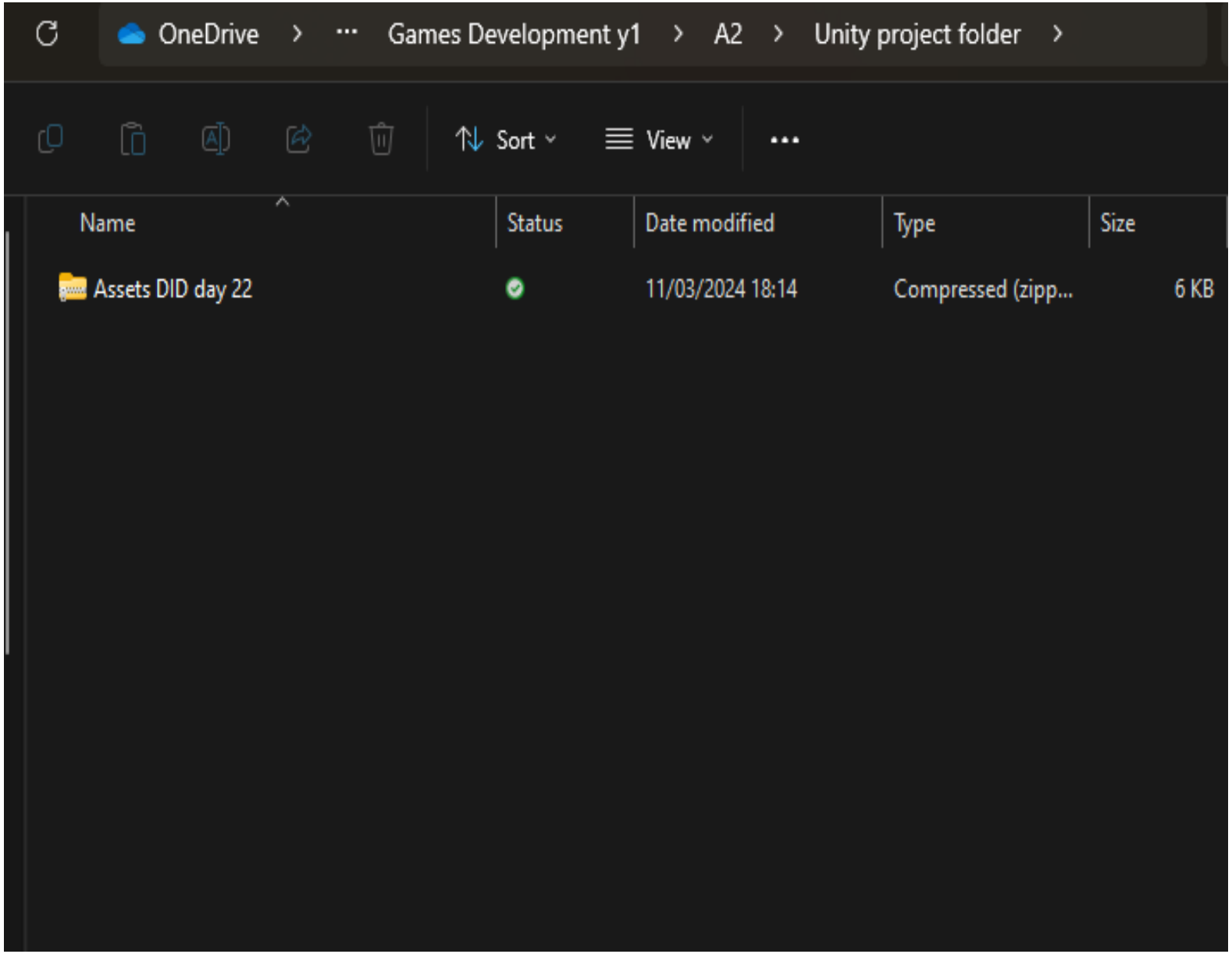
FollowPlayer.cs | PlayerController.cs

Miscellaneous Files | FollowPlayer

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class FollowPlayer : MonoBehaviour
6  {
7      public GameObject player;
8      private Vector3 offset = new Vector3(0, 2, -10);
9      // Start is called before the first frame update
10     void Start()
11     {
12
13     }
14
15     // Update is called once per frame
16     void LateUpdate()
17     {
18         // Offset camera infront of and above the player by adding it to the player's position + a new Vector3 so it is above and infront
19         transform.position = player.transform.position + offset;
20     }
21 }
22
```

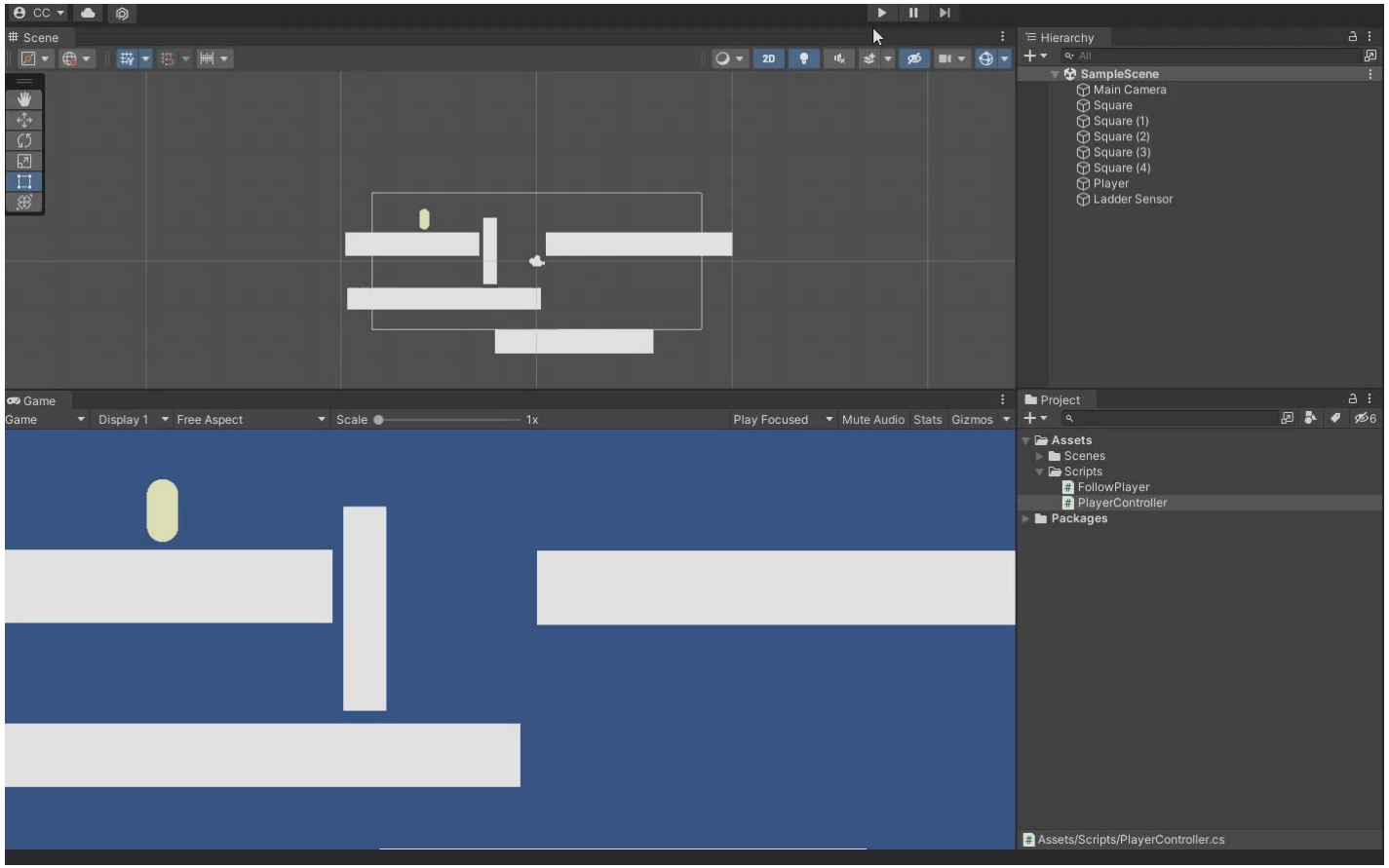
BACKUPS– DAY 22

- I incrementally backed up the assets for the game in the OneDrive, which I will do every time I make any change.
- At the end of each prototype, I will backup the full project in my C-Drive and to an external USB, so more major changes are saved in multiple locations.



PROGRAMMING – DAY 23

- I started the implementation of ladders with this block out, currently you can move horizontally, and the player is only able to move vertically once on the ladder – you can not leave the ladder yet, nor are there limits to how far you move vertically.



PROGRAMMING – DAY 23

- This was achieved by coding the previously planned game logic of an is on ladder variable, which dictates if you can move vertically or horizontally – depending on if you're on the ladder or not.
- I added a box collider to the top of the ladder with "is trigger" checked, so when you enter this trigger that is tagged with "ladder" the variable becomes true.

PlayerController.cs

Miscellaneous Files — PlayerController

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class PlayerController : MonoBehaviour
6  {
7      public float horizontalInput;
8      public float verticalInput;
9      public float speed = 10.0f;
10     private bool isOnLadder = false;
11     // Start is called before the first frame update
12     void Start()
13     {
14
15     }
16
17     // Update is called once per frame
18     void Update()
19     {
20         horizontalInput = Input.GetAxis("Horizontal");
21         verticalInput = Input.GetAxis("Vertical");
22         if(isOnLadder == false)
23         {
24             transform.Translate(Vector3.right * horizontalInput * Time.deltaTime * speed);
25         }
26
27         if(isOnLadder == true)
28         {
29             transform.Translate(Vector3.up * verticalInput * Time.deltaTime * speed);
30         }
31
32     }
33
34     // method for entering colliders
35     private void OnTriggerEnter2D(Collider2D other)
36     {
37         if(other.gameObject.CompareTag("Ladder"))
38         {
39             isOnLadder = true;
40             Debug.Log("ON ladderr");
41         }
```

CAVE CONCEPT ART – DAY 23

- I finished the day with the cave concept art, showing the basic dark stone in the background and the simple cave stone over the top that should be a tile map when made into the final asset.
- The level design and placement here is almost final, except for the fossil at the bottom – which I added to show what the larger fossils may look like – as a fossil of this size won't be this close to the cave entrance.

FOSSIL CONCEPT ART – DAY 24

- The final part of the environmental concept art is the fossils, where I drew the other two fossils out of the four main fossils I'll include in the game – the other two are on the previous cave concept art.
- On the left is the claw from the Therizinosaurus, and the other (the focus of this concept art) is the head of the Dunkleosteus – the references from my previously created mood board have been inserted below each piece of concept art on this slide.

PEER FEEDBACK – DAY 24

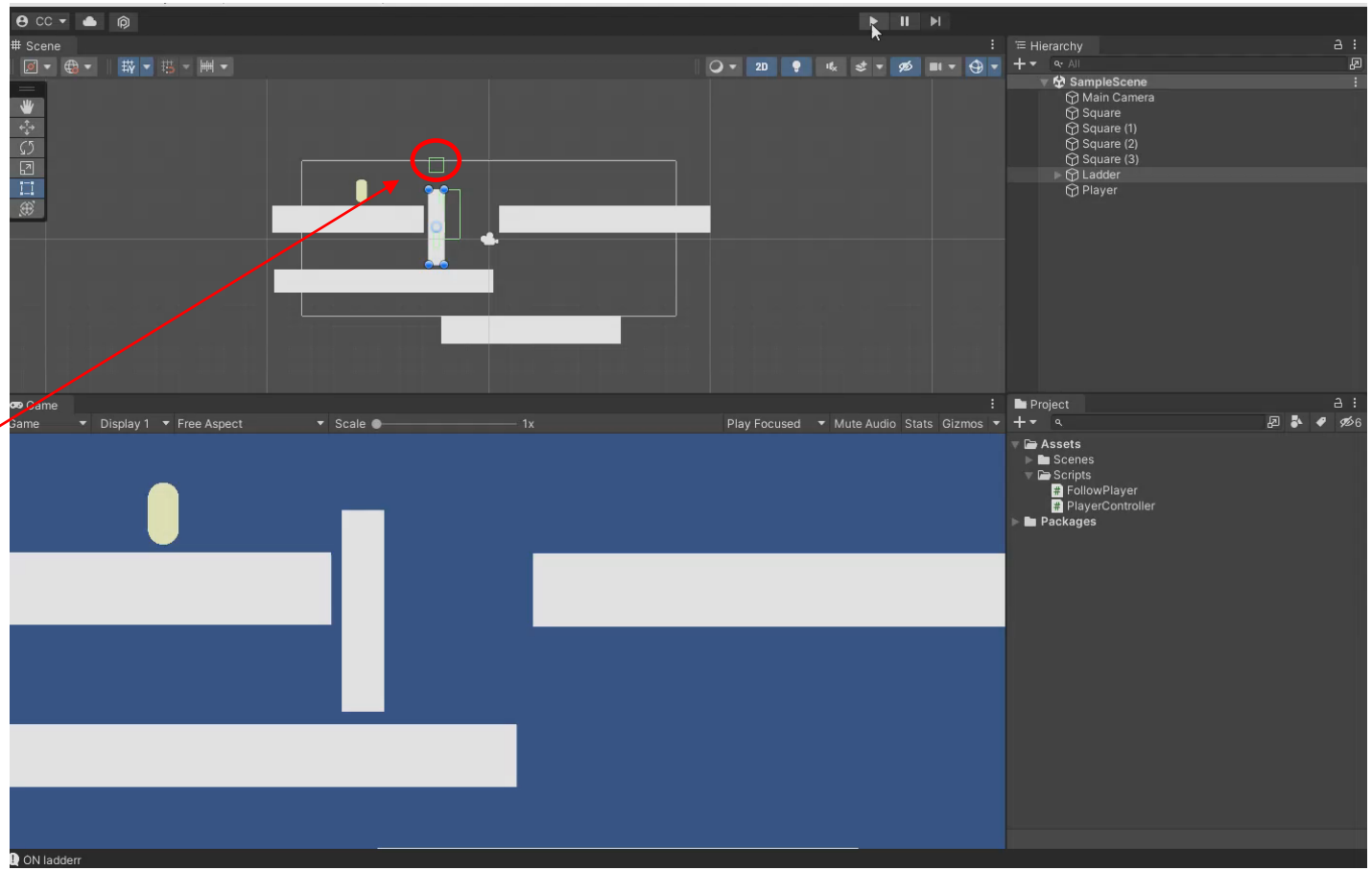
- Fossil concept art:


The fossils may be too detailed and contrast to the rest of the game's art


The concept is accurate to the real fossil

PROBLEM SOLVING – DAY 25

- I added in the ability to dismount the ladder by pressing the space key, and a collider above the ladder to limit how far the player can travel vertically – the result of these additions is shown on the right in this video.



PlayerController.cs

Miscellaneous Files | PlayerController

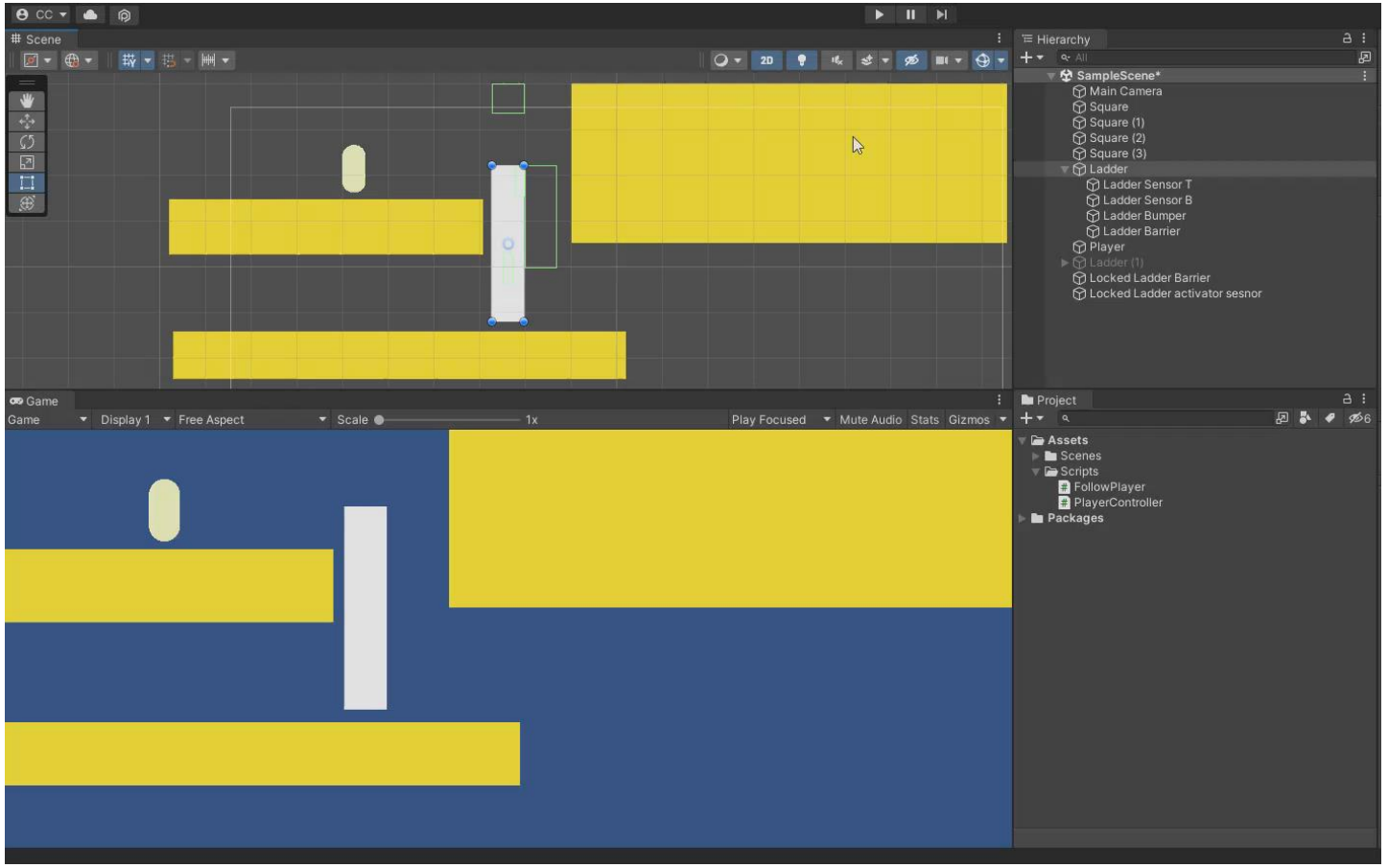
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerController : MonoBehaviour
{
    public float horizontalInput;
    public float verticalInput;
    public float speed = 10.0f;
    private bool isOnLadder = false;
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if(Input.GetKeyDown(KeyCode.Space))
        {
            isOnLadder = false;
        }
        horizontalInput = Input.GetAxis("Horizontal");
        verticalInput = Input.GetAxis("Vertical");
        if(isOnLadder == false)
        {
            transform.Translate(Vector3.right * horizontalInput * Time.deltaTime * speed);
        }

        if(isOnLadder == true)
        {
            transform.Translate(Vector3.up * verticalInput * Time.deltaTime * speed);
        }

    }

    // method for entering colliders
    private void OnTriggerEnter2D(Collider2D other)
    {
        if(other.gameObject.CompareTag("Ladder"))
        {
            isOnLadder = true;
            Debug.Log("ON ladderr");
        }
    }
}

```

PROBLEM SOLVING – DAY 25

- The only code added for the dismount to initially work was this if statement, setting the is on ladder variable to false upon being pressed – allowing horizontal movement again.

PROBLEM SOLVING – DAY 26

- I finished the ladder by organizing and naming all the colliders attached to it, and by adding a force to the player when they enter the bumper trigger – so the player is dismounted automatically.
- I added two more colliders via empty objects for the locked ladders, one of these stops you from moving over the edge where you haven't yet unlocked a ladder and the other is a trigger where you will be able to place the ladder from 0 – the colliders are in place, but the placing ladders feature still needs to be implemented.
- The player's walk speed has been halved as it was too fast, but the climb speed is the same as before.



```
43      }
44
45      // method for entering trigger colliders
46      private void OnTriggerEnter2D(Collider2D other)
47      {
48          if(other.gameObject.CompareTag("Ladder"))
49          {
50              isOnLadder = true;
51              Debug.Log("ON ladderr");
52          }
53          if (other.gameObject.CompareTag("Bumper"))
54          {
55              // makes the player jump off the ladder when they reach the top
56              playerRb.AddForce(Vector2.left * bumperForce, ForceMode2D.Impulse);
57              isOnLadder = false;
58          }
```

PROBLEM SOLVING – DAY 26

- To add an impulse force to the player, I got the player's rigid body component and added the impulse to this via a player Rb variable when the player enters the bumper's trigger – is on ladder is also set to false during this process.

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class PlayerController : MonoBehaviour
6   {
7       public float horizontalInput;
8       public float verticalInput;
9       public float speed = 5f;
10      public float climbSpeed = 10;
11      private bool isOnLadder = false;
12      public Rigidbody2D playerRb;
13      private float bumperForce = 0.0004f;
14      // Start is called before the first frame update
15      void Start()
16      {
17          playerRb = GetComponent<Rigidbody2D>();
18      }
19
20      // Update is called once per frame
21      void Update()
22      {
23          if(Input.GetKeyDown(KeyCode.Space))
24          {
25              isOnLadder = false;
26          }
27          // assigns the horizontal axis input (A and D keys) to the horizontal axis variable
28          horizontalInput = Input.GetAxis("Horizontal");
29          // assigns the vertical axis input (W and S keys) to the vertical axis variable
30          verticalInput = Input.GetAxis("Vertical");
31          if(isOnLadder == false)
32          {
33              // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
34              transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
35          }
36
37          if(isOnLadder == true)
38          {
39              // move vertically (y axis) with the W and S keys over time each second at a speed of 10
40              transform.Translate(Vector2.up * verticalInput * Time.deltaTime * climbSpeed);
41          }
42
43      }
```

CHARACTER CONCEPT ART – DAY 27

- For the character concept art, I combined different elements of characters from my mood board to create my own character – mainly consisting of Alan Grant's hat, Mr. B's face, Nigel Marven and Steve Irwin's clothing, and a head light attached to the hat.

CHARACTER CONCEPT ART – DAY 27

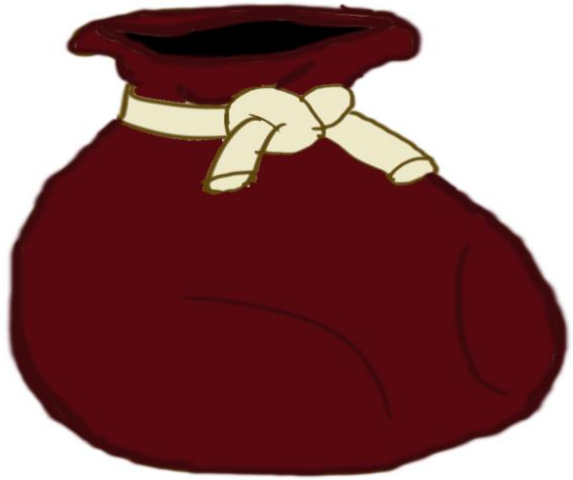
THE CONCEPT WENT THROUGH MANY EXPERIMENTATIONS BEFORE REACHING ITS FINAL POINT ON THE PREVIOUS SLIDE, THIS SLIDE SHOWS SOME OTHER COLORS AND CONCEPTS I TRIED OUT THAT I CHANGED OR REMOVED LATER – INCLUDING THE JACKET, A WHITE CIRCLE TO REPRESENT THE HEADLIGHT, A DARKER SHIRT COLOR, AND A PINKER SKIN TONE.

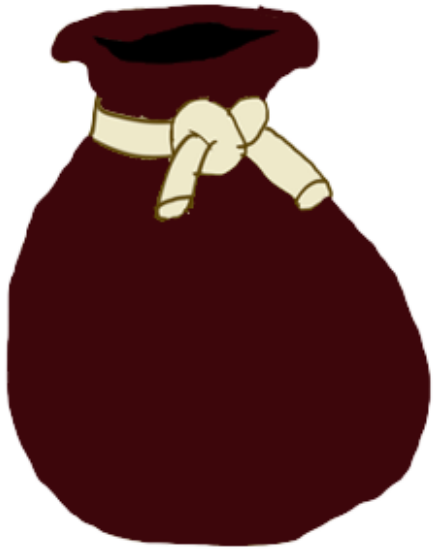
PEER FEEDBACK – DAY 27

- Character concept art:
  - ✓ The character has a simple design with identifiable features, which is good for pixel art

EQUIPMENT CONCEPT ART – DAY 28

- I briefly worked on a bit of equipment concept art – completing this concept for the fossil bag, a small pouch that will contain the fossil samples (like fossilized teeth or claws)
- The version at the top is the final iteration, underneath are two other colors I tried out before coming to the final color and adding the creases with a darker color.



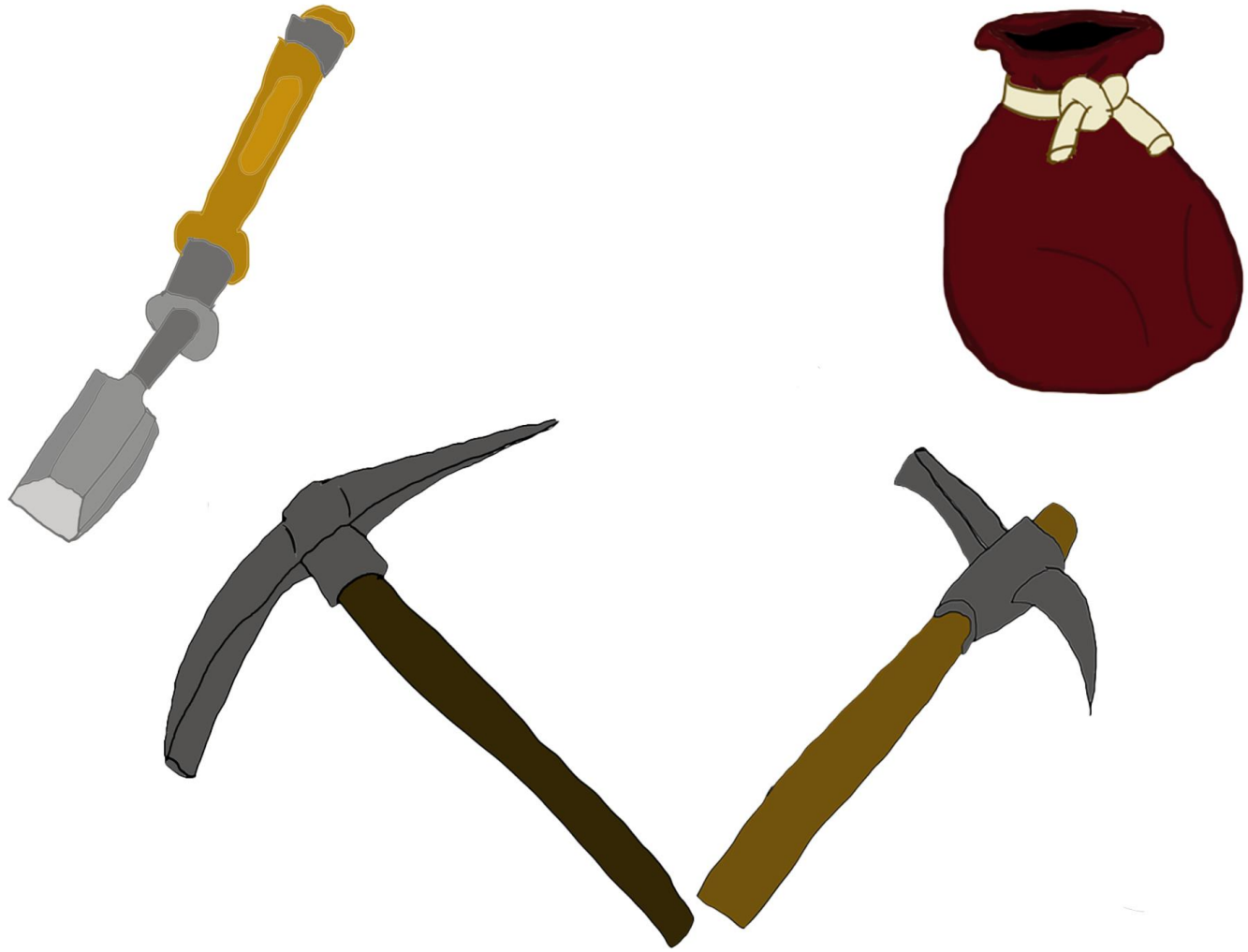


CONTENTS – WEEK 5

ART MILESTONE PROJECT MANAGEMENT PROGRAMMING

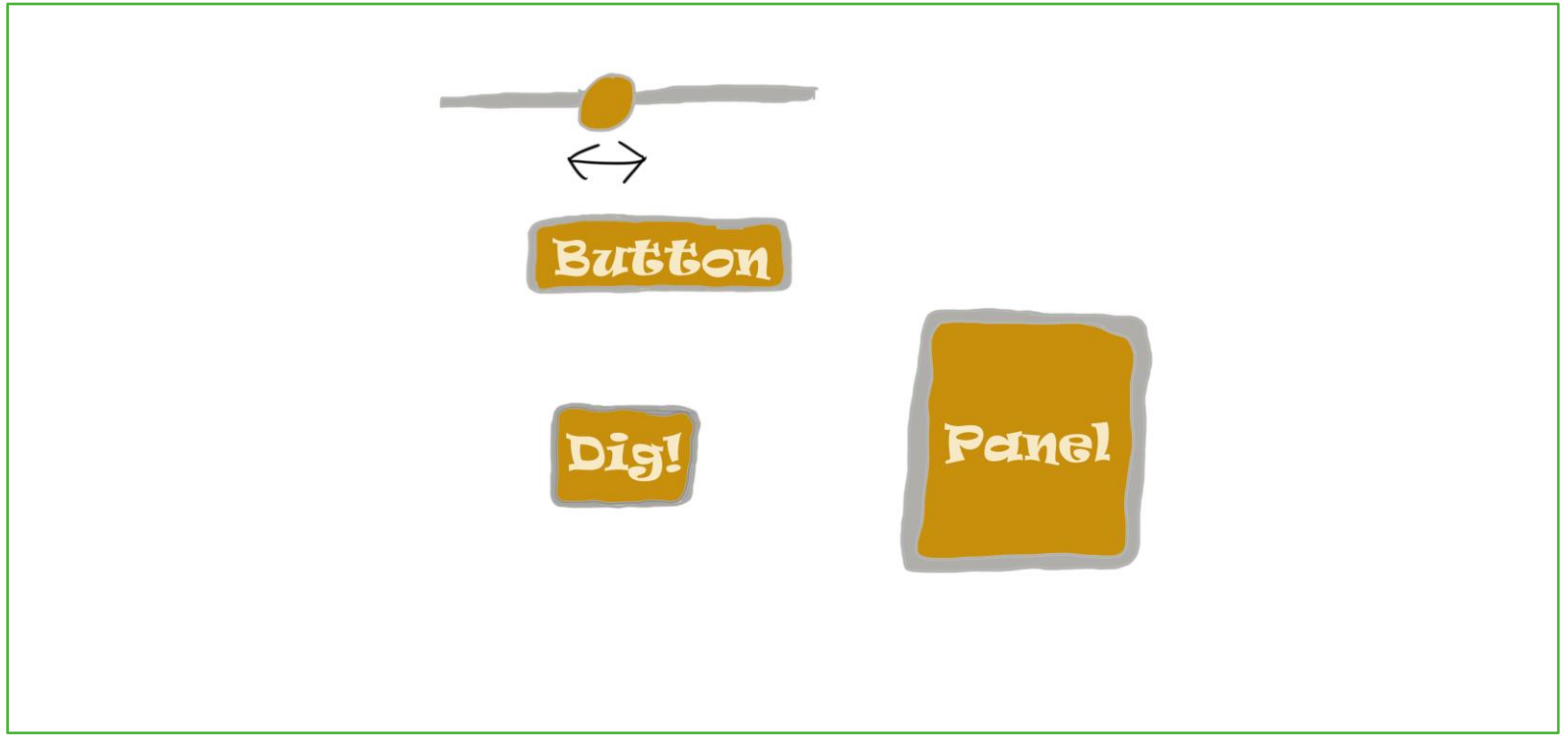
EQUIPMENT CONCEPT ART – DAY 29

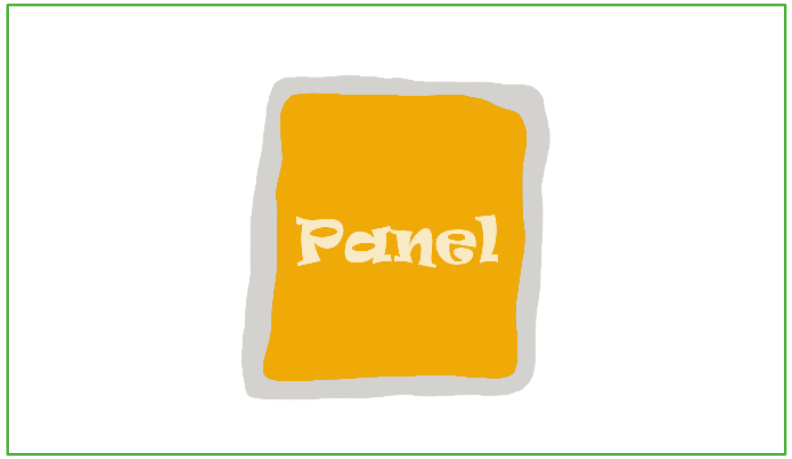
- To finish the equipment concept art, I drew the tools that will be used in the game – this being the chisel and two variations of the pickaxe – the pickaxe on the left is the final version and the one on the right was the first attempt, as I experimented with different colors for the handle of the pickaxe and with different shapes for the metal.

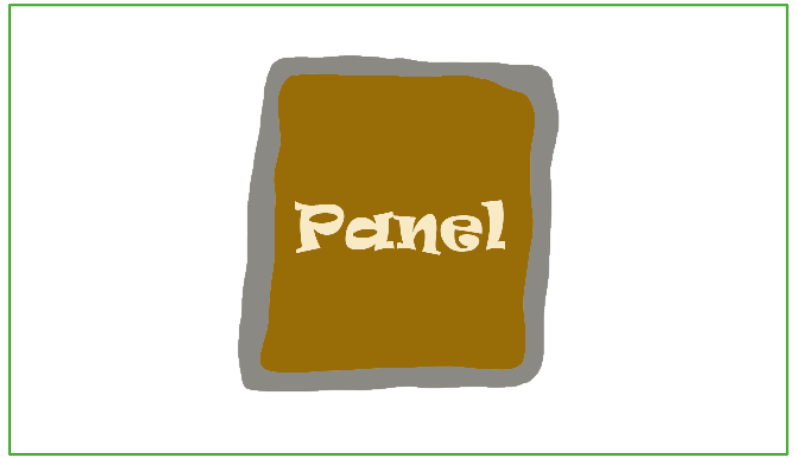


UI CONCEPT ART – DAY 29

- The concept art for the UI mainly involved the button and panel designs, each button/panel will have the same design but will contain different text, along with the audio slider design – which may be implemented depending on if I include any audio or music in this game, as it is not required in any capacity for this project and will only be added if I have spare development time after implementing everything else.
- These colors have been chosen as they are the same colors of the cave and fossils in the previous cave concept art – although I experimented with some other colors for the panel and its border, which I didn't end up using, shown underneath the main collection of UI concept art.

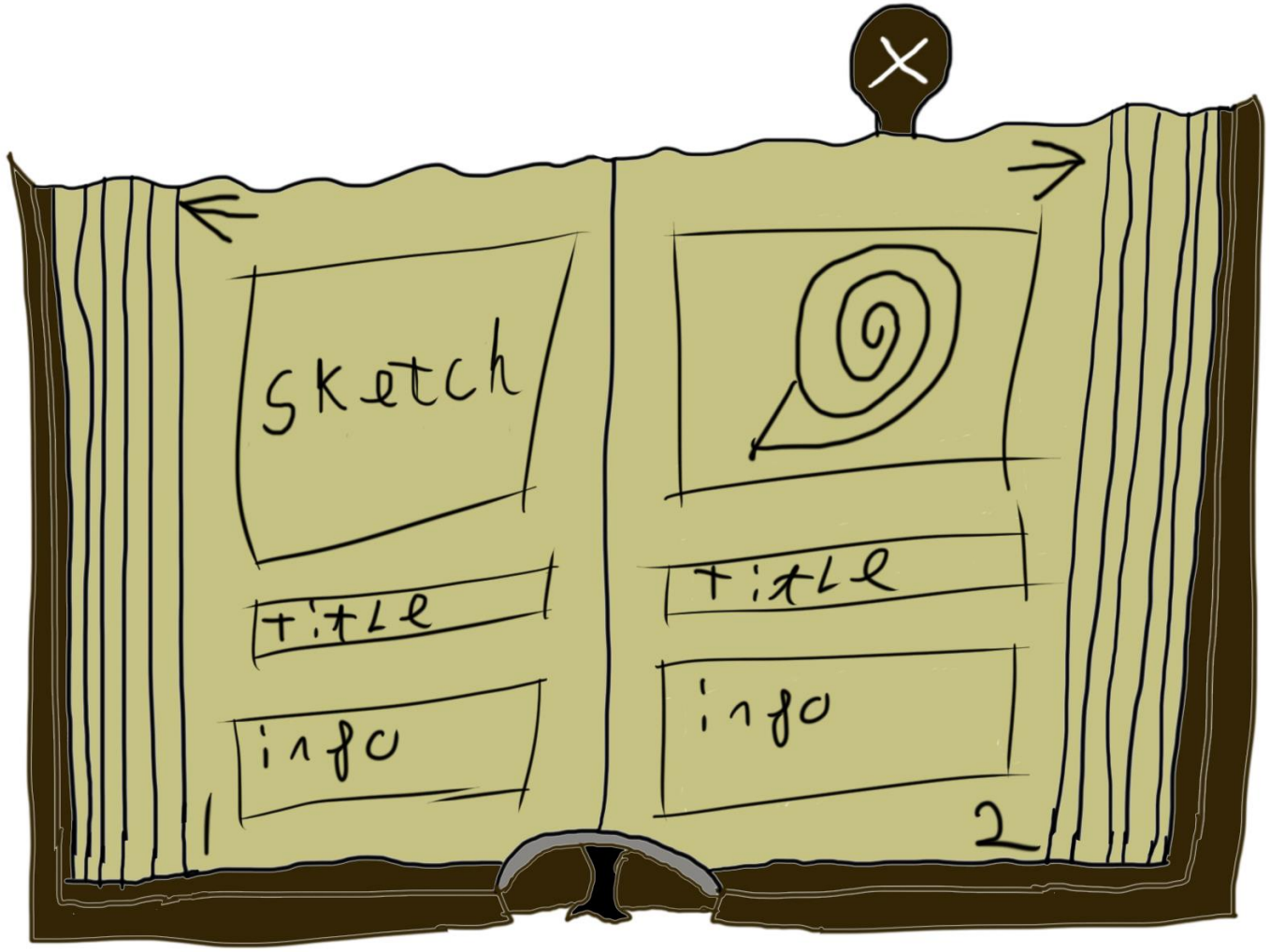






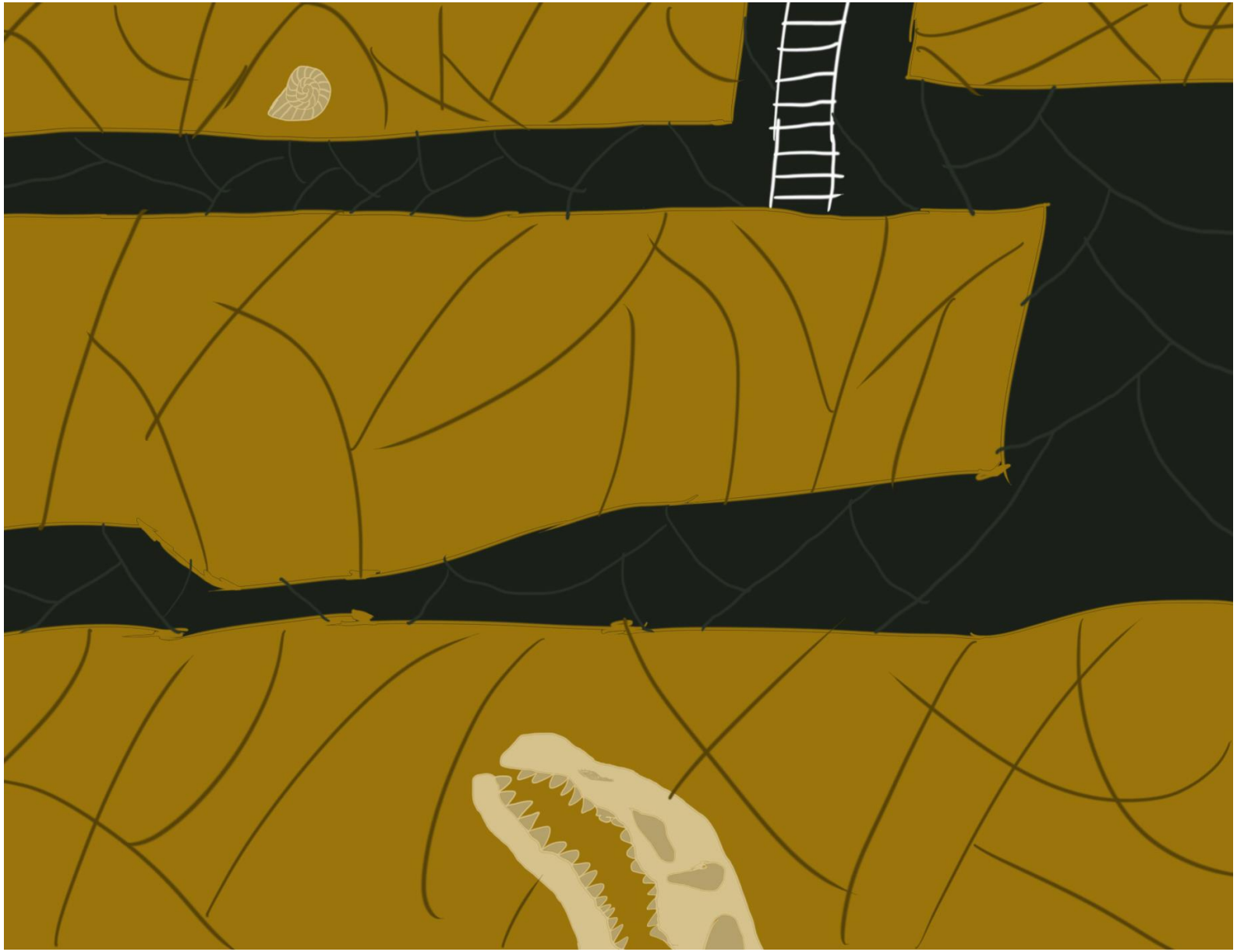
UI CONCEPT ART – DAY 29

- Concept art as a phase came to an end with the completion of this journal concept – the log the player will keep of their progress and discoveries, showing the sketch of the fossils they find as well as its information if they manage to bring the fossil back to the dig site.



CAVE CONCEPT ART UPDATE – DAY 29

- I briefly returned to the cave concept art to add depth to the background rock, adding lines to the solid color to make it look more natural.



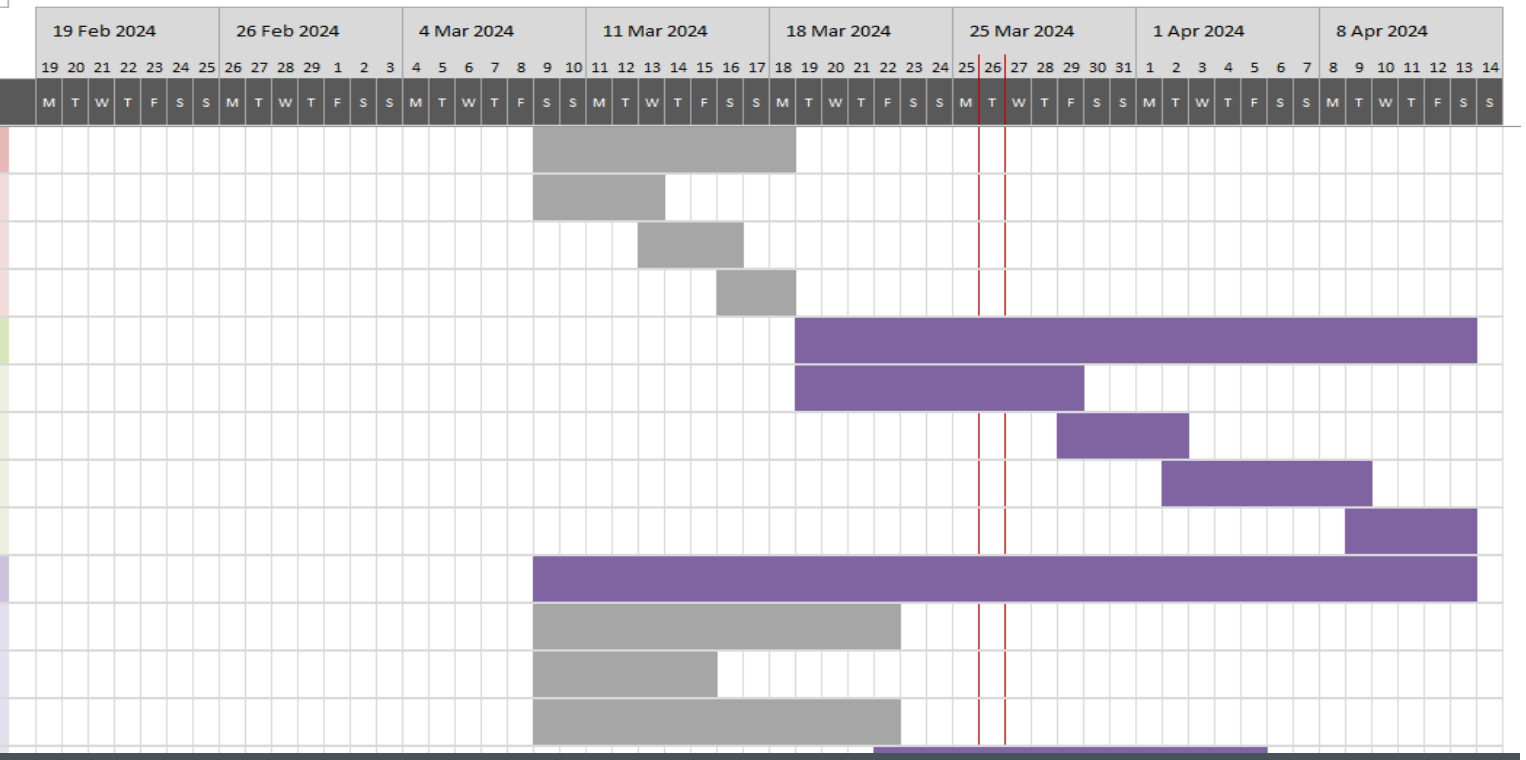
Dig it, Doug!

FCOT studios
Charlie Coll

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: Tue, 20/02/2024
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Concept Art		100%	9/3/24	18/3/24
Environment concept art		100%	9/3/24	13/3/24
Character concept art		100%	13/3/24	16/3/24
UI concept art		100%	16/3/24	18/3/24
Final Sprite and asset development			19/3/24	18/4/24
Environment and tile maps			19/3/24	29/3/24
Fossils			29/3/24	2/4/24
Character and equipment			2/4/24	9/4/24
UI			9/4/24	18/4/24
Programming and game engine work			9/3/24	26/4/24
Prototype 1		100%	9/3/24	22/3/24
Camera implementation		100%	9/3/24	15/3/24
Player controller script		100%	9/3/24	22/3/24

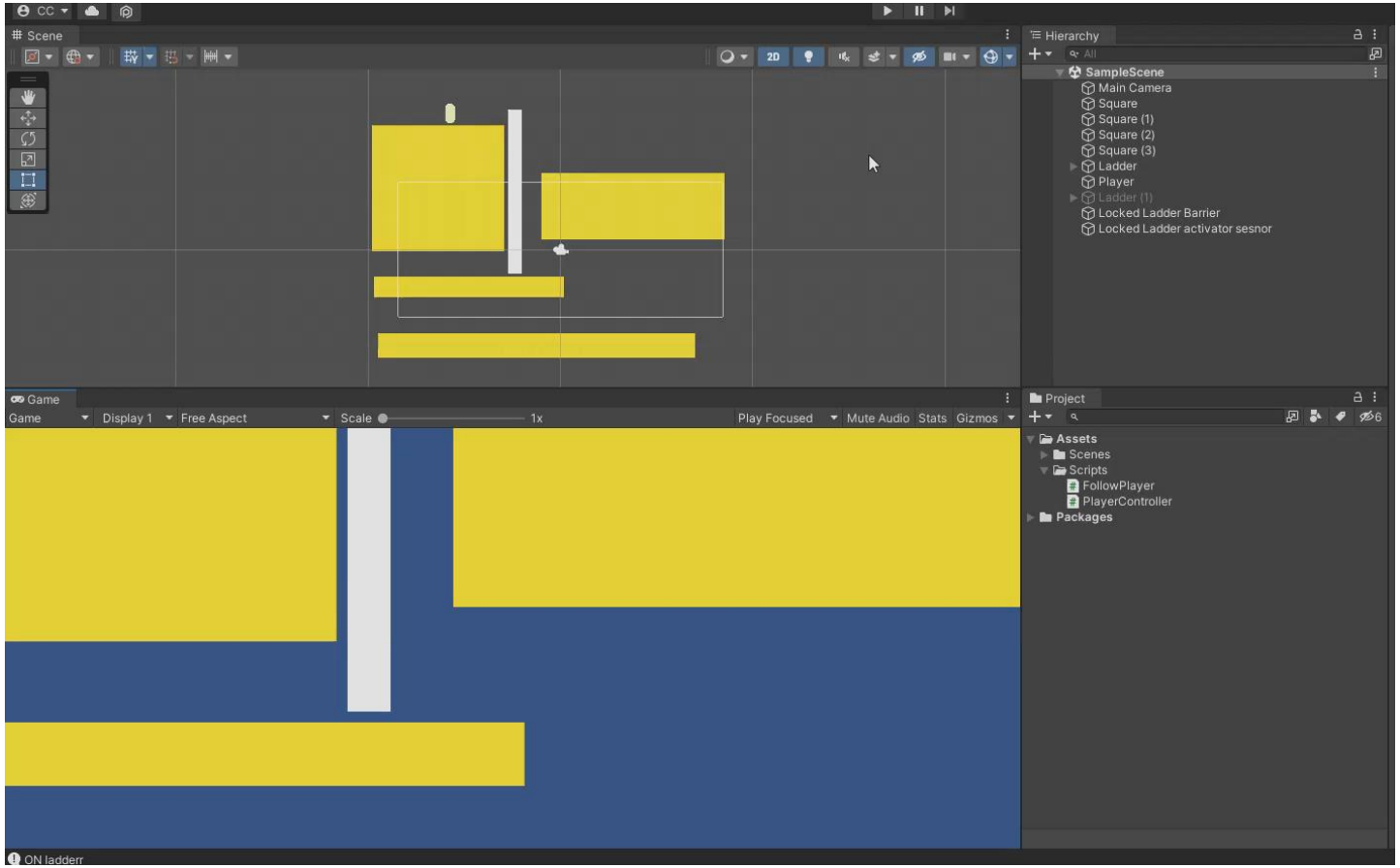


MILESTONE – DAY 29

- With that, I marked the concept art phase as complete on the Gantt – shown above – and on click up's kanban board (this screenshot was taken 8 days afterwards, hence prototype 1 being marked complete as well)

PROGRAMMING – DAY 30

- I finally finished the core implementation of the ladders with the addition of placing ladders down at certain points, there is a barrier present when you have not placed the ladder that is deactivated when you place it.
- I adjusted the scale of the platform you start on, as the final level design will involve the player starting above ground and descending further down into the cave.



troller.cs ⊹ X

neous Files ▾ PlayerController

```
    public float horizontalInput;
    public float verticalInput;
    public float speed = 5f;
    public float climbSpeed = 10;
    private bool isOnLadder = false;
    public Rigidbody2D playerRb;
    private float bumperForce = 0.0004f;
    public GameObject ladder;
    public BoxCollider2D ladderBarrier;
    private bool isInActivator = false;
    private float laddersOwned = 1;
    // Start is called before the first frame update
    void Start()
    {
        playerRb = GetComponent<Rigidbody2D>();
    }

    // Update is called once per frame
    void Update()
    {
        if(Input.GetKeyDown(KeyCode.Space))
        {
            isOnLadder = false;
        }
        // assigns the horizontal axis input (A and D keys) to the horizontal axis variable
        horizontalInput = Input.GetAxis("Horizontal");
        // assigns the vertical axis input (W and S keys) to the vertical axis variable
        verticalInput = Input.GetAxis("Vertical");
        if(isOnLadder == false)
        {
            // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
            transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
        }

        if(isOnLadder == true)
        {
            // move vertically (y axis) with the W and S keys over time each second at a speed of 10
            transform.Translate(Vector2.up * verticalInput * Time.deltaTime * climbSpeed);
        }

        // if the player is near the unlocked ladder, presses E, and has at least 1 ladder
        if(isInActivator == true && Input.GetKeyDown(KeyCode.E) && laddersOwned > 0)
        {
            // places ladder
            ladder.gameObject.SetActive(true);
            // removes locked ladder barrier
            ladderBarrier.gameObject.SetActive(false);
        }

    }
```

Miscellaneous Files ▾ PlayerController

```
    // method for entering trigger colliders
    private void OnTriggerEnter2D(Collider2D other)
    {
        if(other.gameObject.CompareTag("Ladder"))
        {
            isOnLadder = true;
            Debug.Log("ON ladderr");
        }
        if (other.gameObject.CompareTag("Bumper"))
        {
            // makes the player jump off the ladder when they reach the top
            playerRb.AddForce(Vector2.left * bumperForce, ForceMode2D.Impulse);
            isOnLadder = false;
        }
        if (other.gameObject.CompareTag("Activator"))
        {
            isInActivator = true;
        }
    }

    private void OnTriggerExit2D(Collider2D other)
    {
        if (other.gameObject.CompareTag("Activator"))
        {
            isInActivator = false;
            Debug.Log("flip");
        }
```

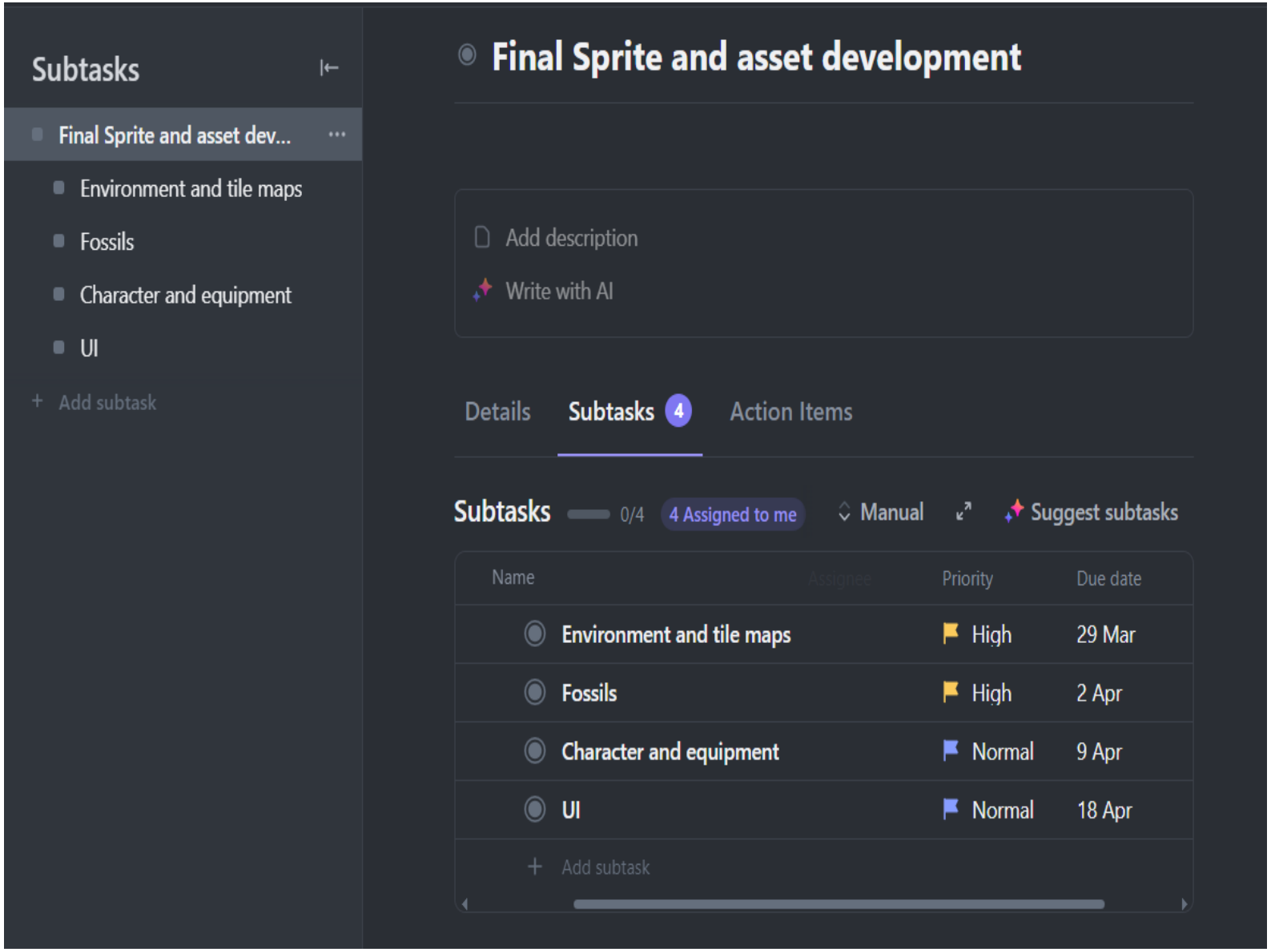
Checks if player is near the locked ladder

PROGRAMMING – DAY 30

THE LOCKED LADDER GAME OBJECT IS INACTIVE FROM THE START AND BECOMES ACTIVE WHEN THE PLAYER PRESSES “E” WHILE THEY ARE NEAR THE LOCKED LADDER – VIA AN IF STATEMENT THAT USES A BOOLEAN GOVERNED BY ON TRIGGER ENTER AND EXIT METHODS, THE INPUT FROM THE “E” KEY, AND A FLOAT THAT REPRESENTS HOW MANY LADDERS THE PLAYER OWNS.

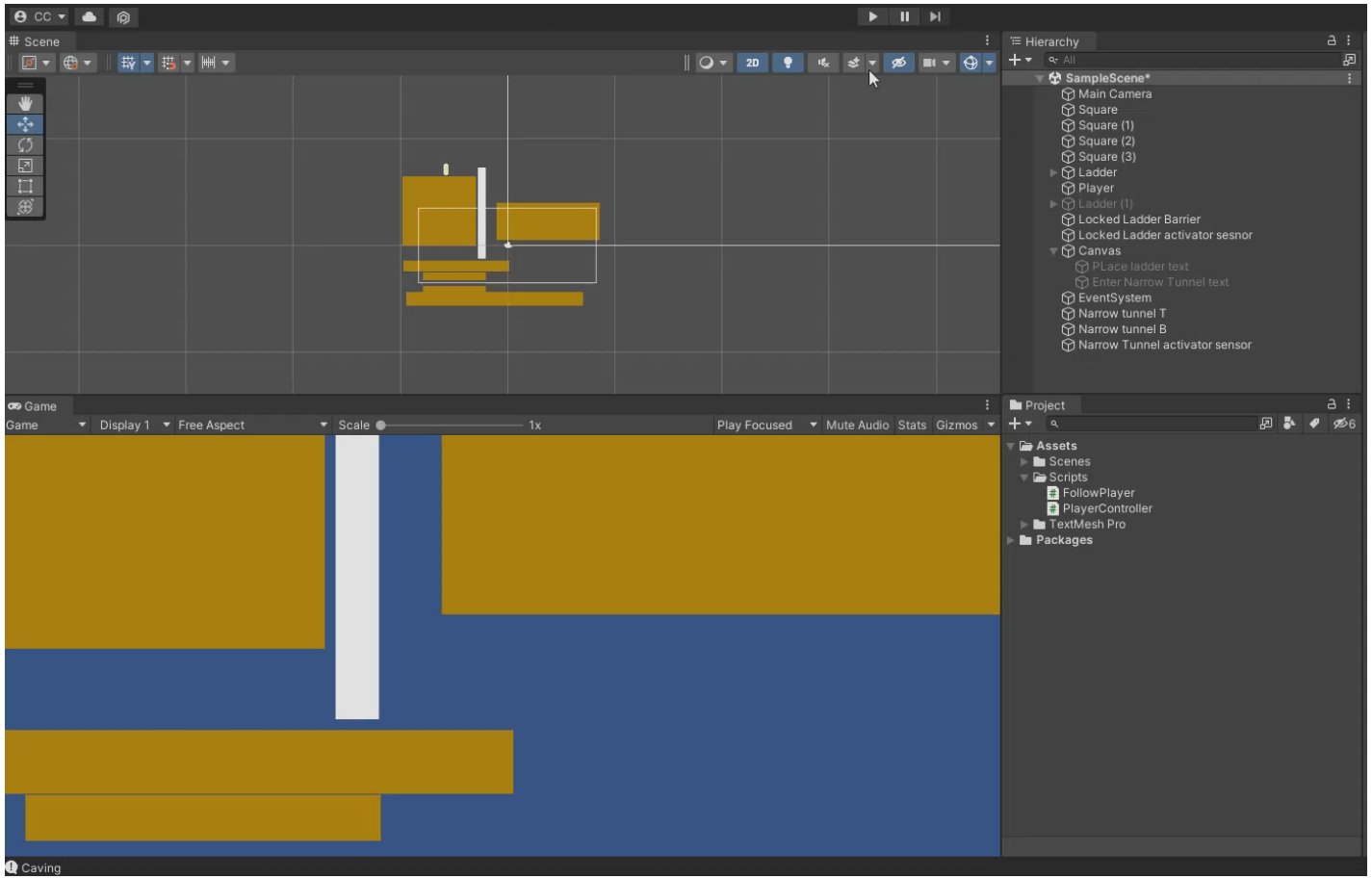
PROJECT MANAGEMENT – DAY 30

- I finalized the subtasks for the upcoming sprite and asset development phase on the click up kanban board (shown on this slide) and the Gantt chart, distributing them over the month-long period, so I can get started with weekly goals as soon as the time comes.



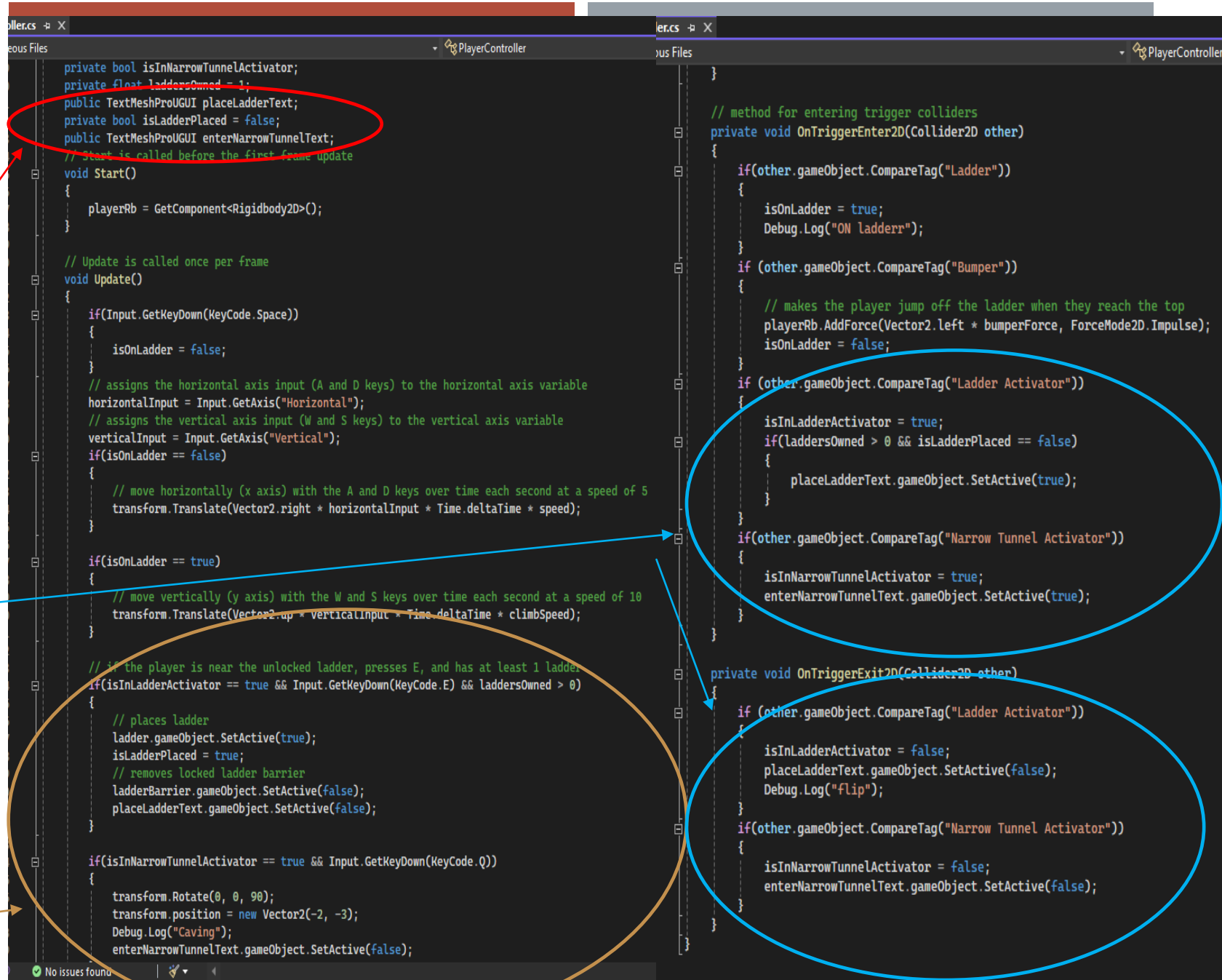
PROGRAMMING – DAY 33

- The completed prototype 1 now contains the player controller script with all its dedicated elements, some having basic UI prompts, like placing ladders and entering the narrow tunnels.



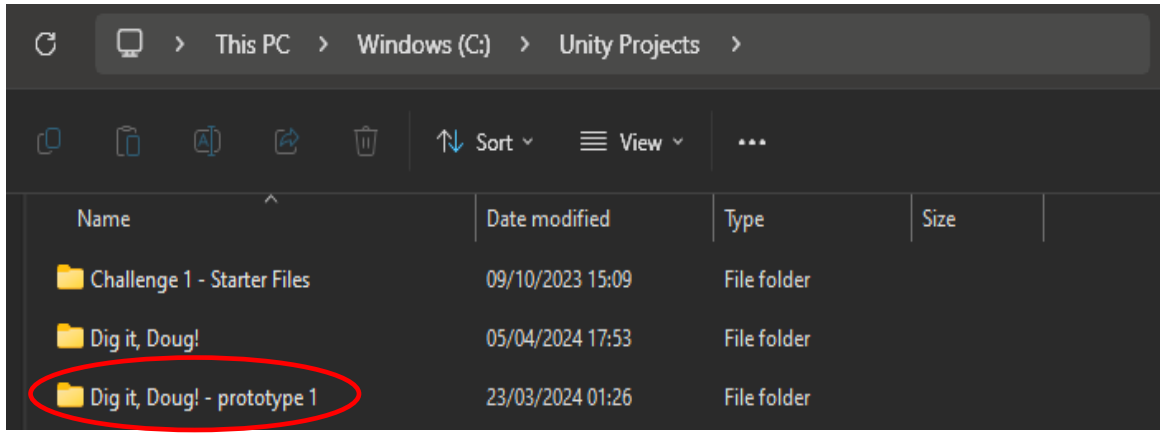
PROGRAMMING – DAY 33

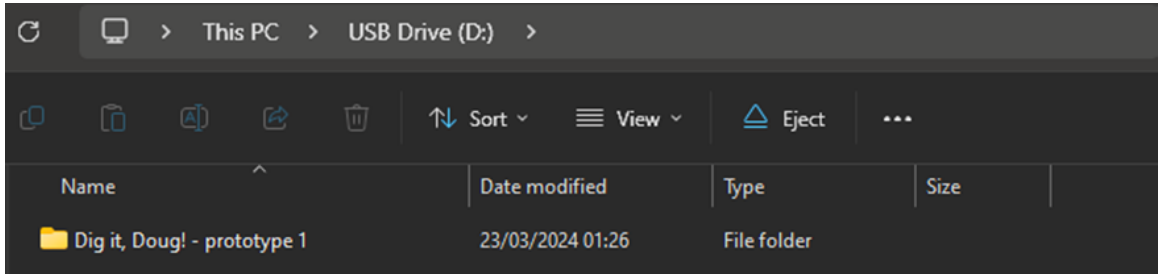
- I got the UI via public variables that use the unity UI and text mesh pro libraries, which have been anchored and customized in the inspector window of unity, to then set active and inactive when the player approaches the event where they need instruction – this works with the on-trigger methods and a variable that checks if you have already placed the ladder or not.
- The player enters the narrow tunnel once they've pressed "Q" by being rotated 90 degrees on the Z axis and being moved to the start point of the tunnel.



```
private bool isInNarrowTunnelActivator;
private float laddersOwned = 1;
public TextMeshProUGUI placeLadderText;
private bool isLadderPlaced = false;
public TextMeshProUGUI enterNarrowTunnelText;
// Start is called before the first frame update
void Start()
{
    playerRb = GetComponent<Rigidbody2D>();
}

// Update is called once per frame
void Update()
{
    if(Input.GetKeyDown(KeyCode.Space))
    {
        isOnLadder = false;
    }
    // assigns the horizontal axis input (A and D keys) to the horizontal axis variable
    horizontalInput = Input.GetAxis("Horizontal");
    // assigns the vertical axis input (W and S keys) to the vertical axis variable
    verticalInput = Input.GetAxis("Vertical");
    if(isOnLadder == false)
    {
        // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
        transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
    }

    if(isOnLadder == true)
    {
        // move vertically (y axis) with the W and S keys over time each second at a speed of 10
        transform.Translate(Vector2.up * verticalInput * Time.deltaTime * climbSpeed);
    }

    // if the player is near the unlocked ladder, presses E, and has at least 1 ladder
    if(isInLadderActivator == true && Input.GetKeyDown(KeyCode.E) && laddersOwned > 0)
    {
        // places ladder
        ladder.gameObject.SetActive(true);
        isLadderPlaced = true;
        // removes locked ladder barrier
        ladderBarrier.gameObject.SetActive(false);
        placeLadderText.gameObject.SetActive(false);
    }

    if(isInNarrowTunnelActivator == true && Input.GetKeyDown(KeyCode.Q))
    {
        transform.Rotate(0, 0, 90);
        transform.position = new Vector2(-2, -3);
        Debug.Log("Caving");
        enterNarrowTunnelText.gameObject.SetActive(false);
```

```
}

// method for entering trigger colliders
private void OnTriggerEnter2D(Collider2D other)
{
    if(other.gameObject.CompareTag("Ladder"))
    {
        isOnLadder = true;
        Debug.Log("ON ladderr");
    }
    if (other.gameObject.CompareTag("Bumper"))
    {
        // makes the player jump off the ladder when they reach the top
        playerRb.AddForce(Vector2.left * bumperForce, ForceMode2D.Impulse);
        isOnLadder = false;
    }
    if (other.gameObject.CompareTag("Ladder Activator"))
    {
        isInLadderActivator = true;
        if(laddersOwned > 0 && isLadderPlaced == false)
        {
            placeLadderText.gameObject.SetActive(true);
        }
    }
    if(other.gameObject.CompareTag("Narrow Tunnel Activator"))
    {
        isInNarrowTunnelActivator = true;
        enterNarrowTunnelText.gameObject.SetActive(true);
    }
}

private void OnTriggerExit2D(Collider2D other)
{
    if (other.gameObject.CompareTag("Ladder Activator"))
    {
        isInLadderActivator = false;
        placeLadderText.gameObject.SetActive(false);
        Debug.Log("flip");
    }
    if(other.gameObject.CompareTag("Narrow Tunnel Activator"))
    {
        isInNarrowTunnelActivator = false;
        enterNarrowTunnelText.gameObject.SetActive(false);
    }
}
```





BACKUPS– DAY 33

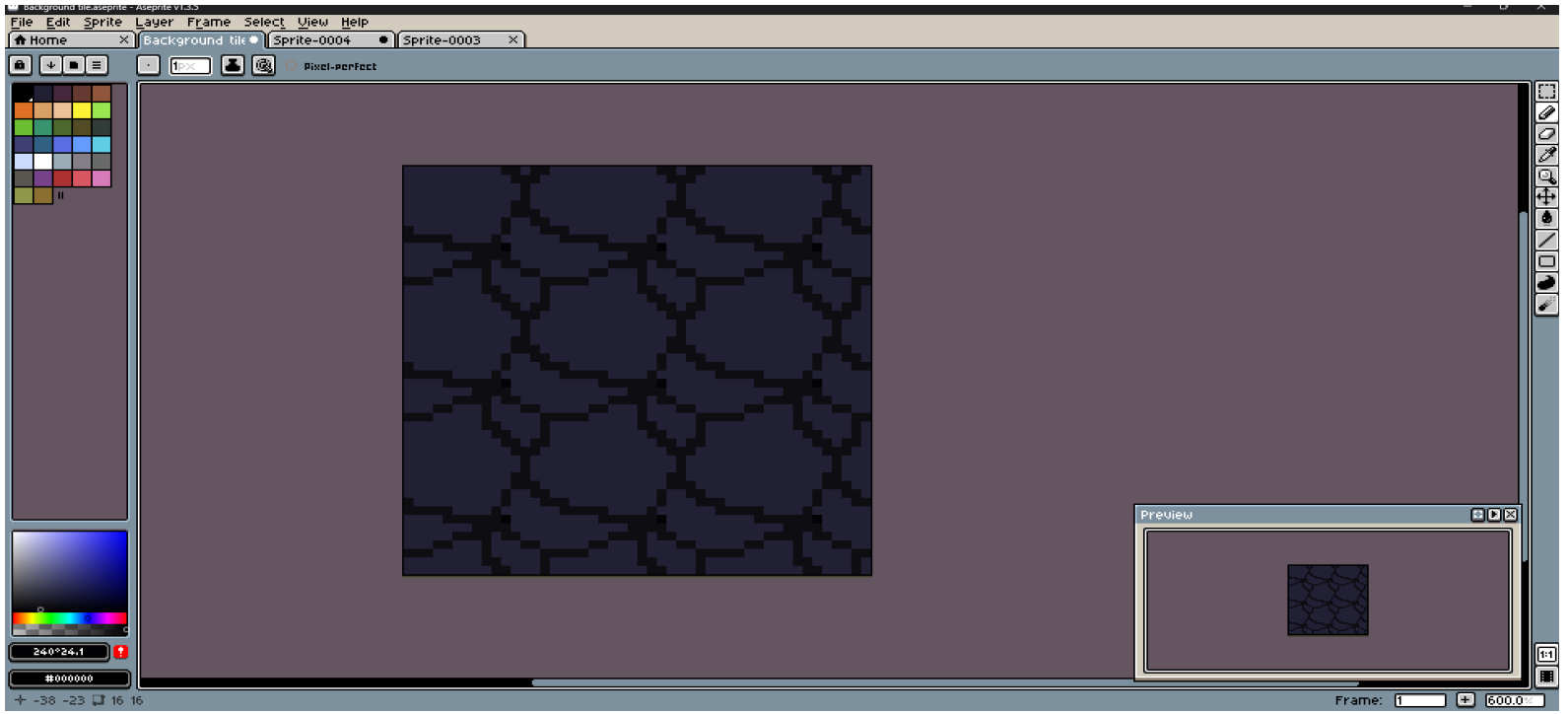
I FULLY BACKED UP THE COMPLETED PROTOTYPE 1 ONTO MY C-DRIVE AND MY USB DRIVE.

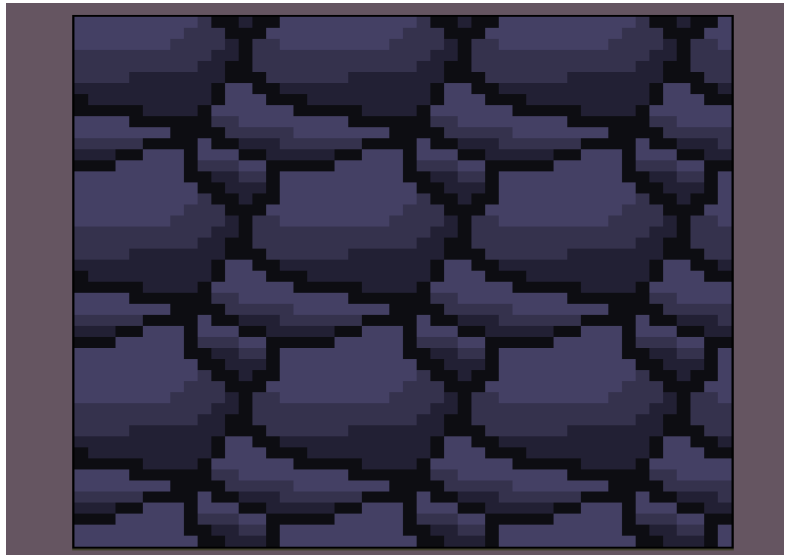
CONTENTS – WEEK 6

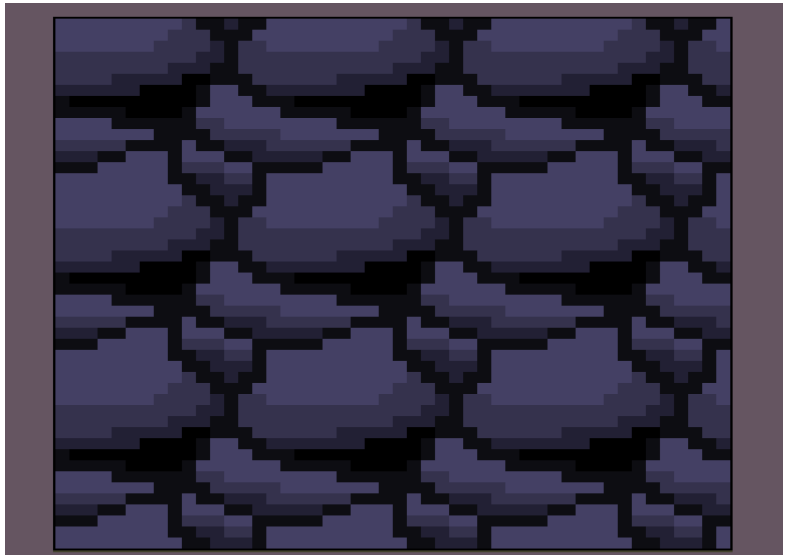
ART FEEDBACK PROGRAMMING PROJECT MANAGEMENT

BACKGROUND TILE ART – DAY 36

- The first final form of pixel art I worked on was this tile for the cave background, using the Aseprite program for this – which has a tiled mode that mirrors the pattern across the X and Y axes multiple times to create a tile.
- The image at the top is my first iteration involving outlines, then below on the left is the result of some shading – on the right is the 3rd version where I added a dark shadow at the bottom of the larger rock, that I ended up removing – which makes the final edition of the tile.

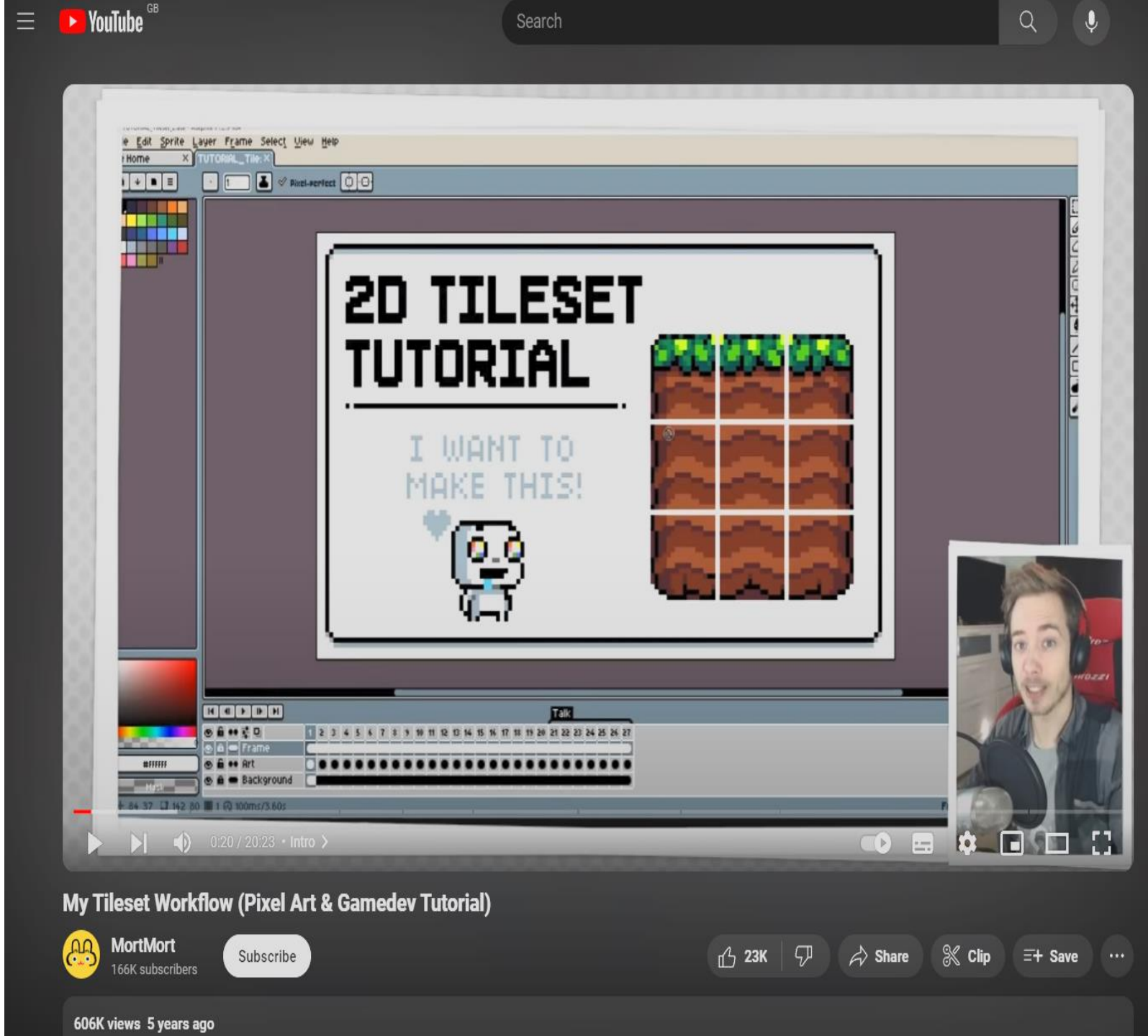






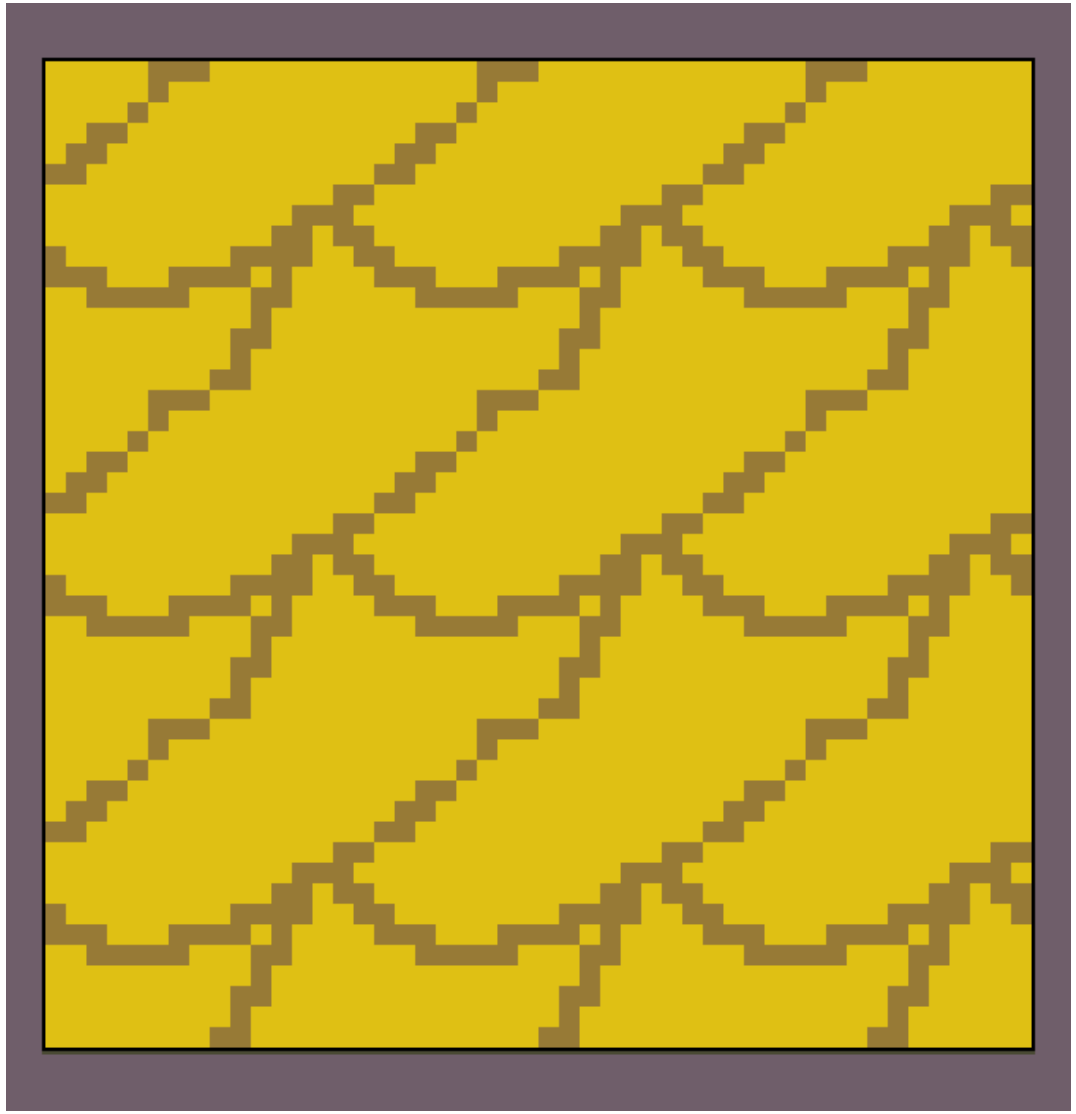
TUTORIAL – DAY 36

- Credit is due to Mort Mort's YouTube tutorial, as watching this taught me the process and tools/methods that I used when making my own tiles and tile sets in Aseprite.
- I'm using and learning this program for the first time while making the art for this project, but it is a risk worth taking as this app is dedicated to the creation of pixel art and it has been recommended to me on multiple occasions – I wouldn't know where to start making pixel art otherwise!



CAVE TILE ART – DAY 36

- I started creating the first iteration of the main cave tile, after finishing the background tile, shown to the right.



PEER FEEDBACK – DAY 37

- Prototype 1:

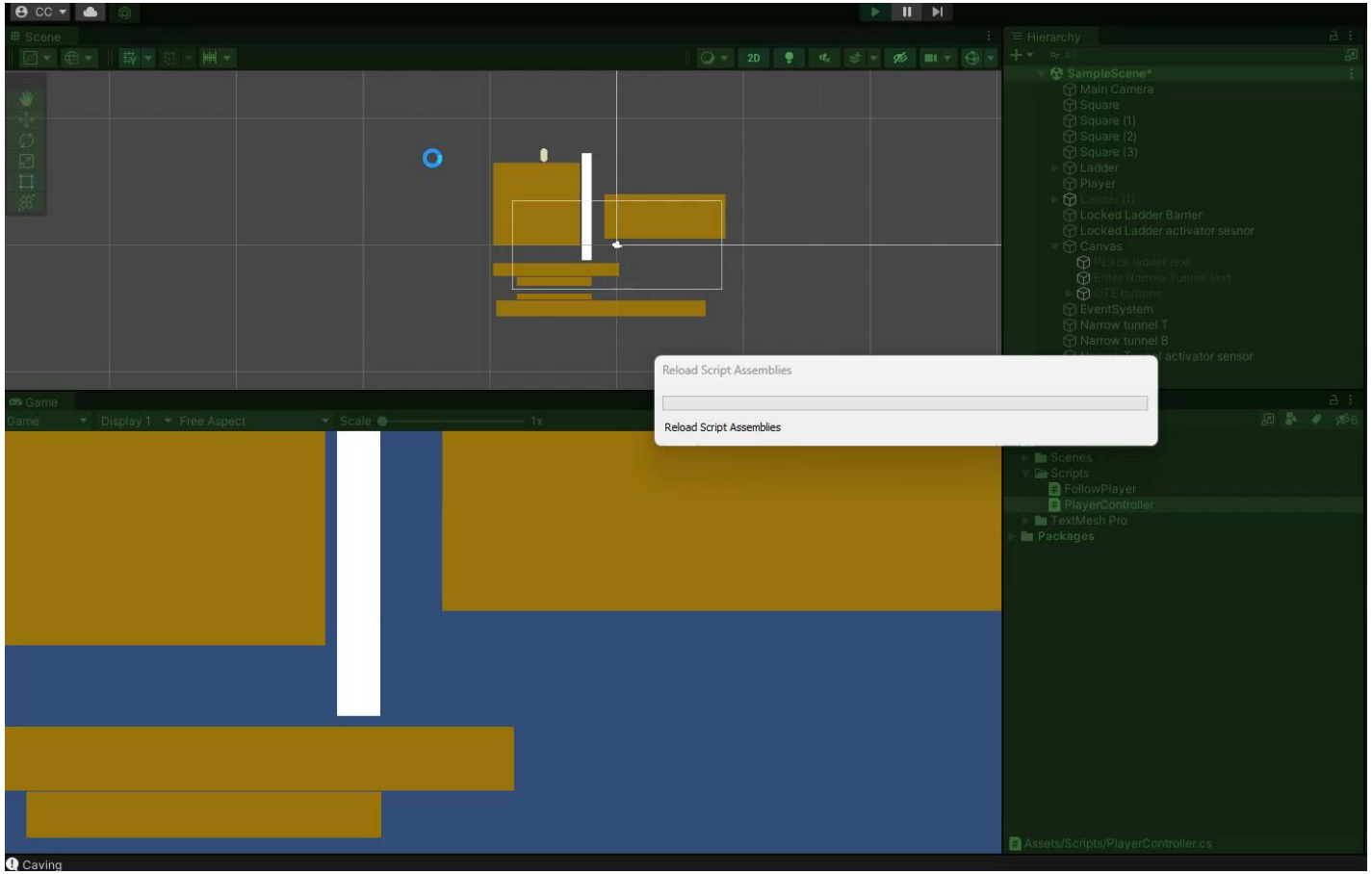

Gravity shouldn’t be on the ladders; you should only move down the ladder when you press “S”


The UI popping up is useful


The mechanics are interesting

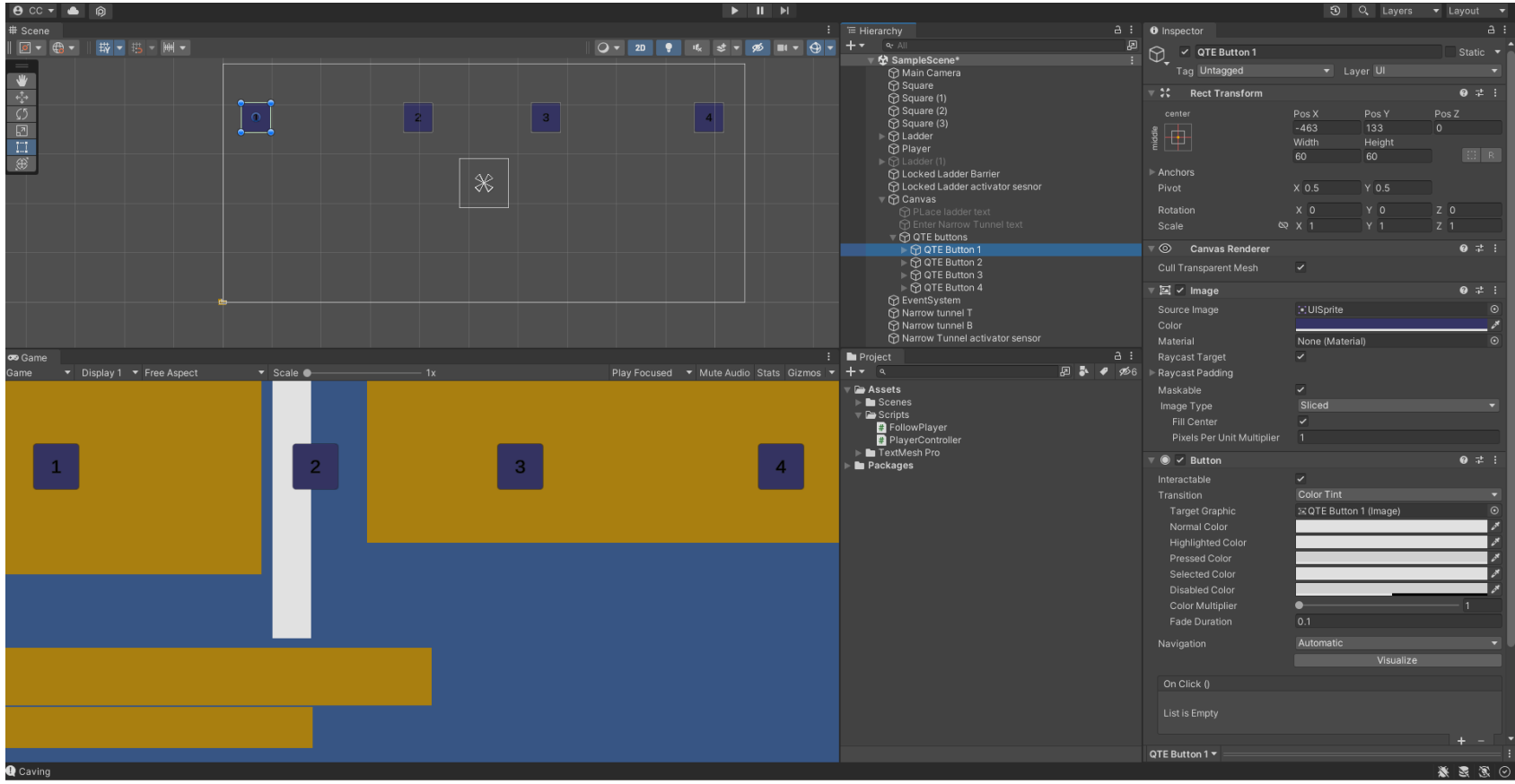
RESPONSE TO FEEDBACK – DAY 37

- The 2nd prototype began when I acted on the feedback that I received from the 1st prototype, adding the ability to only travel down the ladder with the “S” key and not slide down with gravity.
- I also zoomed the camera in to the player when you enter the narrow tunnel – adding to the claustrophobic and constrained feel of the tight squeeze.



PROGRAMMING – DAY 37

- I also added in the quick time event buttons, which currently don't do anything, that will appear between a random spawn position in a random order when entering the narrow tunnel – there will be a time limit to pressing these buttons before they disappear and then appear again, and if they are pressed in the wrong order the fossil bag will tear, if pressed correctly then the player will advance through the tunnel about 1/3 of the way.



PROBLEM SOLVING – DAY 37

- I stopped the player from being able to still move along the horizontal axis when in the narrow tunnel by adding an is crawling variable that must be false in the if statement that allows horizontal movement – this variable is set to true when the player enters the tunnel with “Q”.
- I set the gravity scale to 0 when the player is on the ladder, so the player only moves down the ladder when the “S” key is pressed, and I set the gravity scale back to 1 when you dismount and return to horizontal movement.
- I also gave the bumper a downwards impulse force, as the left impulse is weaker on its own with the gravity being 0 when you collide with it initially.

```
// assigns the horizontal axis input (A and D keys) to the horizontal axis variable
horizontalInput = Input.GetAxis("Horizontal");
// assigns the vertical axis input (W and S keys) to the vertical axis variable
verticalInput = Input.GetAxis("Vertical");
if(isOnLadder == false && isCrawling == false)
{
    // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
    transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
}

if(isOnLadder == true)
{
    // move vertically (y axis) with the W and S keys over time each second at a speed of 10
    transform.Translate(Vector2.up * verticalInput * Time.deltaTime * climbSpeed);
}

// if the player is near the unlocked ladder, presses E, and has at least 1 ladder
if(isInLadderActivator == true && Input.GetKeyDown(KeyCode.E) && laddersOwned > 0)
{
    // places ladder
    ladder.gameObject.SetActive(true);
    isLadderPlaced = true;
    // removes locked ladder barrier
    ladderBarrier.gameObject.SetActive(false);
    placeLadderText.gameObject.SetActive(false);
}

if(isInNarrowTunnelActivator == true && Input.GetKeyDown(KeyCode.Q))
{
    transform.Rotate(0, 0, 90);
    transform.position = new Vector2(-2, -3);
    Debug.Log("Caving");
    enterNarrowTunnelText.gameObject.SetActive(false);
    isCrawling = true;
}
```

```
if(other.gameObject.CompareTag("Ladder"))
{
    isOnLadder = true;
    Debug.Log("ON ladderr");
    // turns iff gravity when the player goes onto the ladder
    playerRb.gravityScale = 0;
}
if (other.gameObject.CompareTag("Bumper"))
{
    // turns gravity back on when the player gets off the ladder
    playerRb.gravityScale = 1;
    // makes the player jump off the ladder when they reach the top
    playerRb.AddForce(Vector2.left * bumperForce, ForceMode2D.Impulse);
    playerRb.AddForce(Vector2.down * bumperForce, ForceMode2D.Impulse);
    isOnLadder = false;
}
```

FollowPlayer.cs | PlayerController.cs

Miscellaneous Files | FollowPlayer

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class FollowPlayer : MonoBehaviour
6  {
7      public GameObject player;
8      private Vector3 offset = new Vector3(0, 2, -10);
9      private Camera cam;
10     private PlayerController playerControllerScript;
11     // Start is called before the first frame update
12     void Start()
13     {
14         playerControllerScript = GameObject.Find("Player").GetComponent<PlayerController>();
15         cam = GetComponent<Camera>();
16     }
17
18     // Update is called once per frame
19     void LateUpdate()
20     {
21         // Offset camera infront of and above the player by adding it to the player's position
22         transform.position = player.transform.position + offset;
23     }
24
25     void Update()
26     {
27         if(playerControllerScript.isCrawling == true)
28         {
29             cam.orthographicSize = 1;
30             offset = new Vector3(0, 0, -10);
31         }
32     }
33 }
34
```

PROGRAMMING – DAY 37

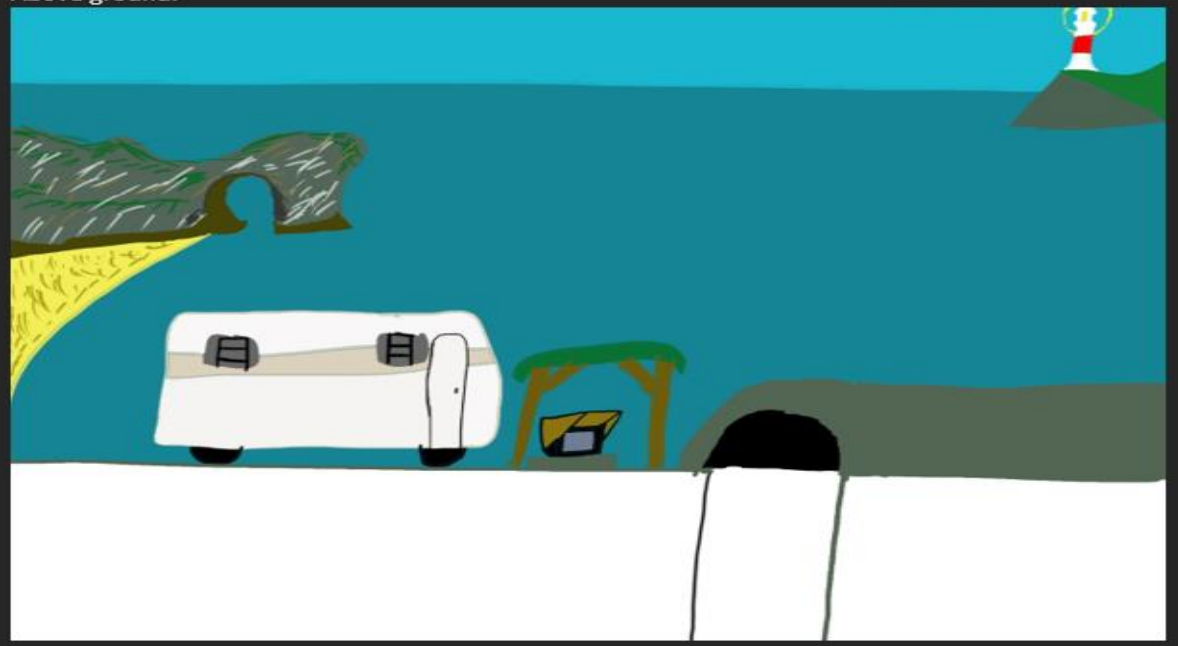
- I returned to the follow player script to zoom the camera in closer to the player when they enter the narrow tunnel – done by assigning the camera to its own variable and changing the orthographic size of the camera to 1 with a new offset when the is crawling variable from the player controller script is true.

Cave.

Fossils.

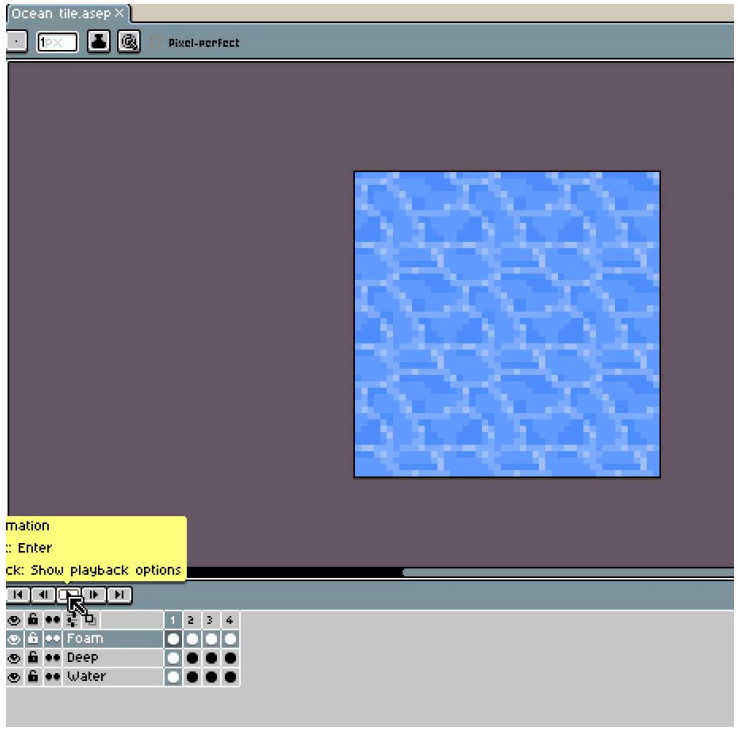
Concept Art:

Above ground.



GAME DESIGN DOCUMENT – DAY 38

I ADDED MY CONCEPT ART INTO THE GAME DESIGN DOCUMENT TO MAKE ITERATION 1.2, WHICH IS THE FINAL ITERATION AS THE REST OF THE DEVELOPMENT I HAVE DONE/WILL BE DOING IS PART OF THE FINAL PRODUCTION OF THE GAME – WHICH SHOULD NOT BE INCLUDED IN A BLUEPRINT FOR MAKING THE GAME AS THE IT SHOULD ONLY INCLUDE PRE-PRODUCTION AND CONCEPTS.

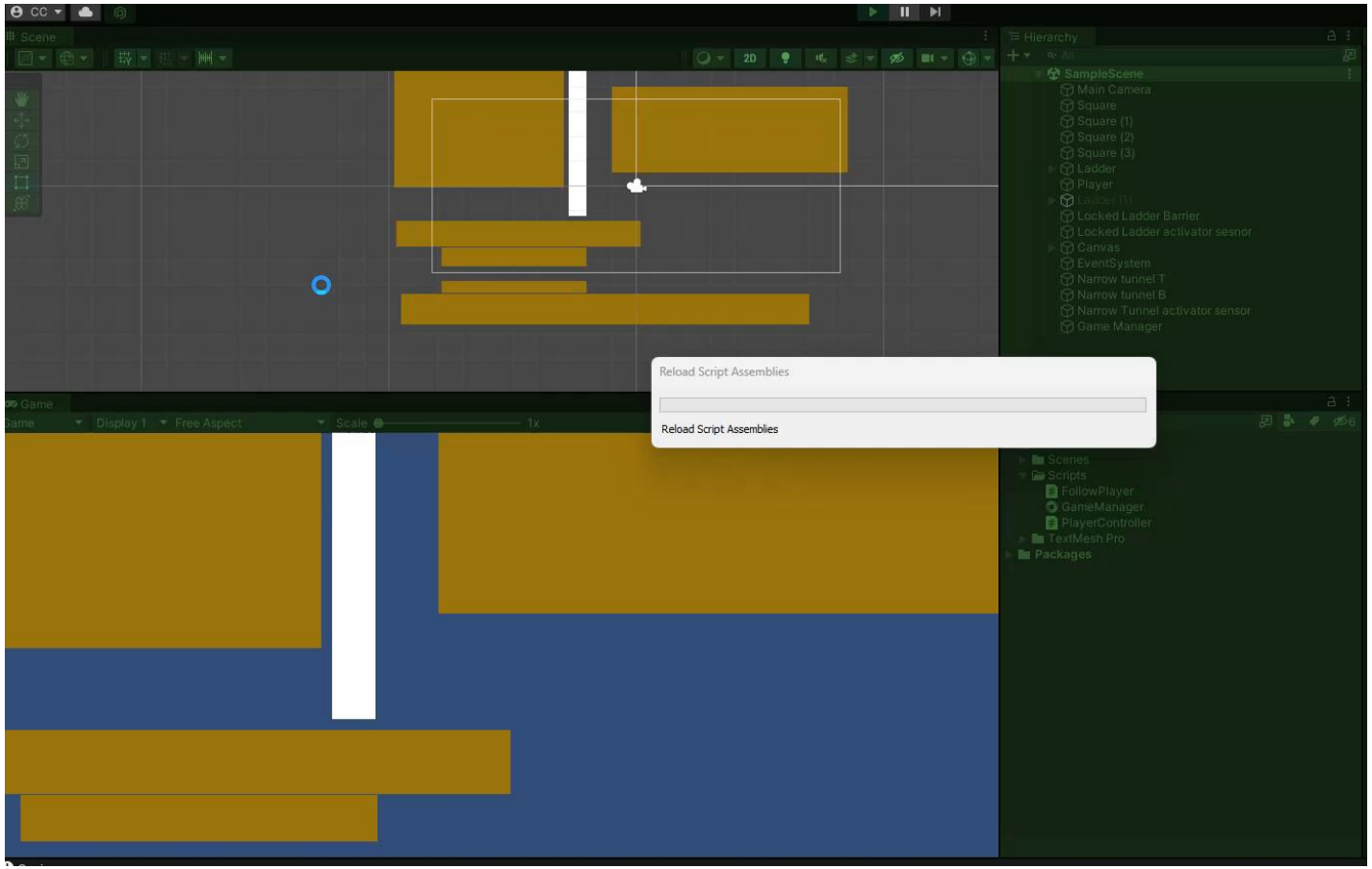


SEA TILE ART – DAY 38

- I used a tutorial I found on YouTube to create and animate a tile for the sea that will be part of the above ground environment – this animation only has four frames that slowly repeat small changes within three layers.

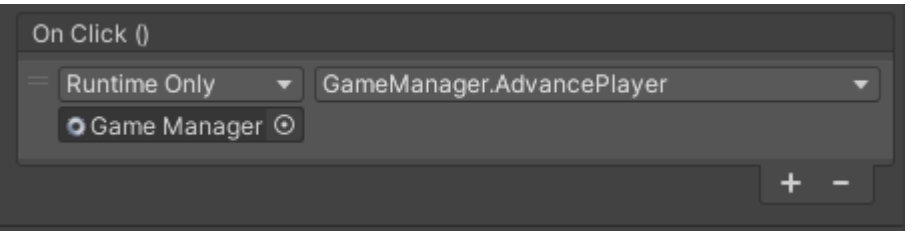
PROGRAMMING – DAY 39

- I created an empty “Game Manager” object and gave it a script of the same name – this new script holds the functions that the quick time event buttons will perform when clicked.
- As of now, the 1st button advances the player 1/3 of the way through the tunnel each time it is clicked.



PROGRAMMING – DAY 39

- I got the is crawling variable from the player controller script and used it in an if statement that sets all the QTE buttons active when it's true.
- I then created the method that slightly moves the player through the narrow tunnel when the QTE button is clicked – this method was added to the 1st button's on click method in the inspector window of unity – shown below.



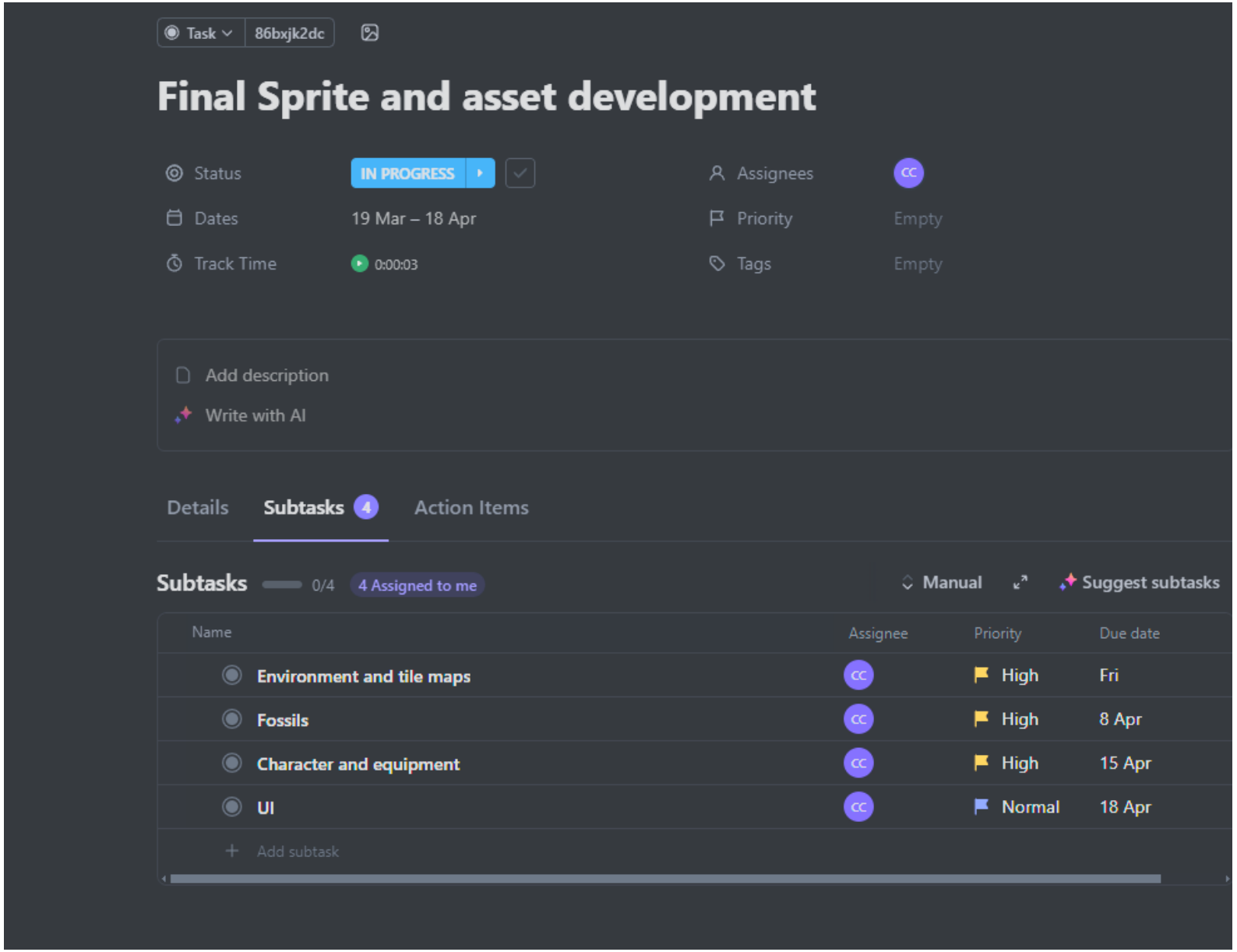
GameManager.cs

Miscellaneous Files — GameManager

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4   using UnityEngine.UI;
5
6   public class GameManager : MonoBehaviour
7   {
8       private PlayerController playerControllerScript;
9       public GameObject quickTimeCollection;
10      public Button[] quickTimeButtons;
11      public GameObject player;
12      // Start is called before the first frame update
13      void Start()
14      {
15          playerControllerScript = GameObject.Find("Player").GetComponent<PlayerController>();
16      }
17
18      // Update is called once per frame
19      void Update()
20      {
21          if (playerControllerScript.isCrawling == true)
22          {
23              quickTimeCollection.gameObject.SetActive(true);
24          }
25      }
26
27      public void AdvancePlayer()
28      {
29          player.transform.Translate(Vector2.up * 200 * Time.deltaTime);
30      }
31  }
32
```

DEADLINE UPDATE – DAY 41

- I adjusted the deadlines of the final sprite and asset subtasks (shown on click up on the right) – allowing more time for the environment by aligning its deadline with prototype 2/implementing tile maps and giving more time to making the fossils and character by cutting the UI development time down significantly.



CONTENTS – WEEK 7

ART

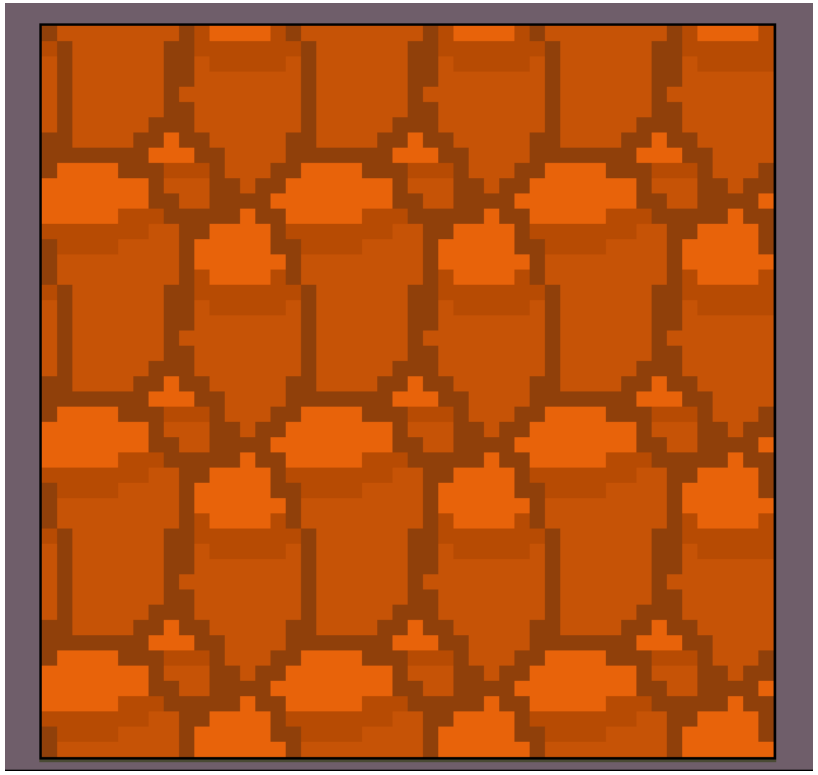
PROGRAMMING

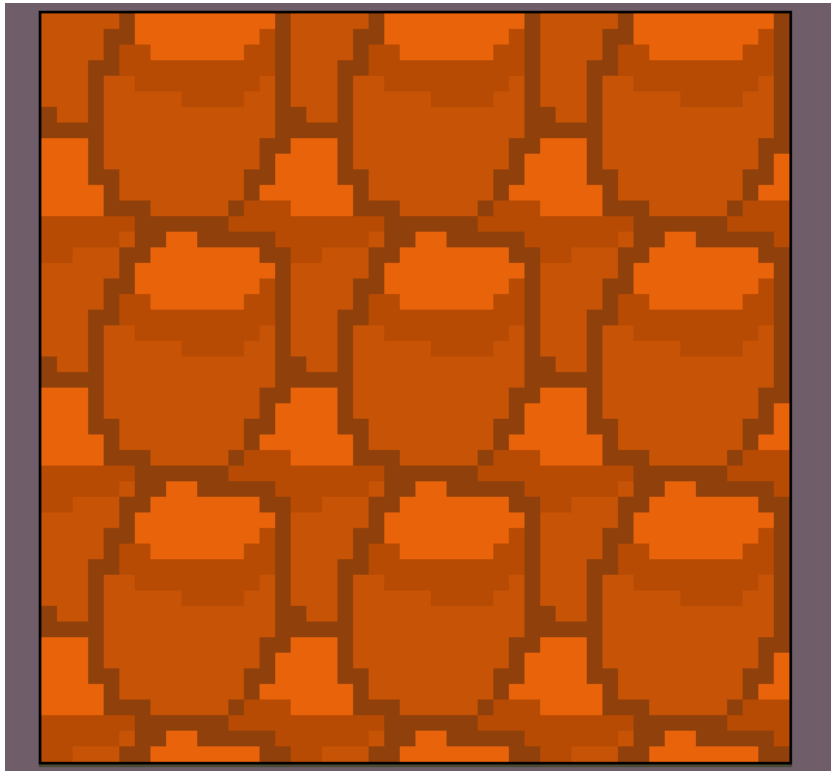
MILESTONE

PROJECT MANAGEMENT

CAVE TILE ART – DAY 45

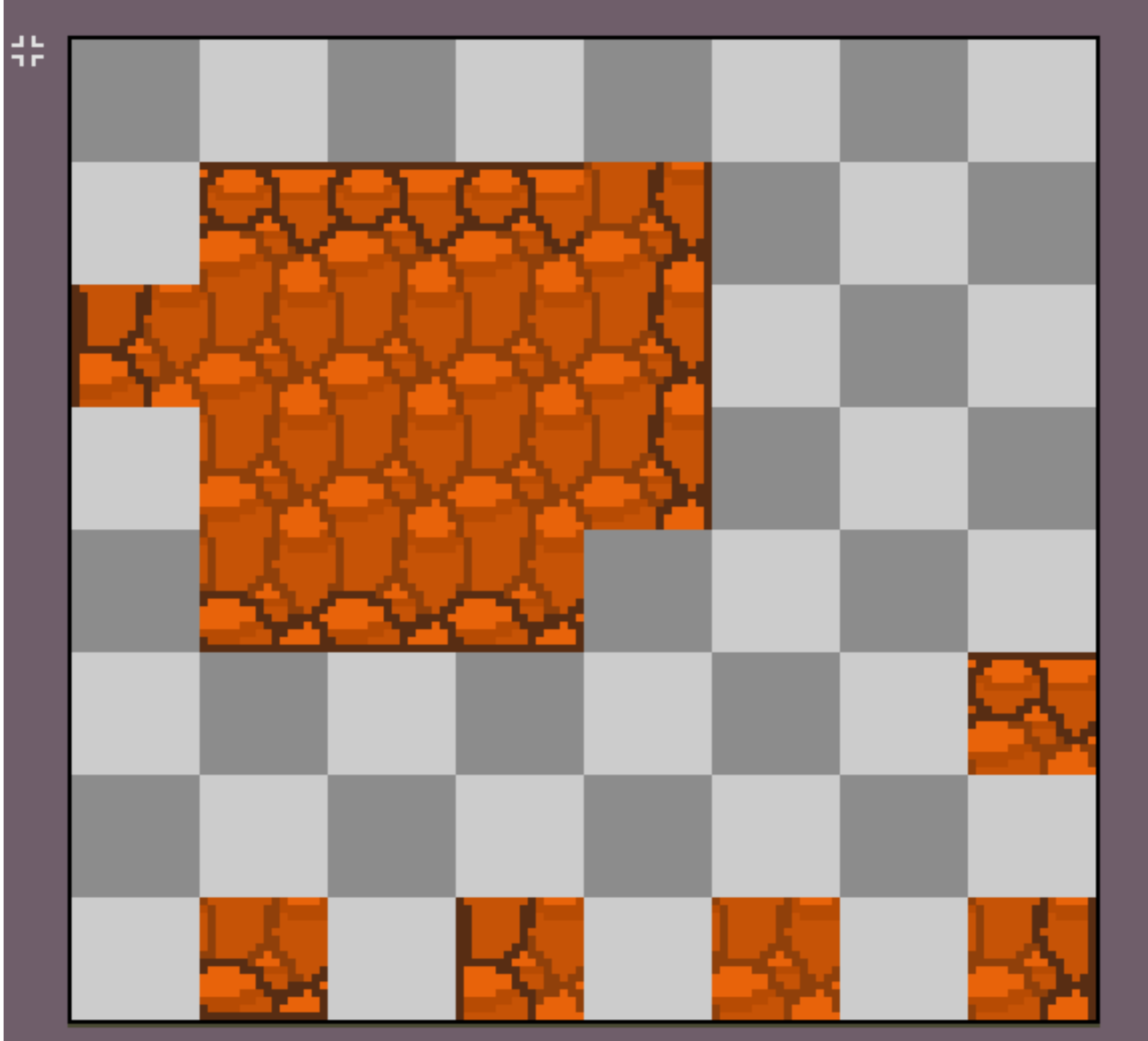
I EXPERIMENTED WITH THESE TWO VERSIONS OF THE CAVE TILE BEFORE FINALIZING IT – EVENTUALLY DECIDING THAT THE VERSION SHOWN ON THE LEFT WILL BE ITS FINAL ITERATION.





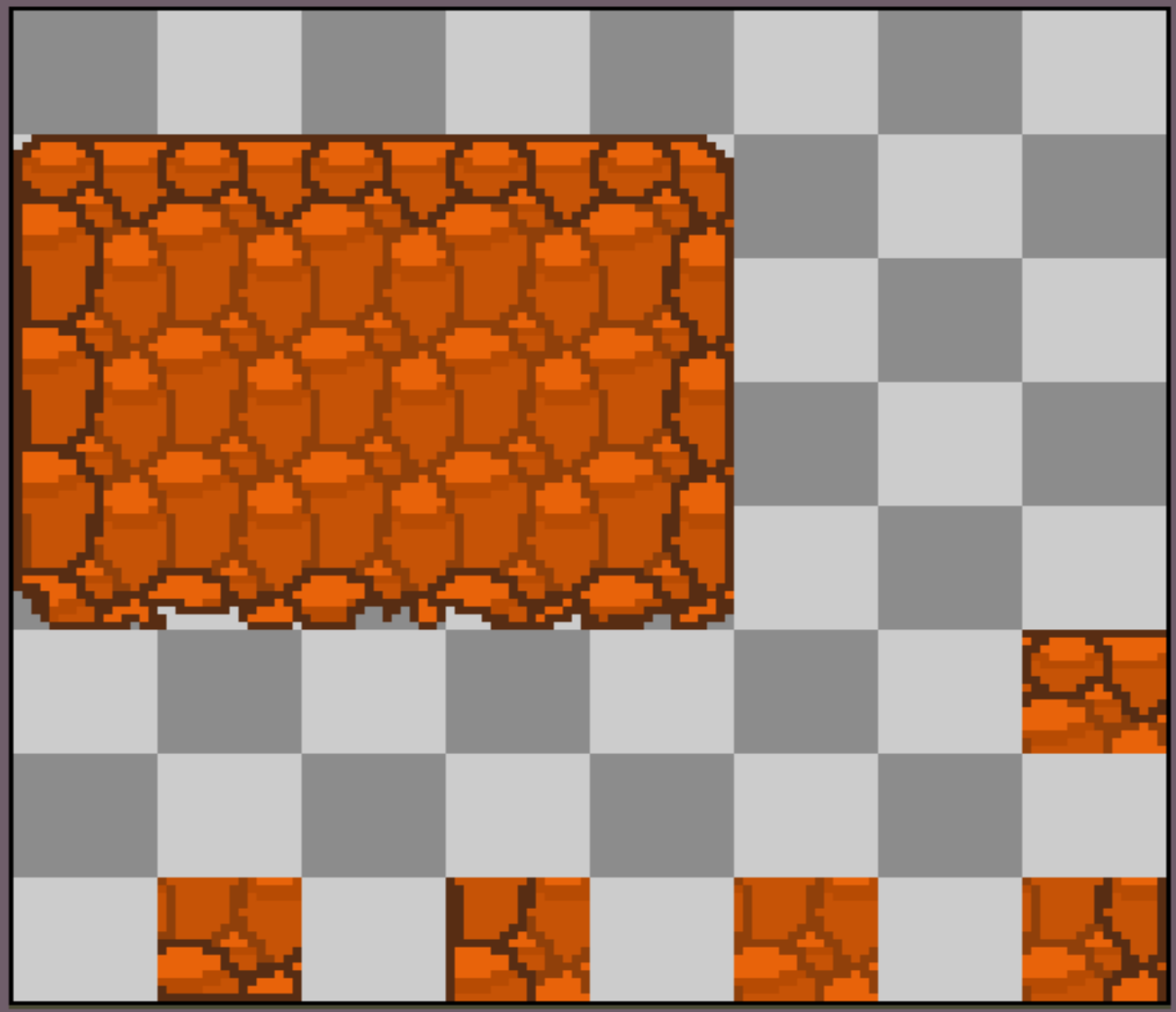
CAVE TILE ART – DAY 45

- I changed the tiled mode to only the X axis to create the top and bottom edge tiles and did the same but with the Y axis for the left and right edges.
- Finally, I brought the tile set together into a 128x128 grid (the tiles are 16x16) to form what is shown on the right – I still need to sort out the corners as the edges don’t join up.



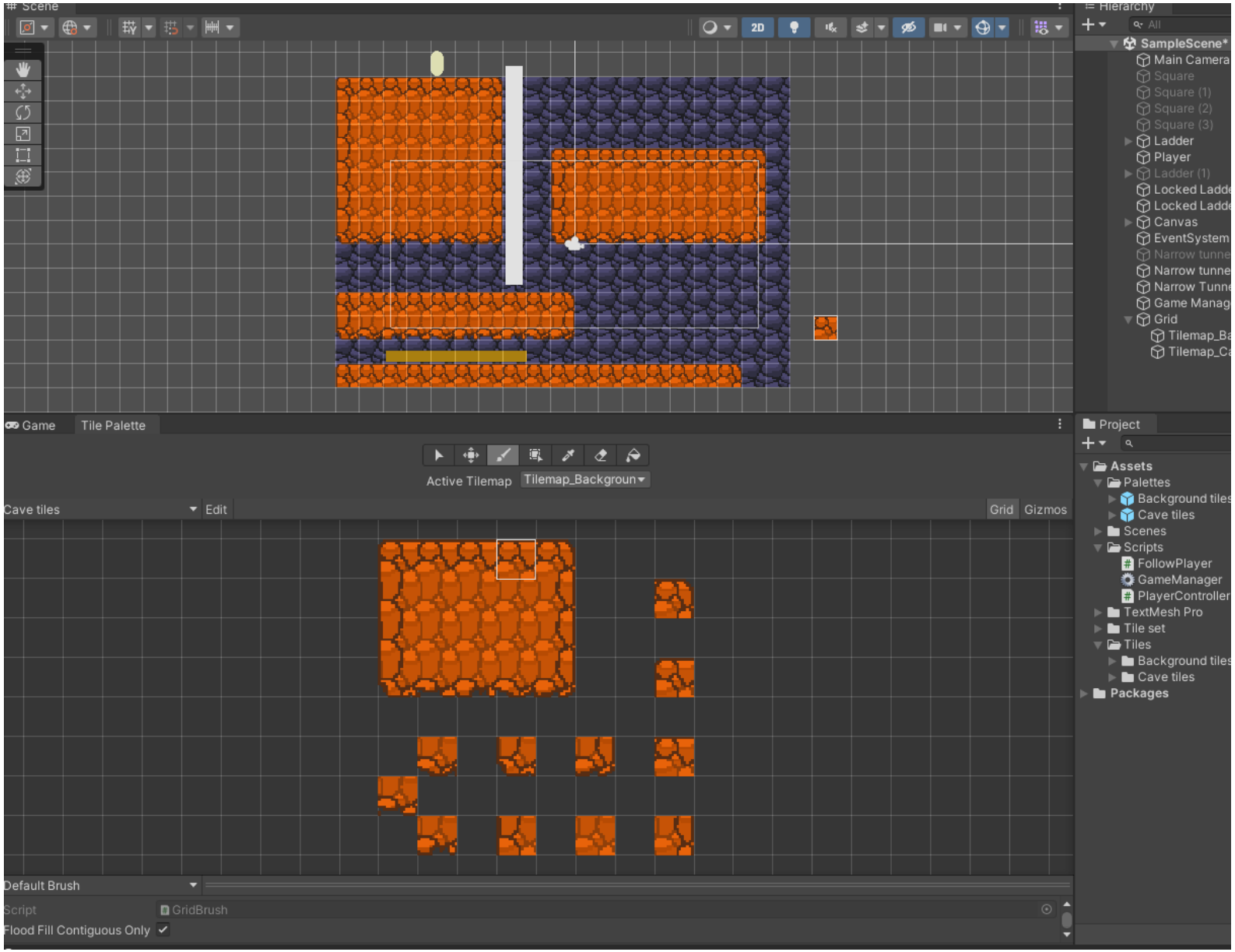
CAVE TILE ART – DAY 46

- I duplicated the top tiles and tweaked them manually to create the corner tile, then edited a cracked effect onto the bottom tiles with the eraser and brush tools.



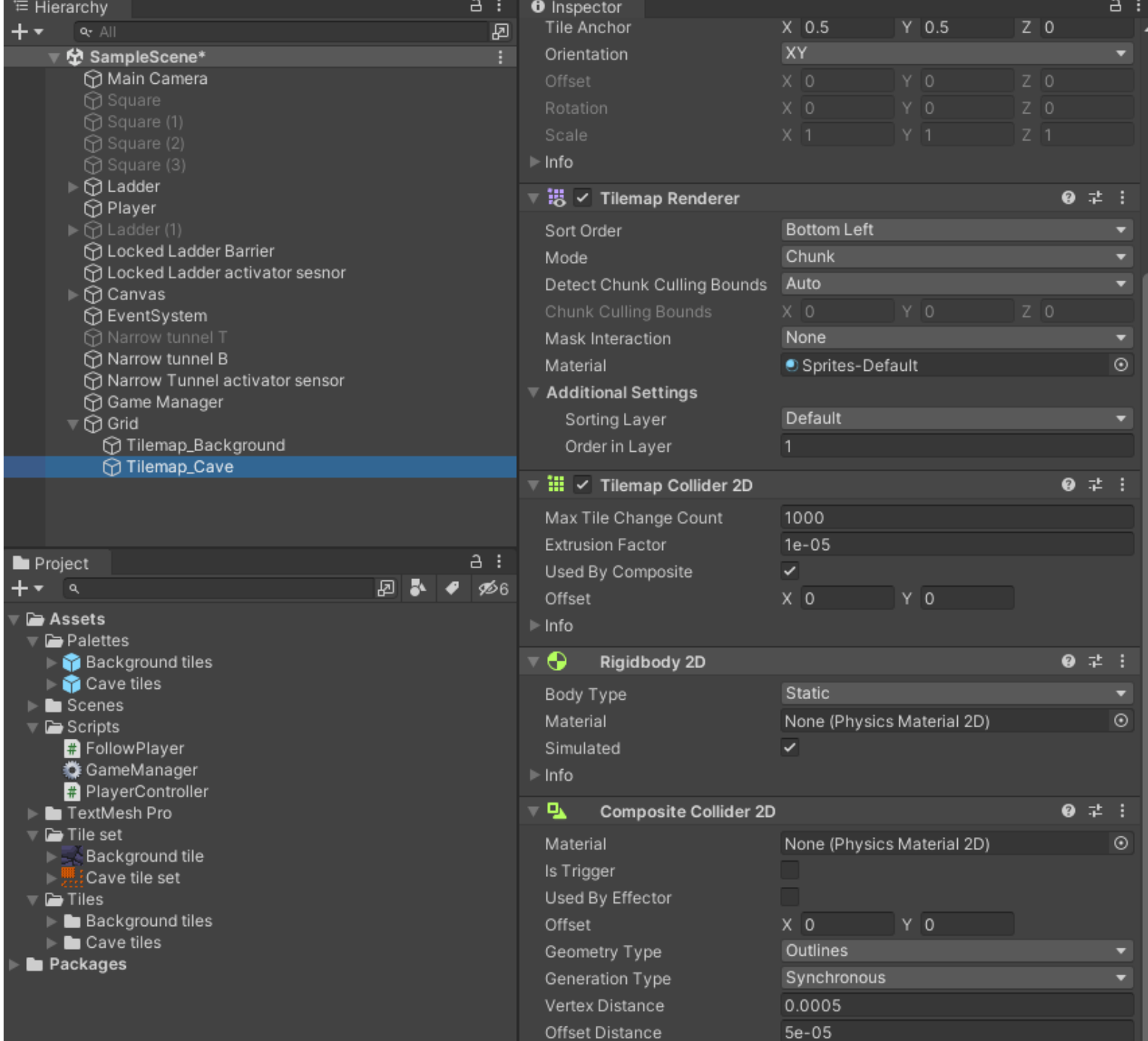
ASSET IMPLEMENTATION AND DESIGN – DAY 46

- I imported the cave tile set and the background tile into unity and adjusted their compression and filter settings in the inspector before adding them both into their own new palette and tile map grid.
- Once that was done, I then was able to paint the background tiles in and paint the different cave tiles over the top of them as they are on their own layer in front of the background tile grid.



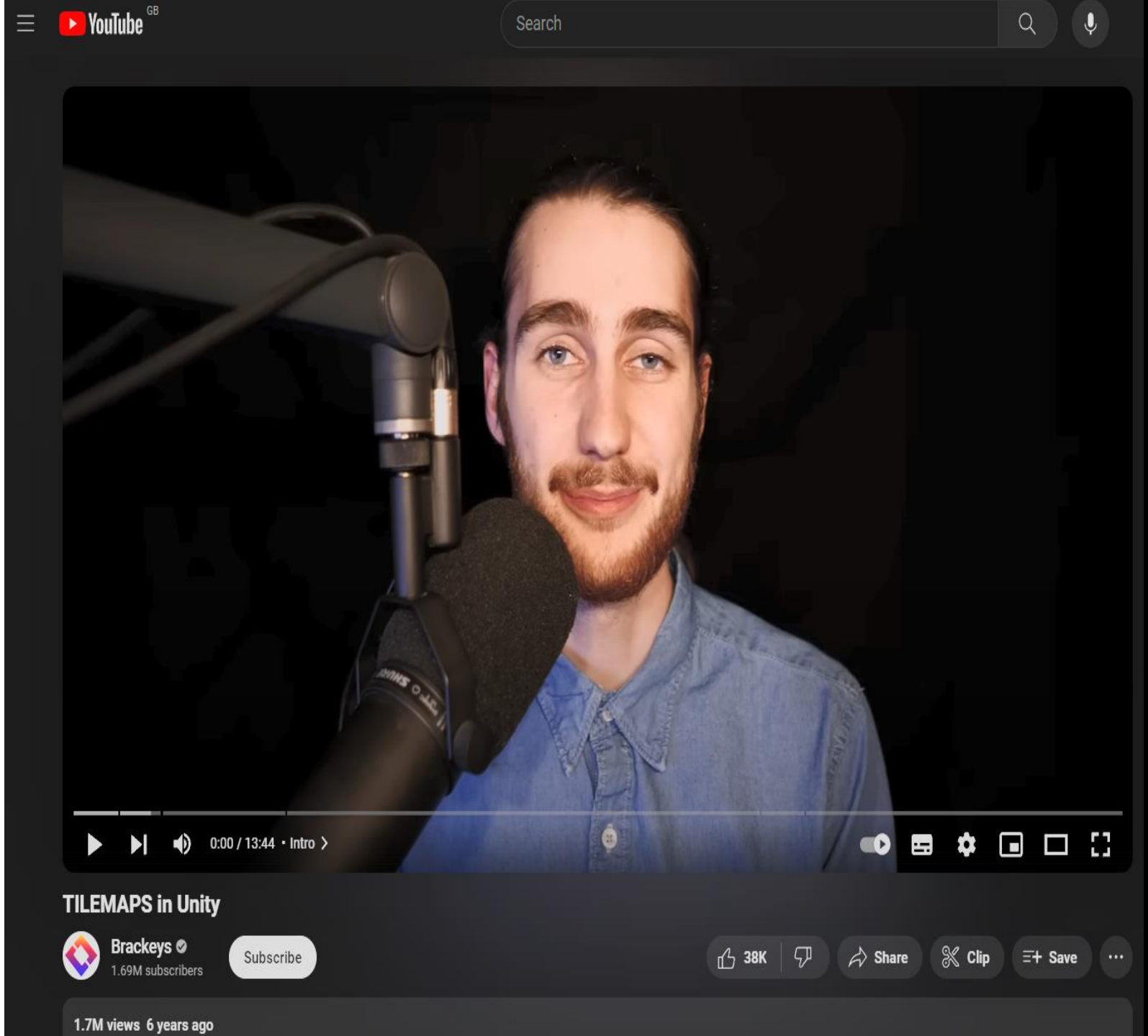
PROBLEM SOLVING AND DESIGN – DAY 46

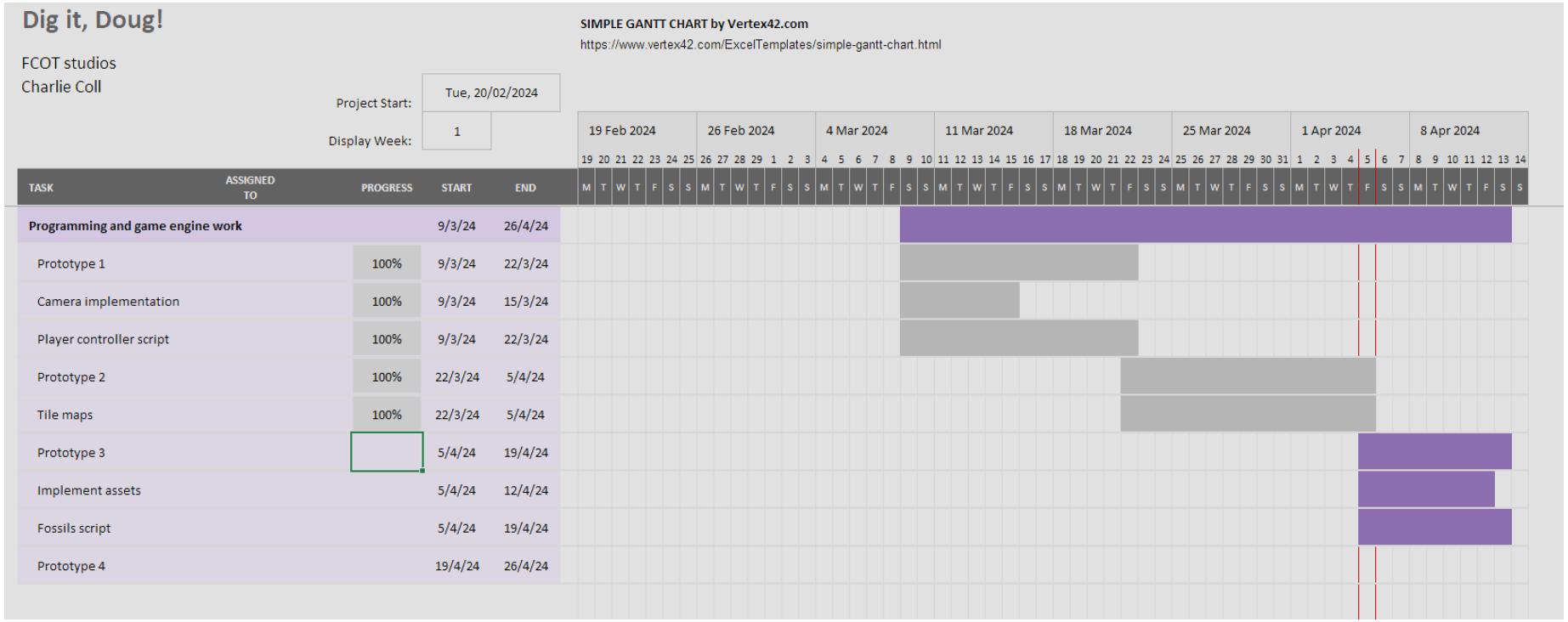
- I finally gave the cave tiles a tile map collider 2D with a composite collider – which the tile map collider uses to give each segment of tiles one collider instead of an individual collider for every tile – which gives the tiles a rigid body, where is et the body type to static.



TUTORIAL – DAY 46

- I learnt this process of adding tile sets into unity via a YouTube tutorial by Brackeys.





Dig it, Doug!

FCOT studios
Charlie Coll

SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

Project Start: Tue, 20/02/2024
Display Week: 1

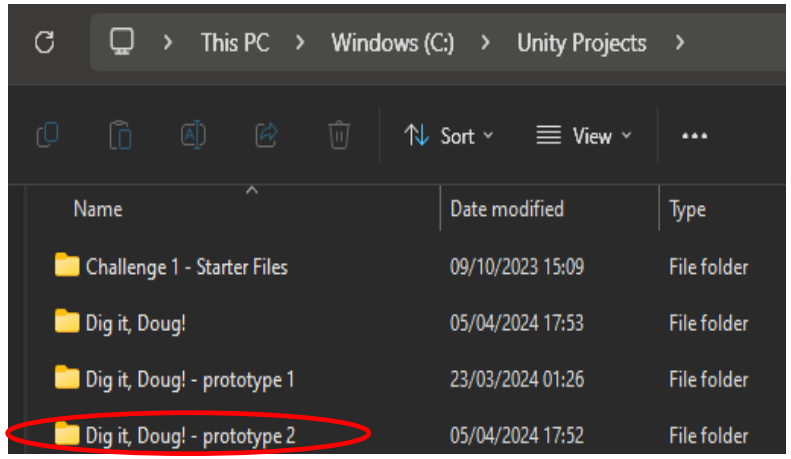
TASK	ASSIGNED TO	PROGRESS	START	END
Programming and game engine work			9/3/24	26/4/24
Prototype 1		100%	9/3/24	22/3/24
Camera implementation		100%	9/3/24	15/3/24
Player controller script		100%	9/3/24	22/3/24
Prototype 2		100%	22/3/24	5/4/24
Tile maps		100%	22/3/24	5/4/24
Prototype 3			5/4/24	19/4/24
Implement assets			5/4/24	12/4/24
Fossils script			5/4/24	19/4/24
Prototype 4			19/4/24	26/4/24

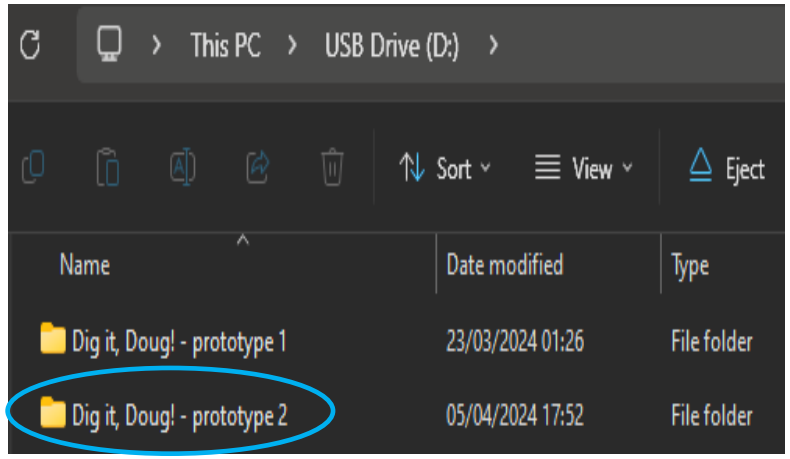
MILESTONE – DAY 47

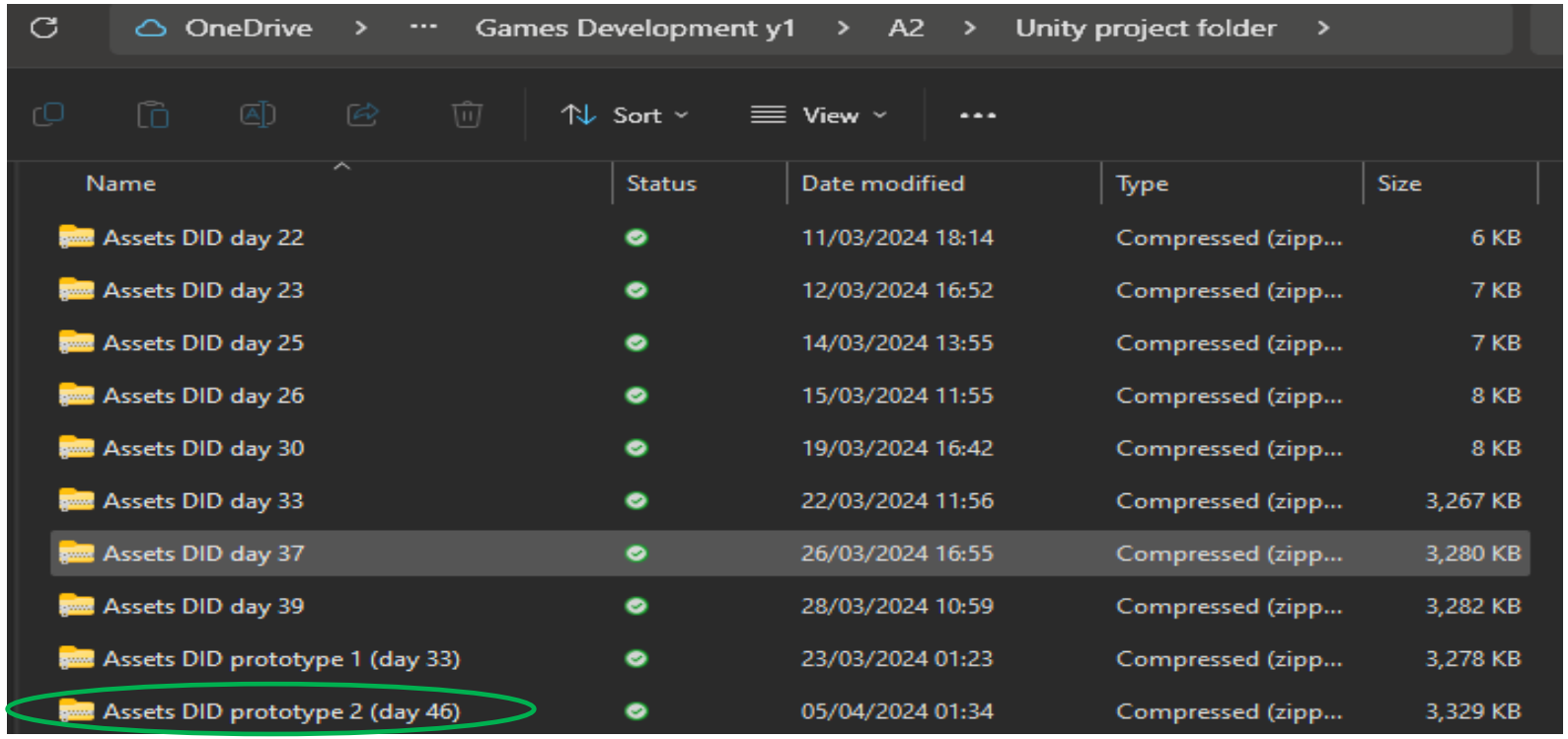
- Prototype 2 is now at its final stage, as I've implemented everything required and learnt how to include my own tile maps in the project.

BACKUPS– DAY 47

- With that, I fully backed up prototype 2's project folder onto my C-Drive and USB drive and backed up its assets to the folder of incremental backups in the OneDrive.

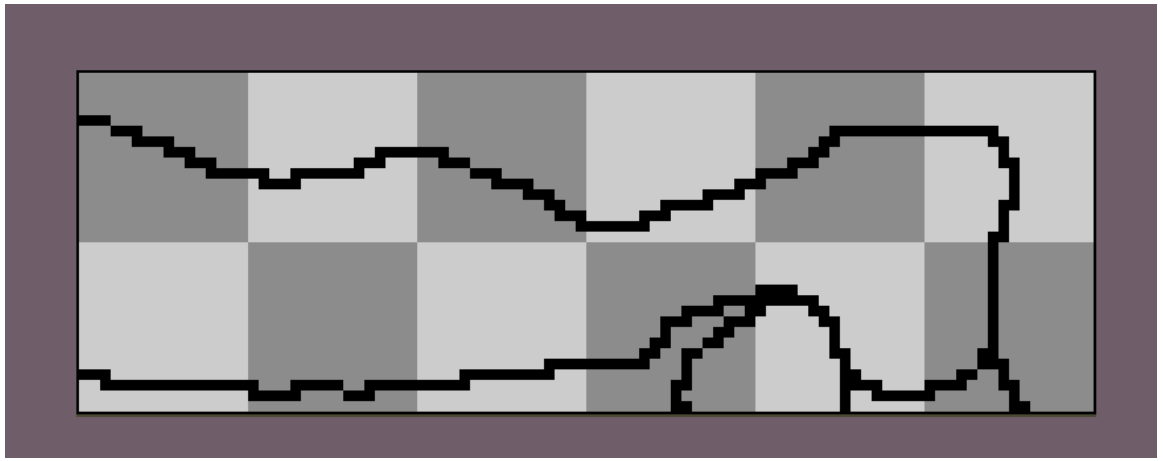


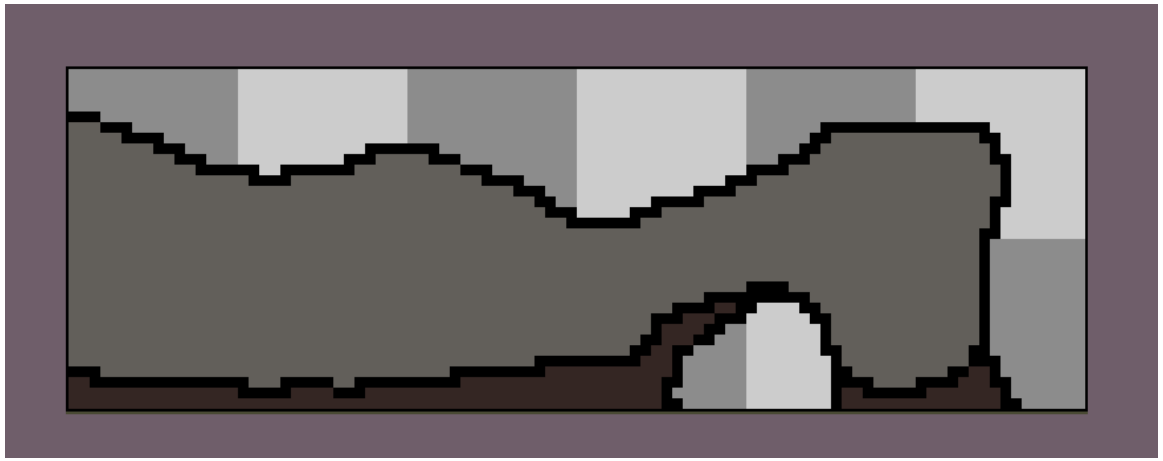




ENVIRONMENT ART – DAY 47

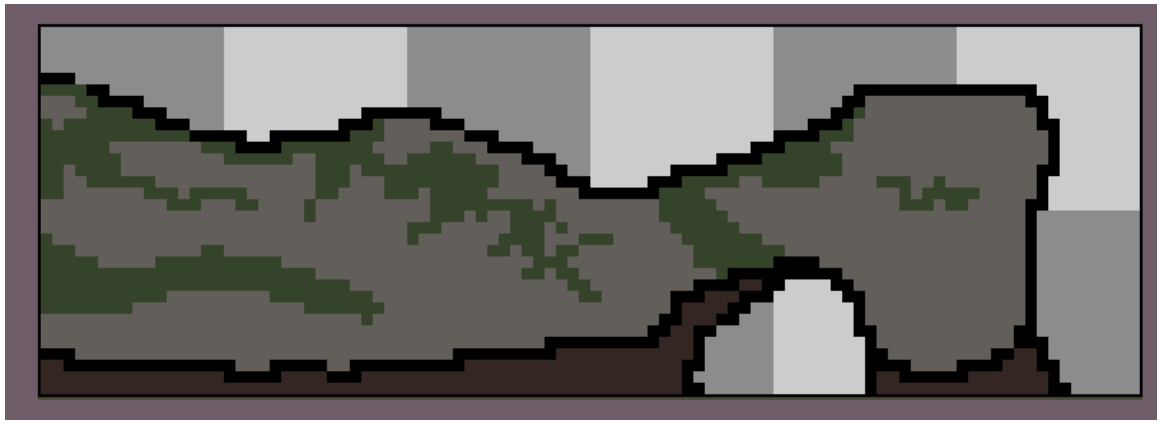
- I started creating the final Durdle door game art by drawing its outline and then filling in the basic colors.





ENVIRONMENT ART – DAY 47

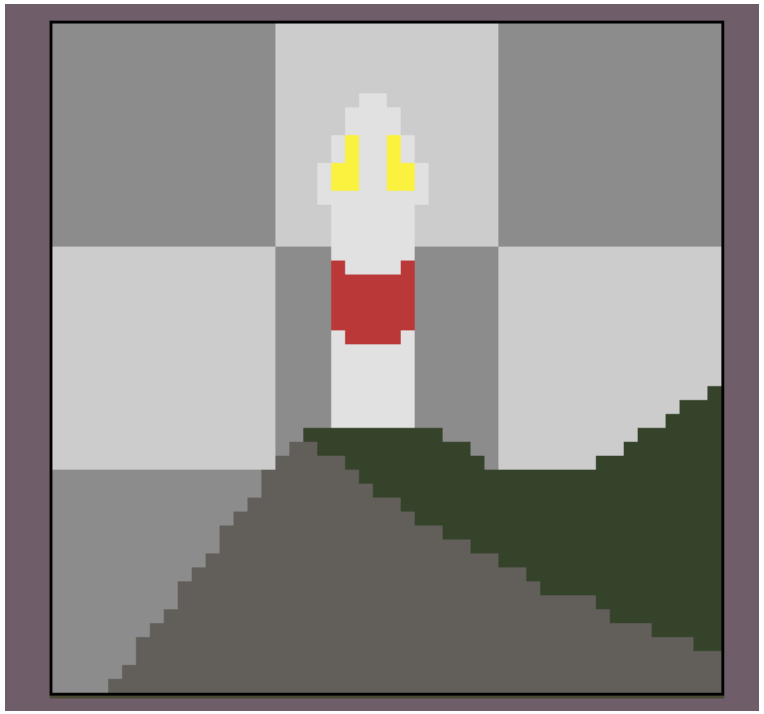
- After the basics, I added in the grass, then the cracks, then I did some shading with various greens, greys, and browns to finish the artwork.

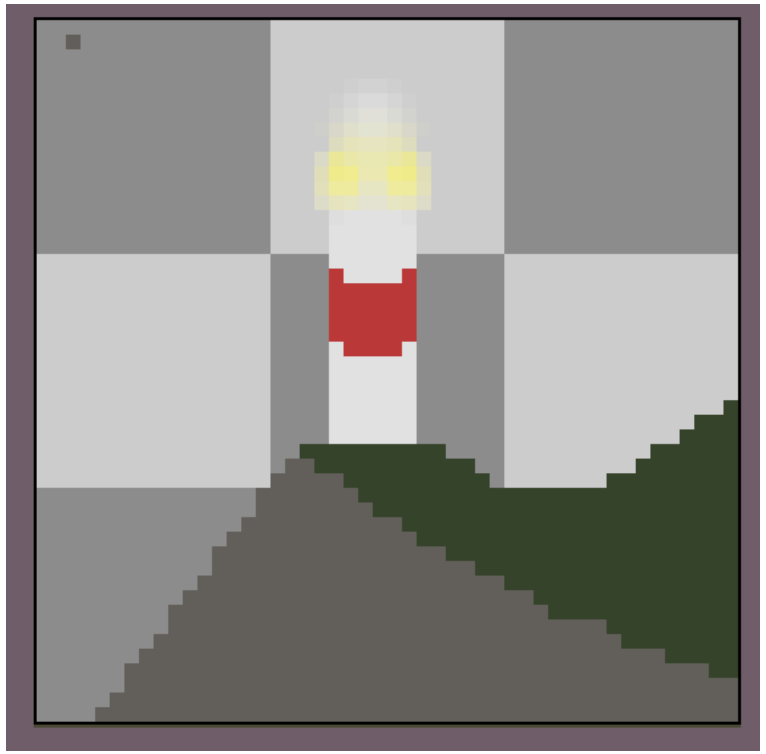


CONTENTS – WEEK 8

ART FEEDBACK MILESTONE ANIMATION PROGRAMMING

ENVIRONMENT ART – DAY 49

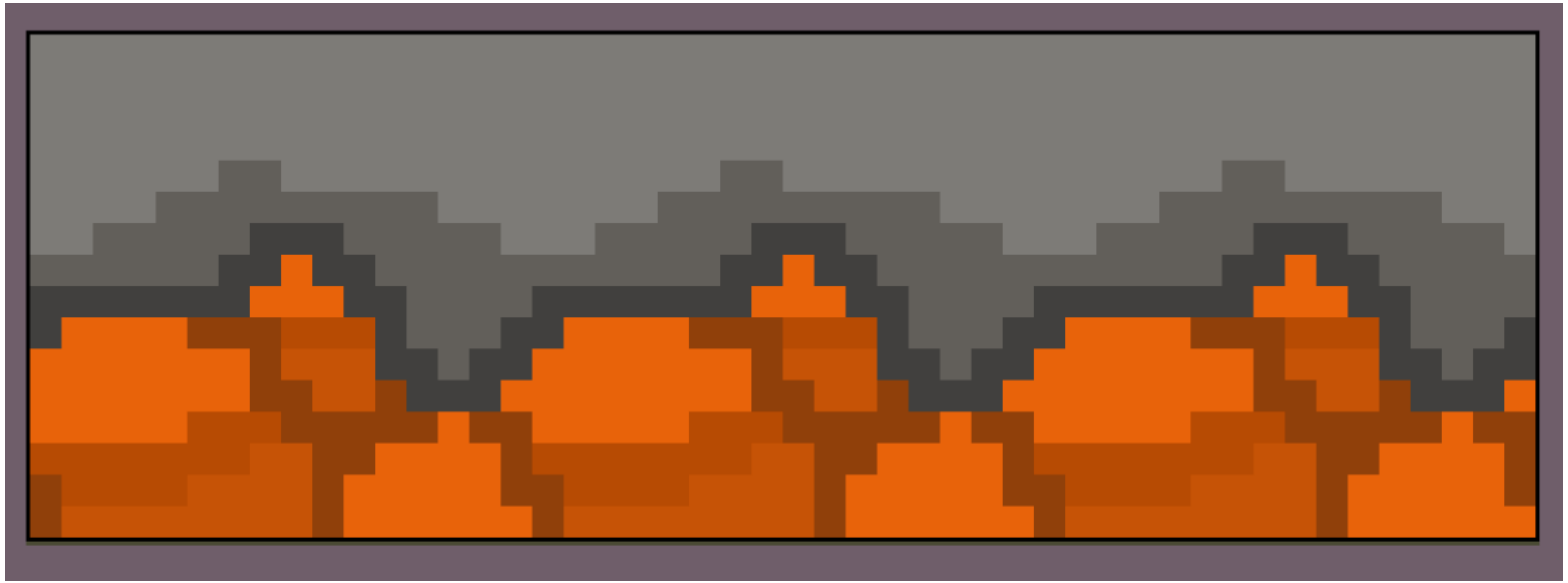




- I started with the basic colors of the peninsula and lighthouse – creating the foundation with these shapes and colors.
- I then discovered the blur tool by accident when switching tools and pressing a different shortcut key, and discovered a good method to create the effect of the light itself.

ENVIRONMENT ART – DAY 49

I ADDED SOME MORE YELLOW AROUND THE LIGHT AND USED THE BLUR TOOL ON IT BEFORE FINISHING THE ART BY SHADING THE PENINSULA.

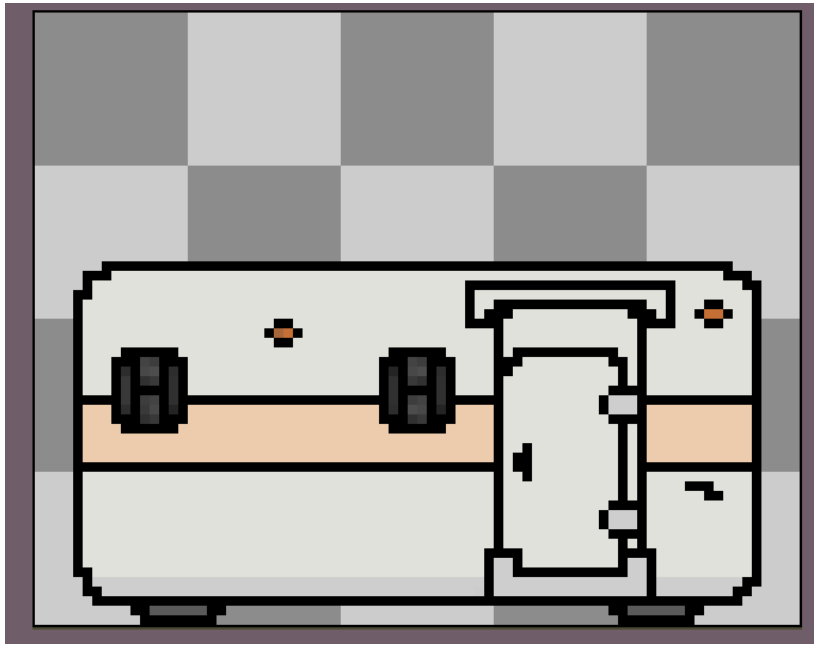


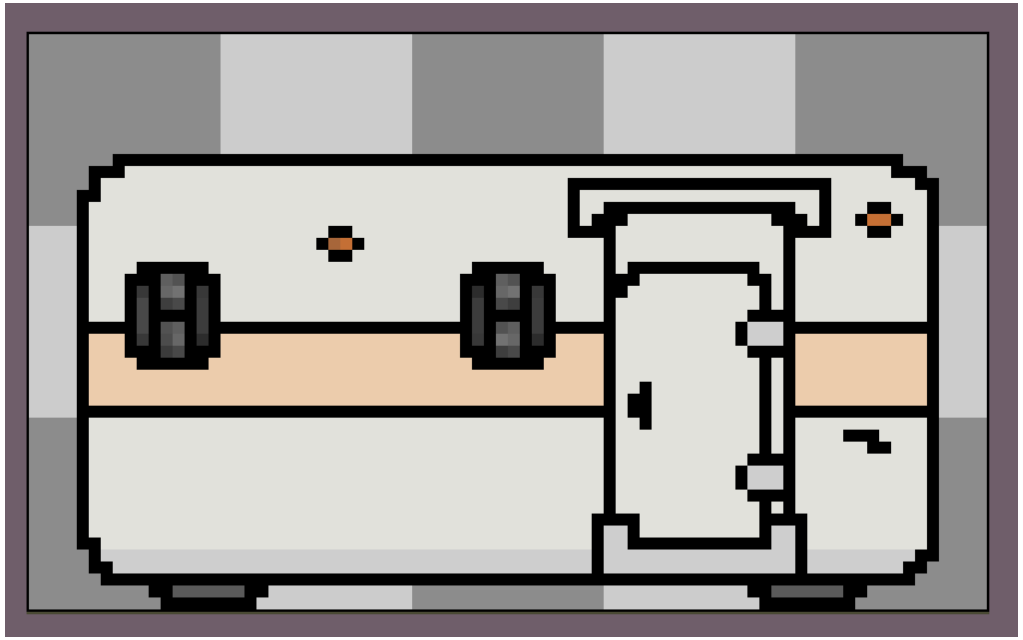
SURFACE TILE ART – DAY 49

- I made another version of the cave top tile – this one featuring grey colors at the top – which will be the surface ground tile that separates the caves from the surface and the coast.

ENVIRONMENT ART – DAY 50

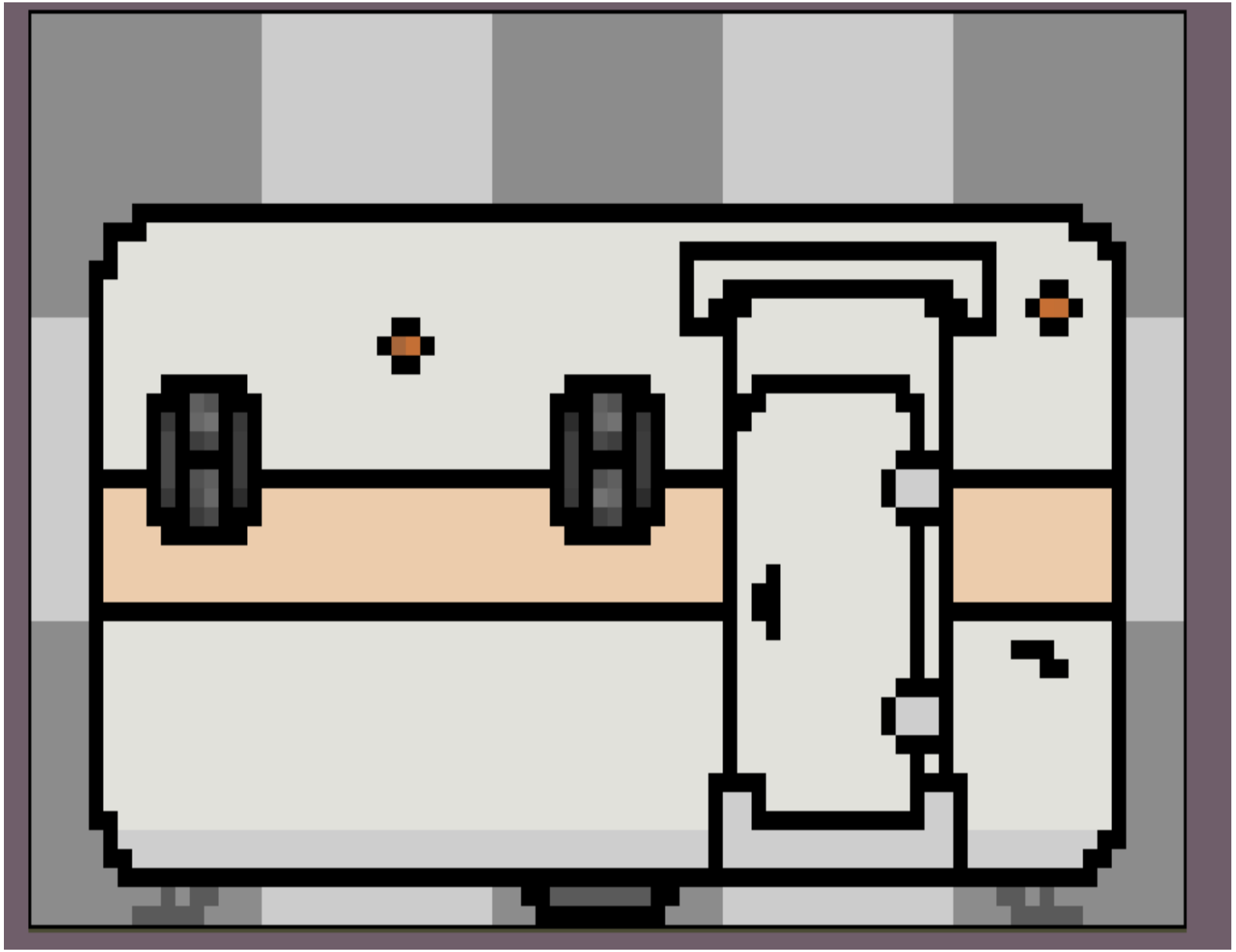
- I made the caravan via the same process, but I forgot to take more screen shots during the first stages of making it as I was too invested in the creation process!
- I used the blur tool to fill in the windows and darken the orange reflectors.
- I finally decreased the canvas size; I only used the first three vertical tiles.





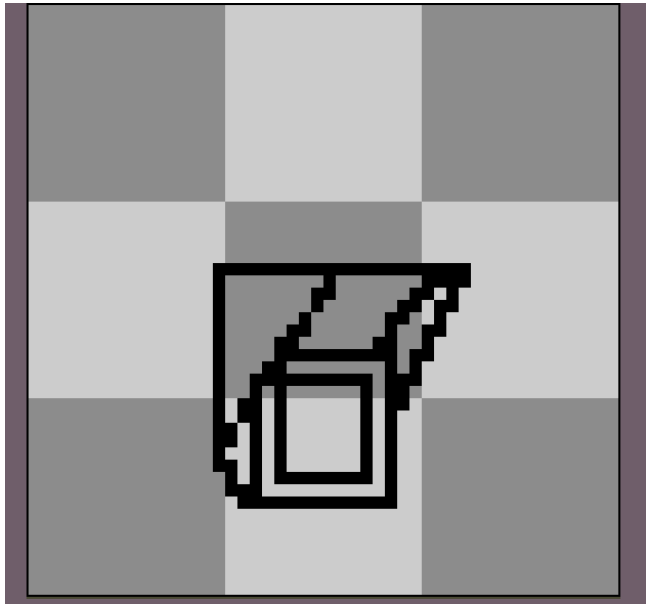
FEEDBACK (FROM MY DAD) – DAY 51

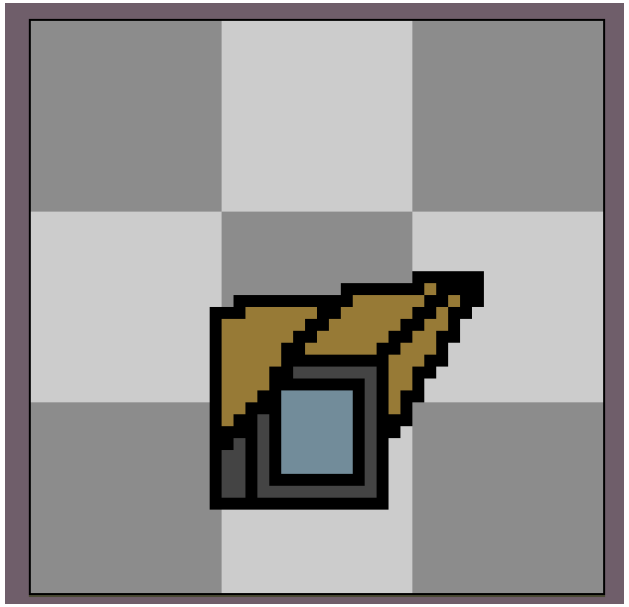
- After seeing the caravan shown on the previous slide, my dad told me that caravans like this one only have wheels in the center as it allows them to be towed by other vehicles and driven round corners (as wheels on caravans don’t turn)
- After receiving this feedback, I made the changes seen on the right.

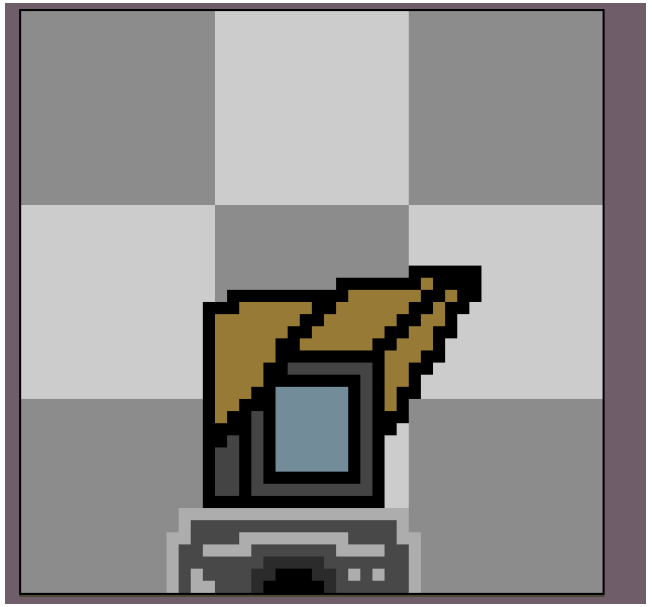


NEW REFERENCE IMAGE – DAY 51

- The reference image I used in my mood board of John Hammond's caravan obstructs the wheels with benches and other items, so I made a guess when adding the wheels onto the caravan based off other vehicles like regular cars.
- The image on the right is the reference I used when updating the caravan after receiving feedback.

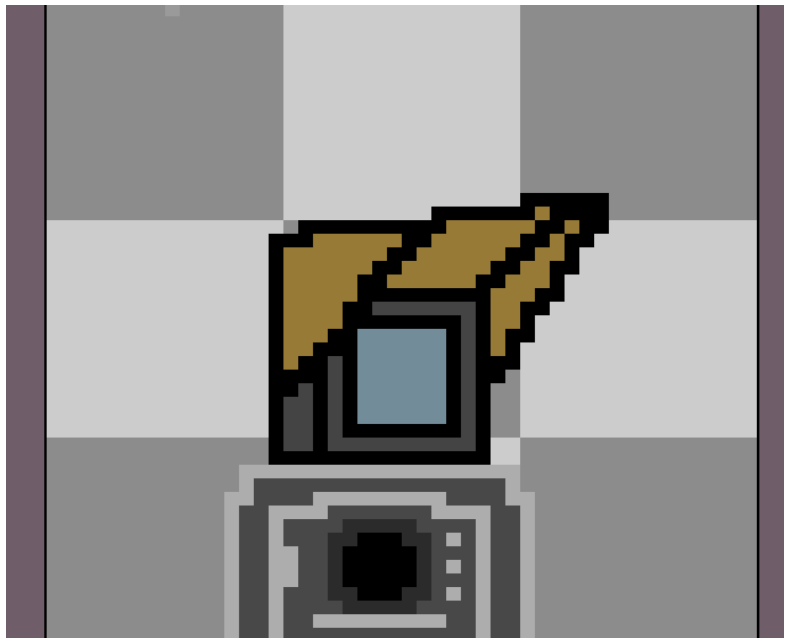


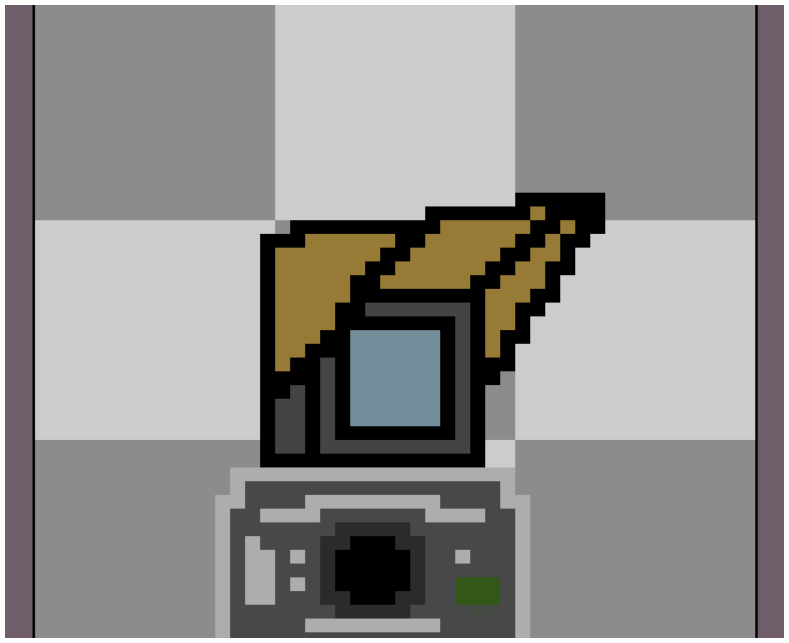


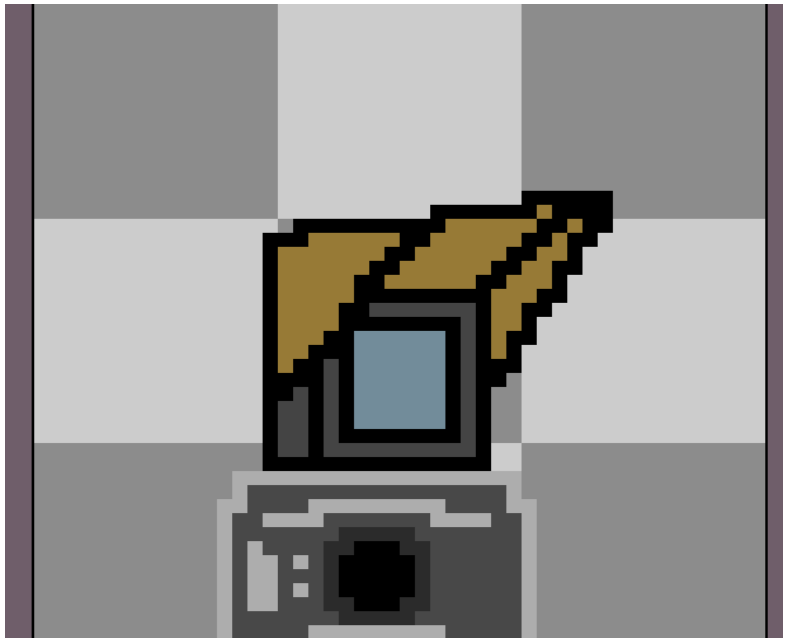


ENVIRONMENT ART – DAY 51

- I started the fossil scanner with the basic outline followed by filling in the colors and making the top of the cardboard visor slant upwards.
- Next, I added the scanner itself that the display sits upon, this went through many iterations of silver colors before I decided on the silver shown above.

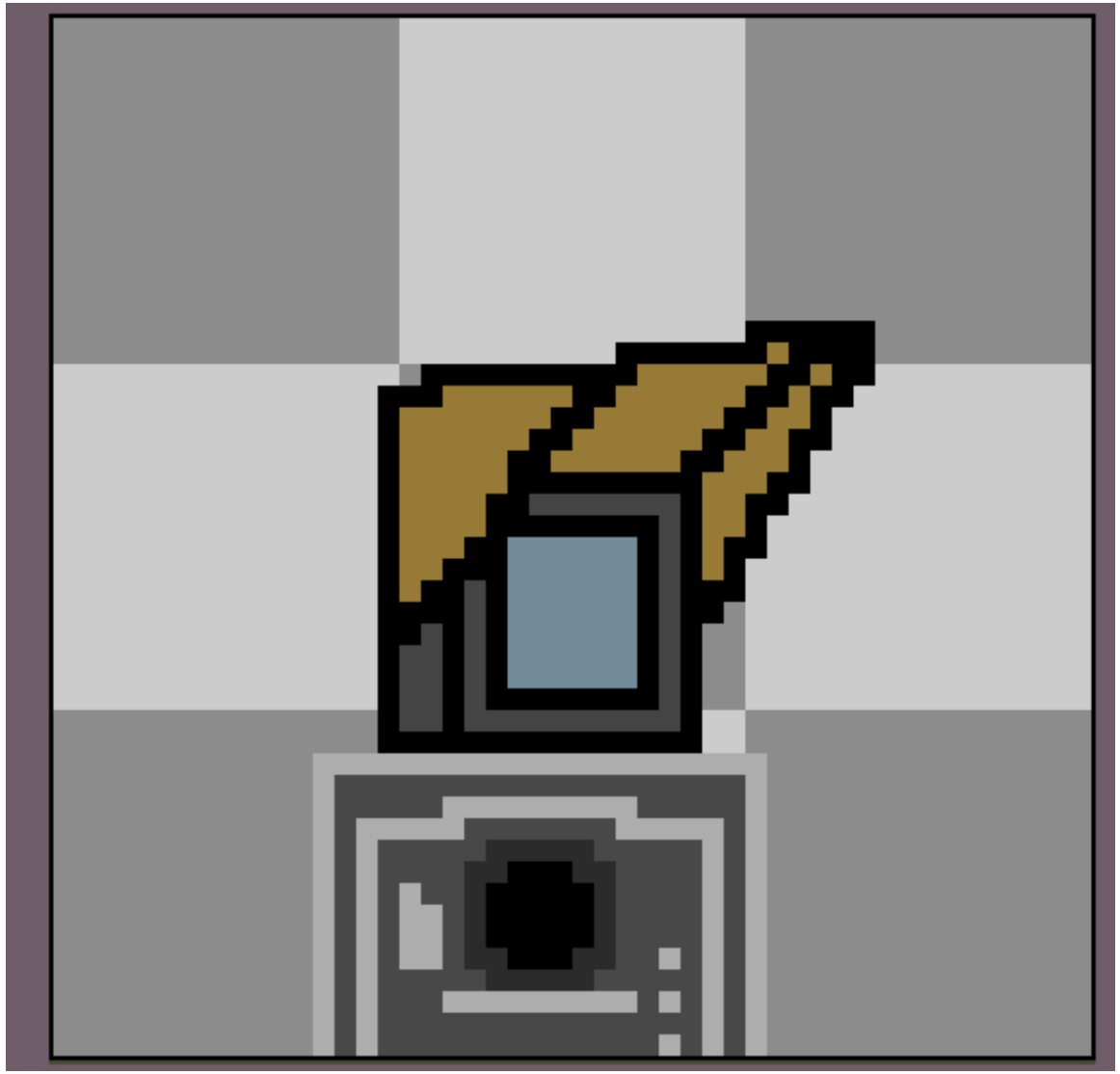






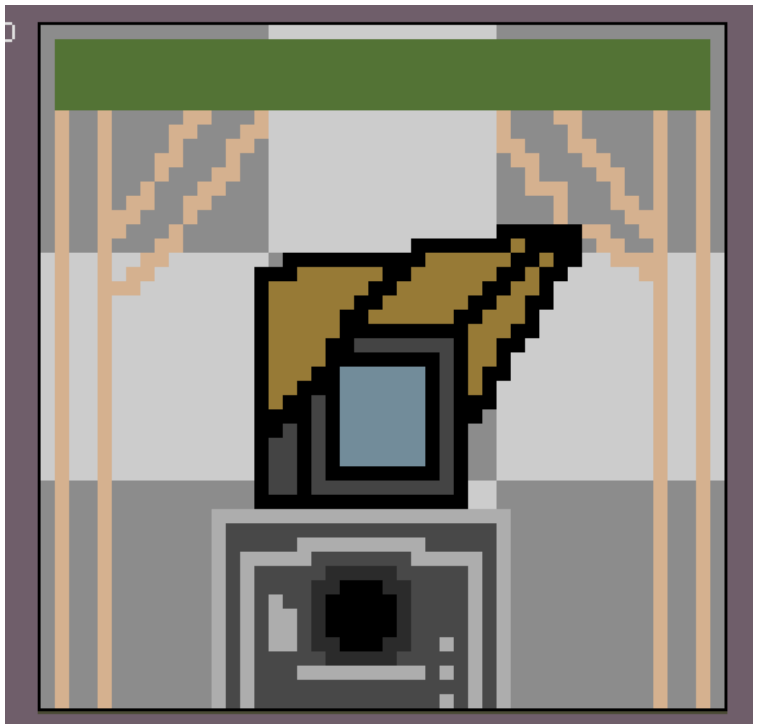
ENVIRONMENT ART – DAY 51

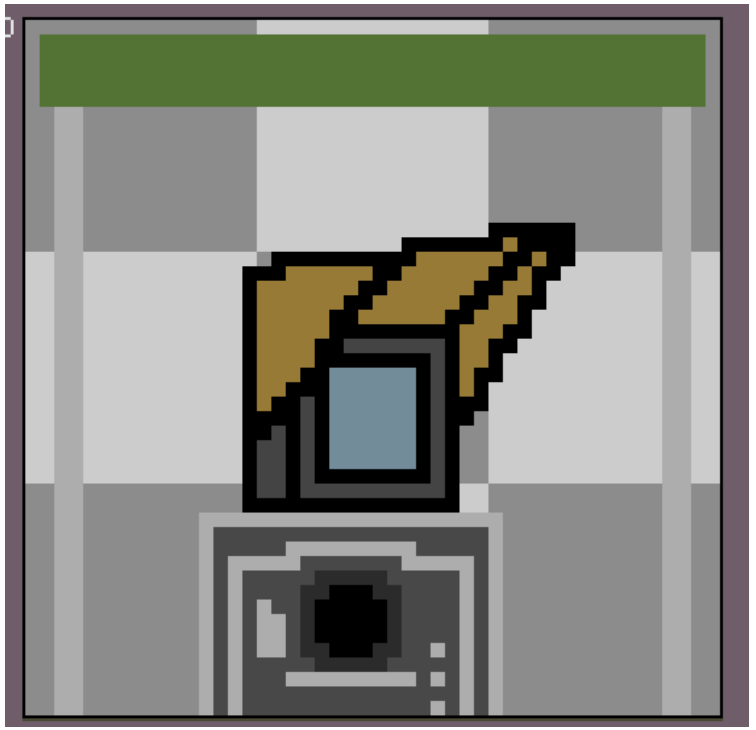
I THOUGHT I SHOULD MAKE MORE OF THE SCANNER VISIBLE SO YOU CAN SEE THE ENTIRETY OF THE SLOT WHERE YOU WOULD PUT THE FOSSIL SAMPLES, THEN I WENT THROUGH A FEW ITERATIONS THAT INCLUDED DIFFERENT BUTTON LAYOUTS BEFORE REACHING THE VERSION ON THE RIGHT.

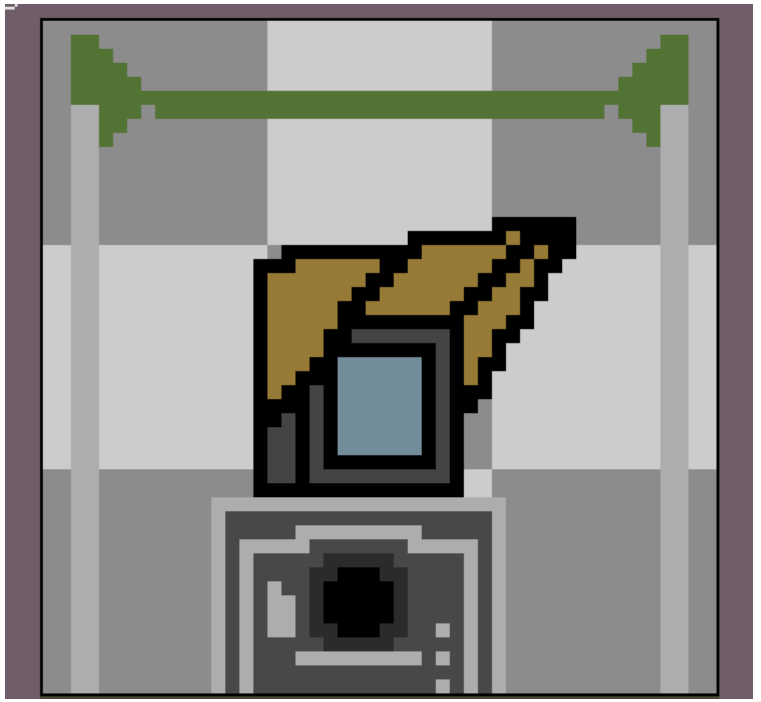


ENVIRONMENT ART – DAY 51

- Finally, I made some more changes to the button layout and the shape of the scanner to complete it.





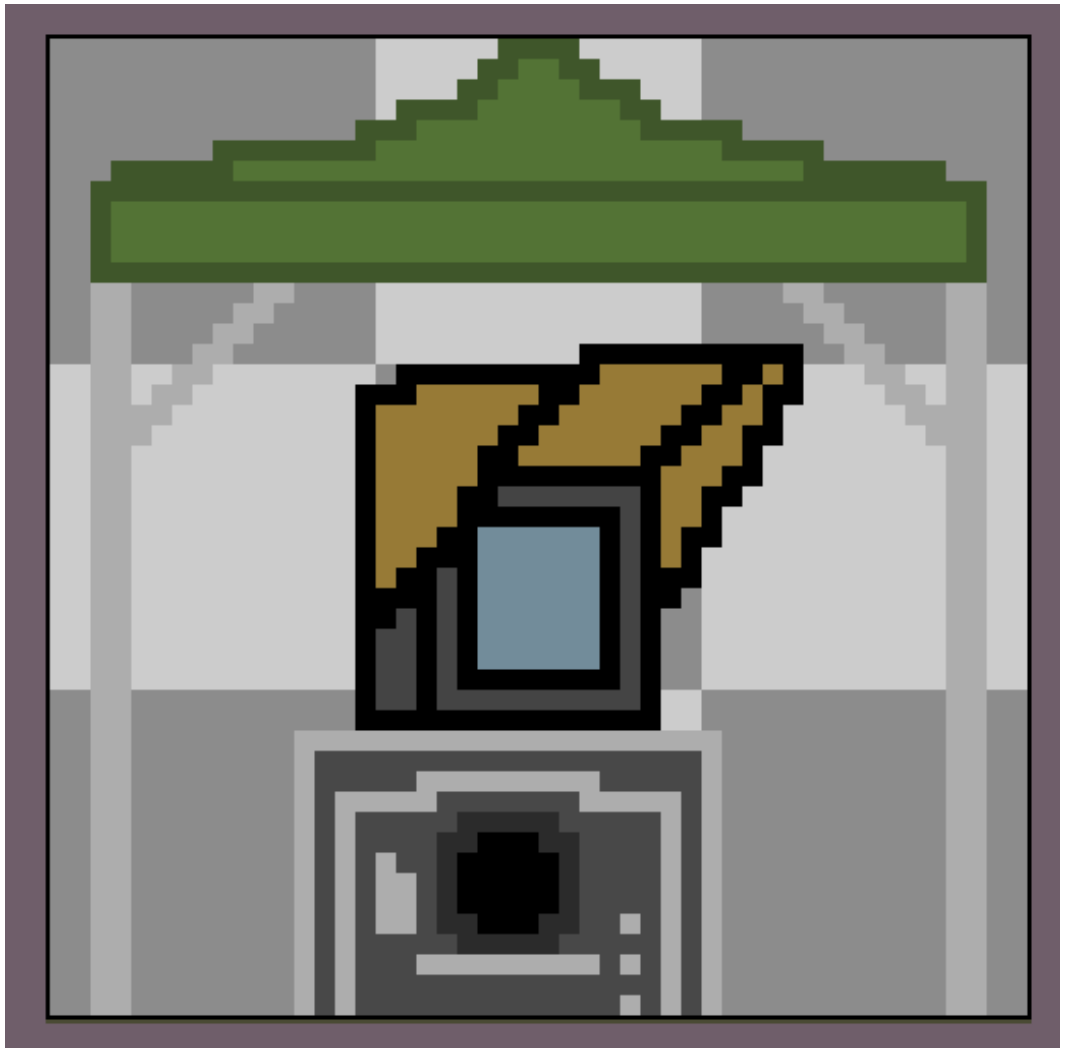


ENVIRONMENT ART – DAY 51

I STARTED TO ADD A GAZEBO FOR THE FOSSIL SCANNER WITH WOODEN SUPPORTS, I CHANGED THESE INTO METAL BEAMS AND EXPERIMENTED WITH THE SHEET THAT ACTS AS THE ROOF.

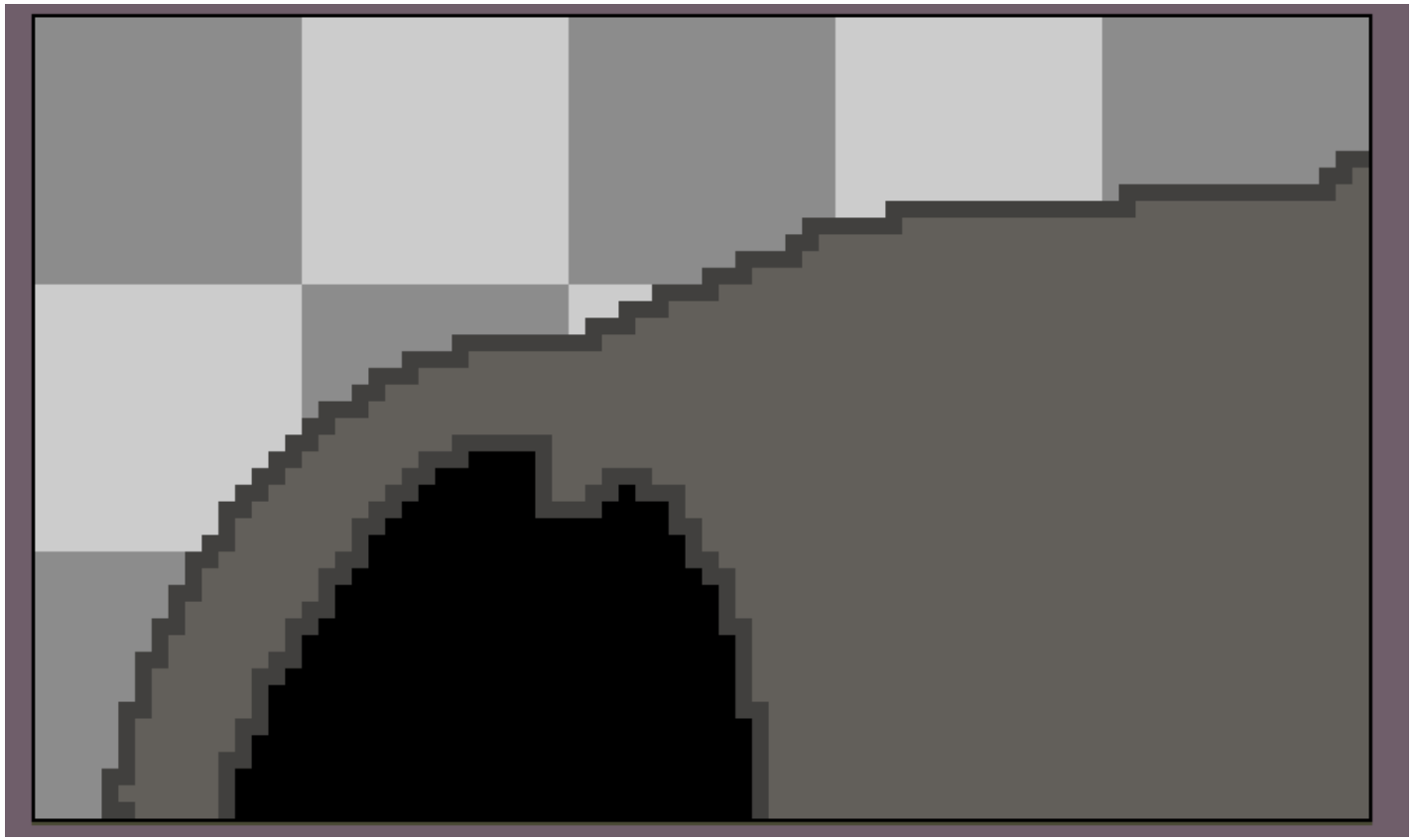
ENVIRONMENT ART – DAY 51

- I ended up settling with this design for the roof – concluding this art.



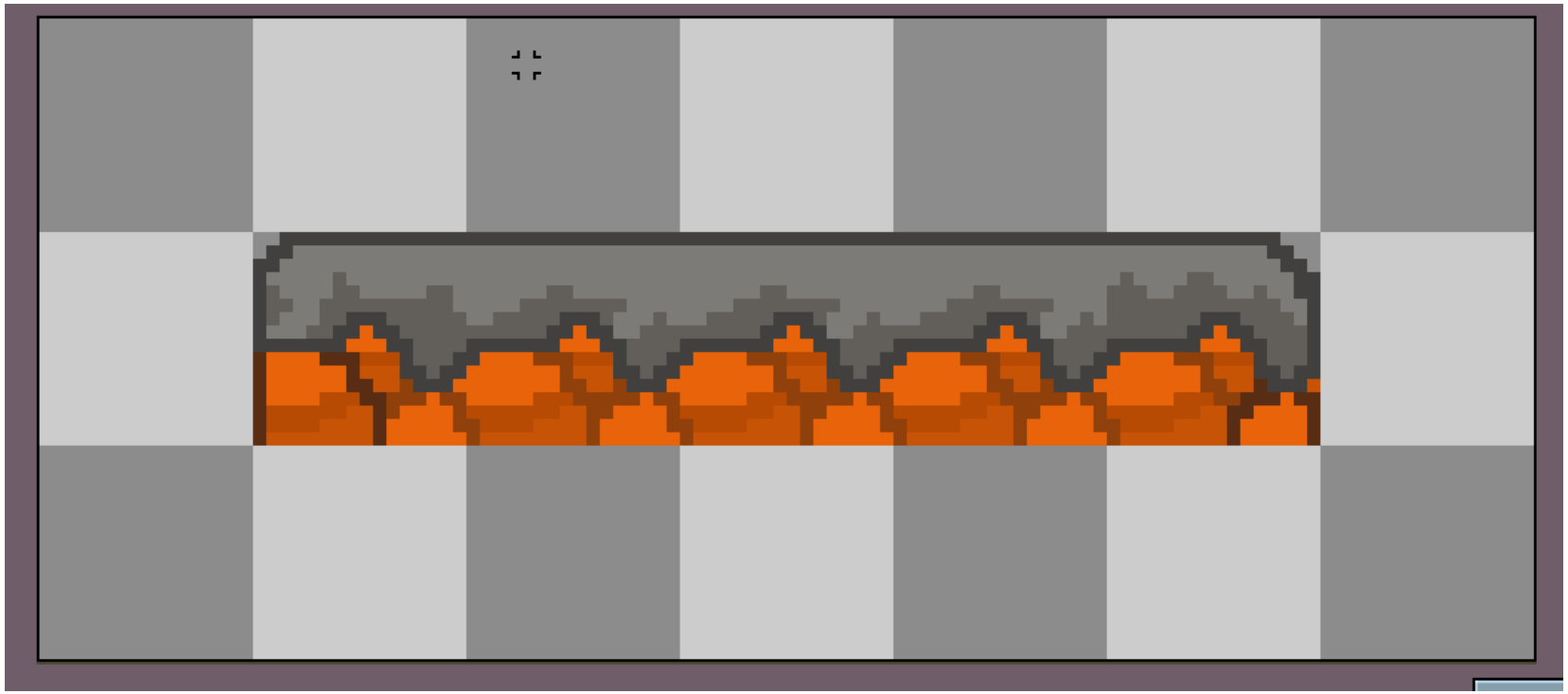
ENVIRONMENT ART – DAY 51

- I began the cave entrance but decided to stop doing it when thinking about implementing it into the game scene – I think an entrance like this doesn't make sense and wouldn't look good for a vertical descent into the cave, and the surface ground with the ladder will be the cave entrance.



ENVIRONMENT ART – DAY 51

- Instead of making the sand its own file, I returned to Durdle door and extended its canvas size to add in this beach using a sandy color and then a darker shade of that color to make the wet sand – where the tide comes in.

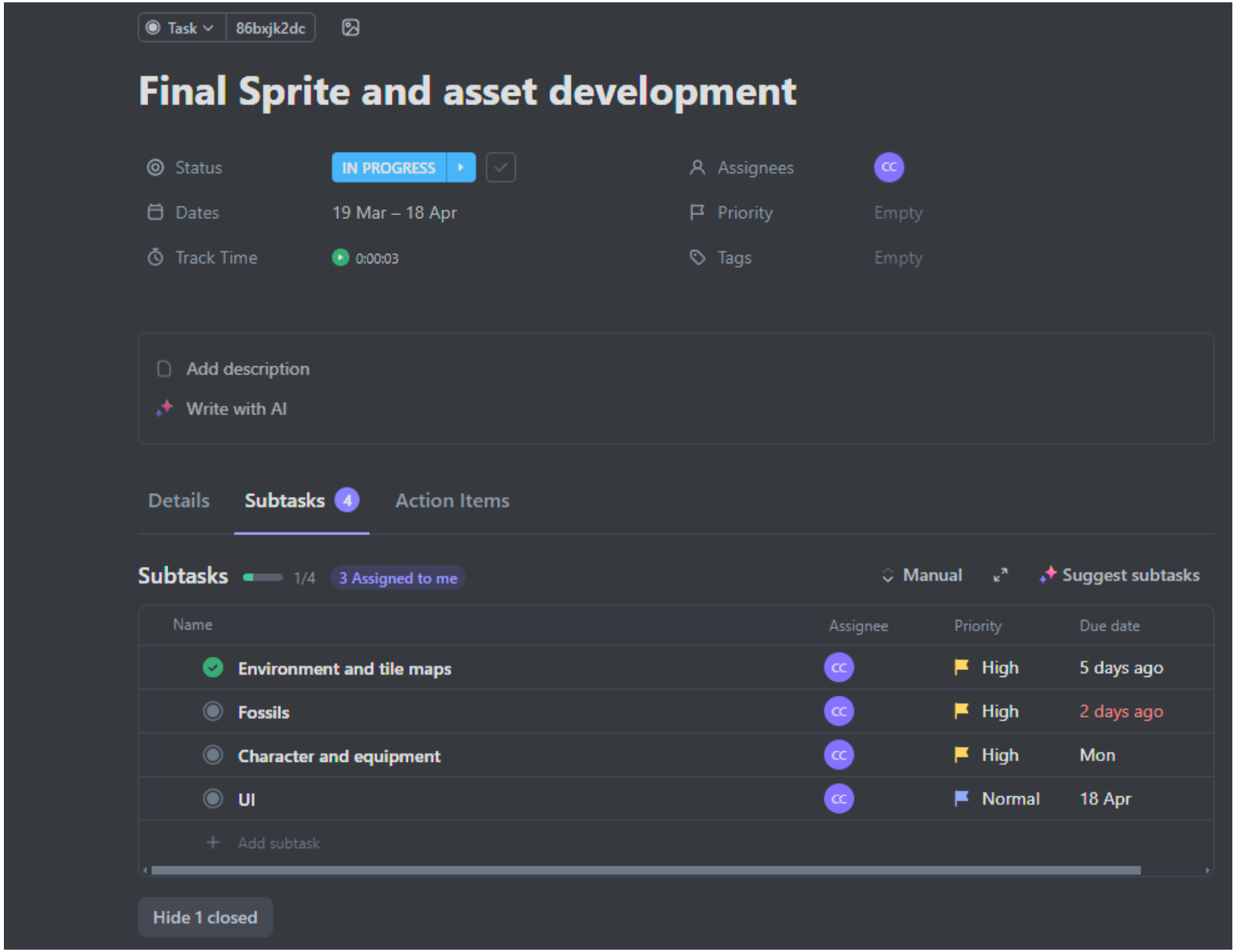


SURFACE TILE ART – DAY 51

- I completed the surface ground tile set with the corner tiles, adding the darker border to the top of all the tiles and adding the stony pattern onto the corner tiles.

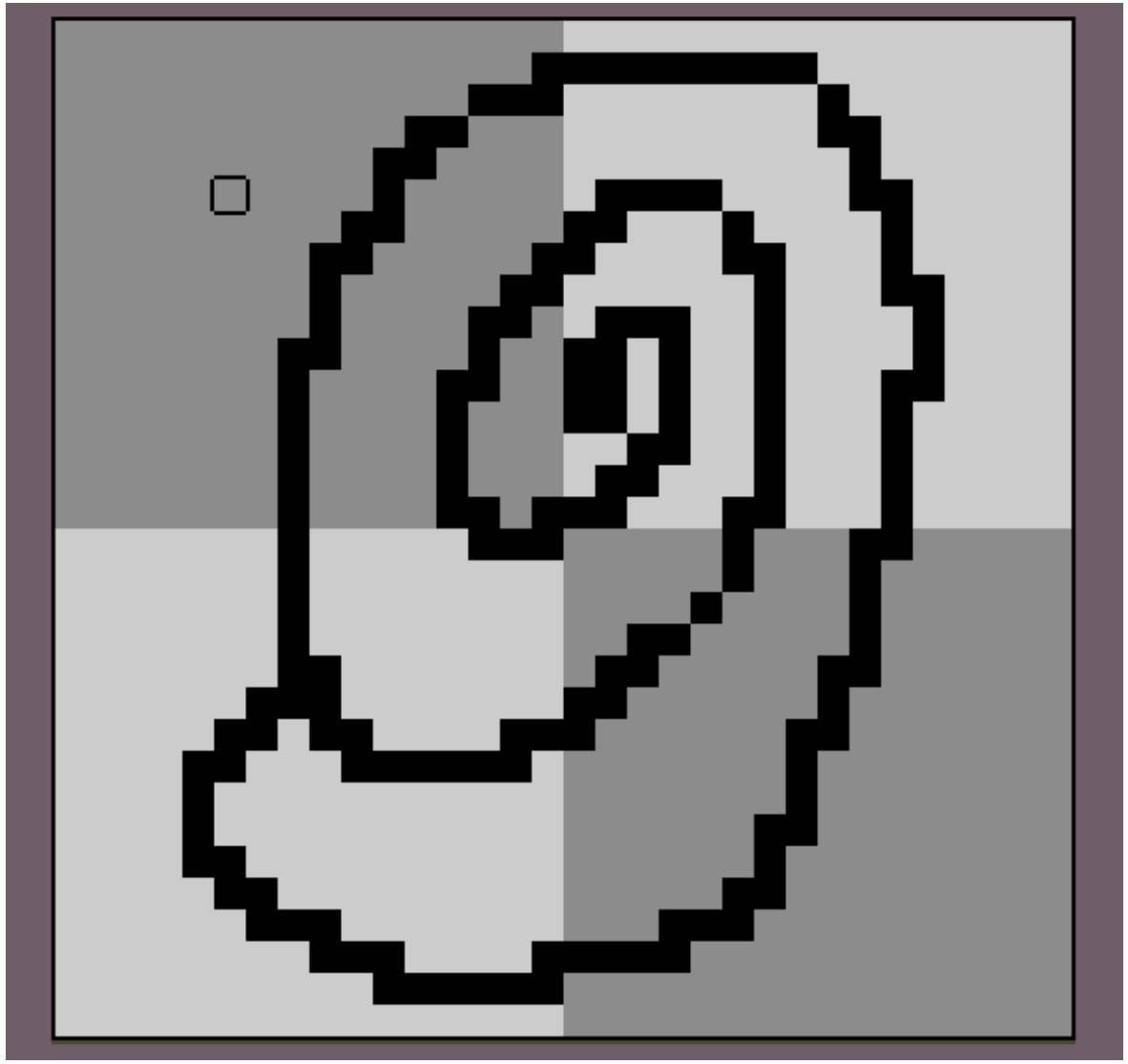
MILESTONE – DAY 51

- Now I can finally mark the environment art subtask as complete – although 5 days later than planned.
- Going forward from this overdue deadline, I now hope to finish the fossils by the 12th (Thursday, day 53) and be left with about 5 days to make the character – which I believe should be enough time.



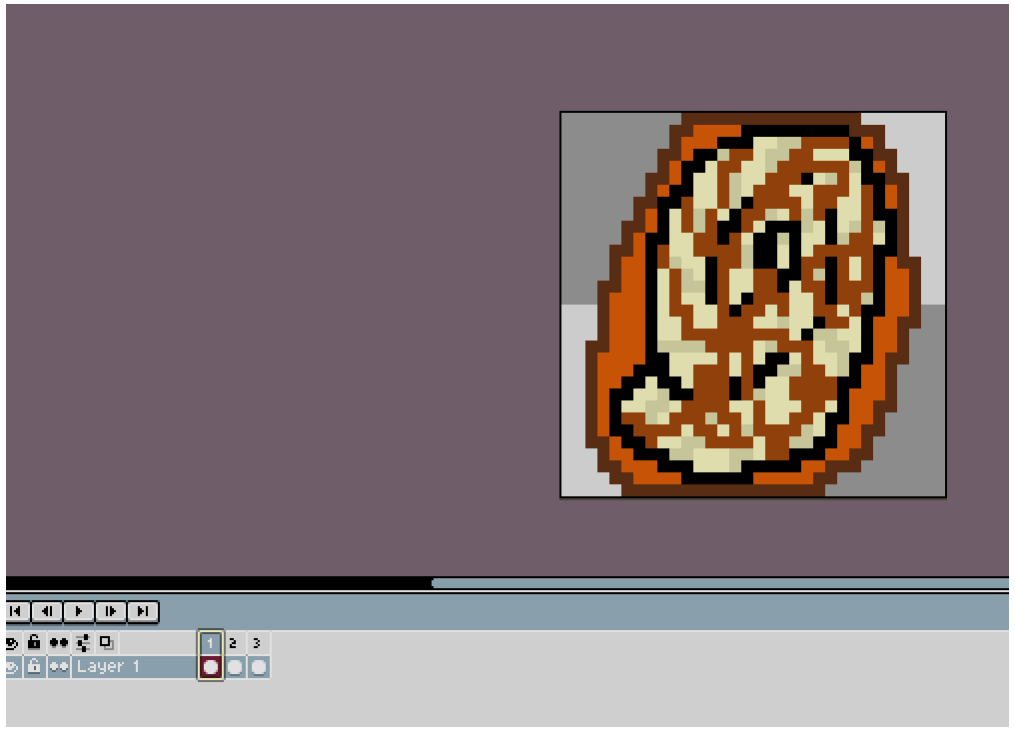
FOSSIL ART – DAY 52

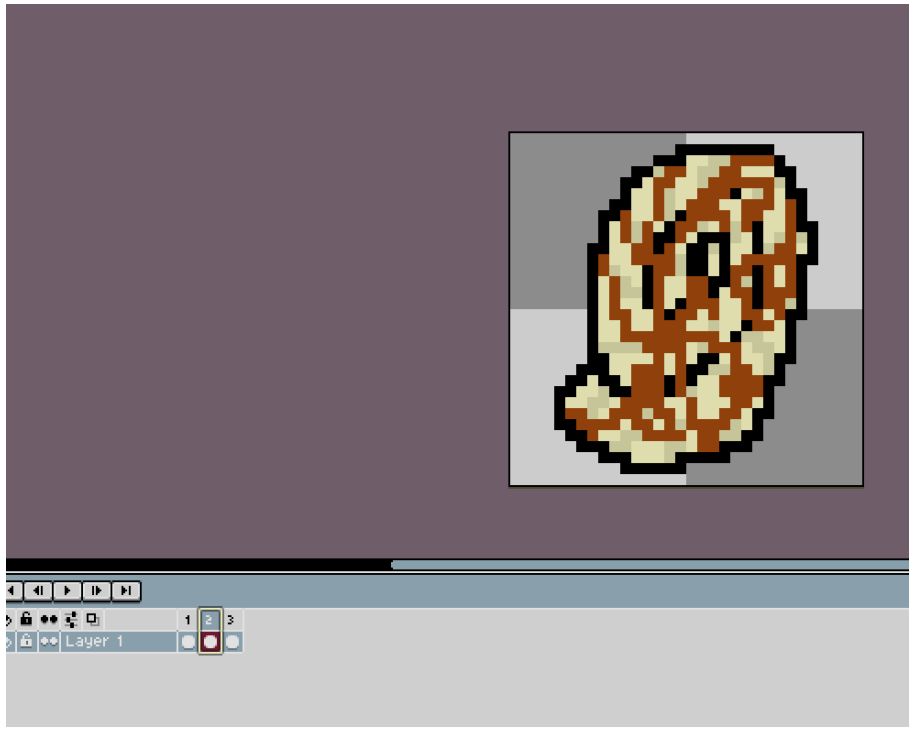
- I saved the outline for the Ammonite fossil as its own file, this will be the sketch of the Ammonite that will be added to the journal.



FOSSIL ART – DAY 52

- I filled in the colors and saved these changes as the fossil itself in a separate file.



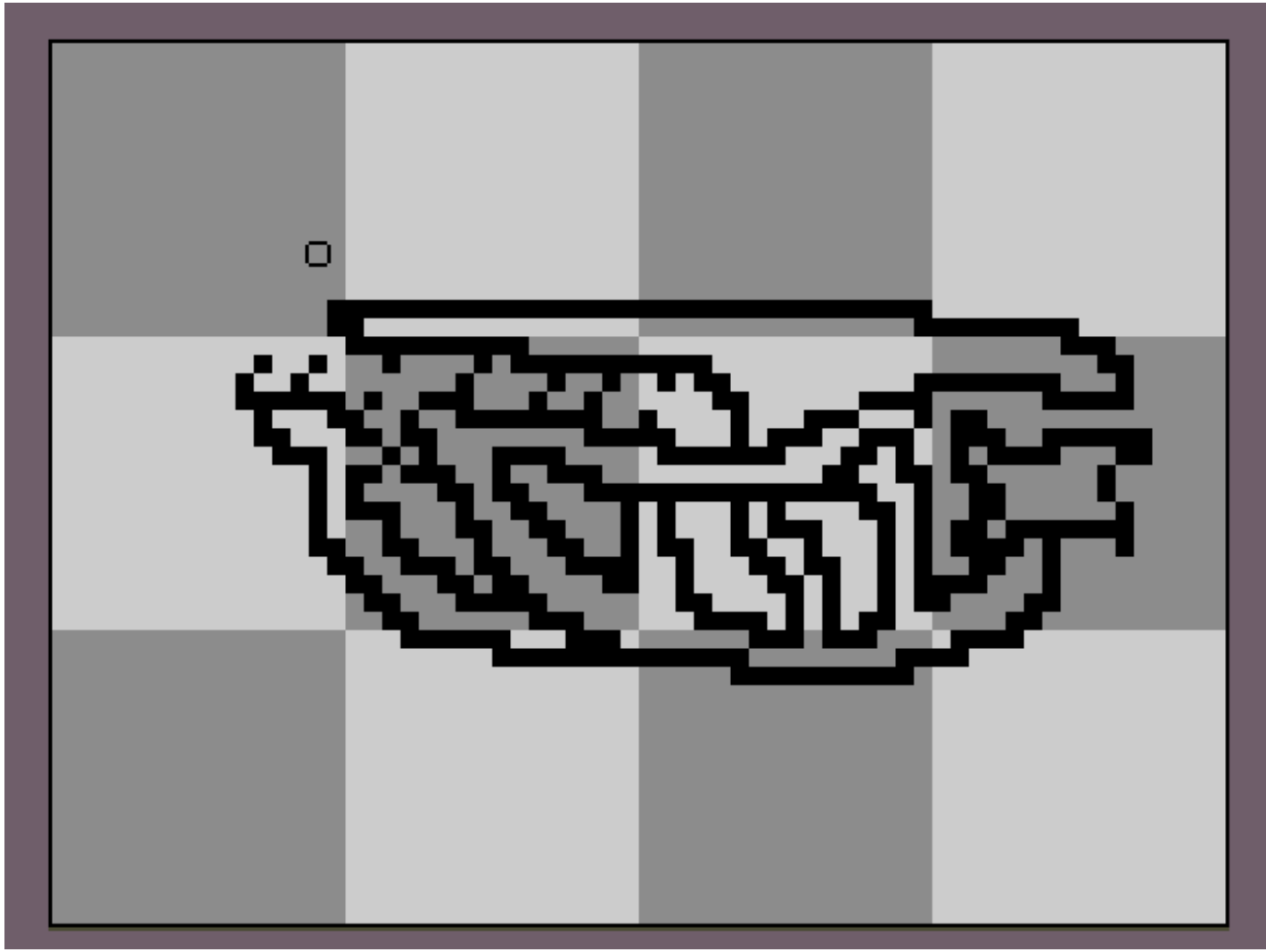


FOSSIL ART – DAY 52

- I then added the rock around the fossil and added in the excess rock over the top of it.
- I created three animation frames that each contain the different stages of rock removal – starting with all the rock, then removing the outer rock after using the pickaxe, then returning to the version shown on the previous slide after using the chisel.

FOSSIL ART – DAY 52

- I started the Coelophysis with its journal sketch – using the images from my mood board for reference.



FOSSIL ART – DAY 53

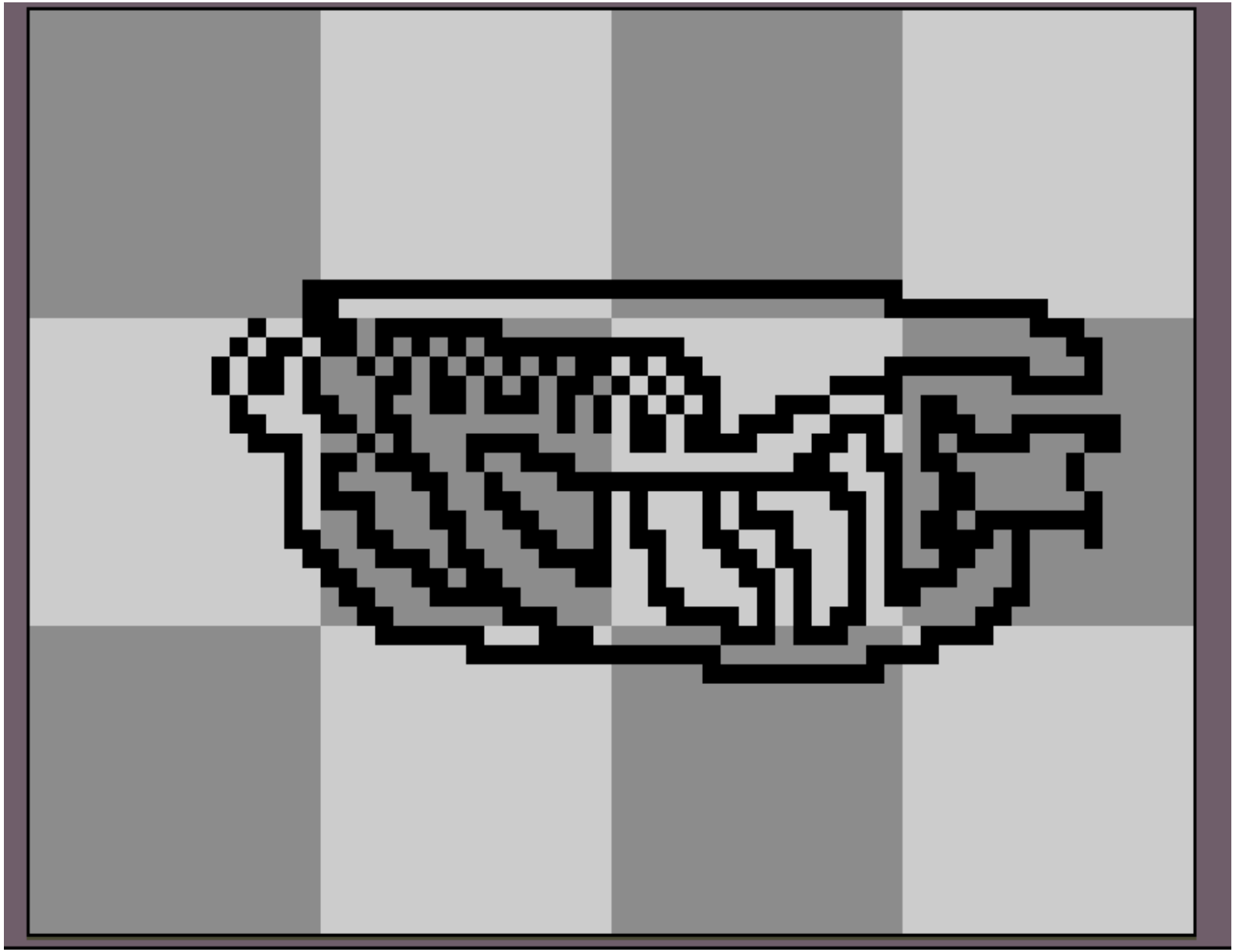
- I filled the outline in with the basic colors, and then worked on a bit of shading to create the final version of the fully excavated fossil.

FOSSIL ART – DAY 53

- I finished this fossil by adding in the excess rock and then the surrounding rock in different animation frames.

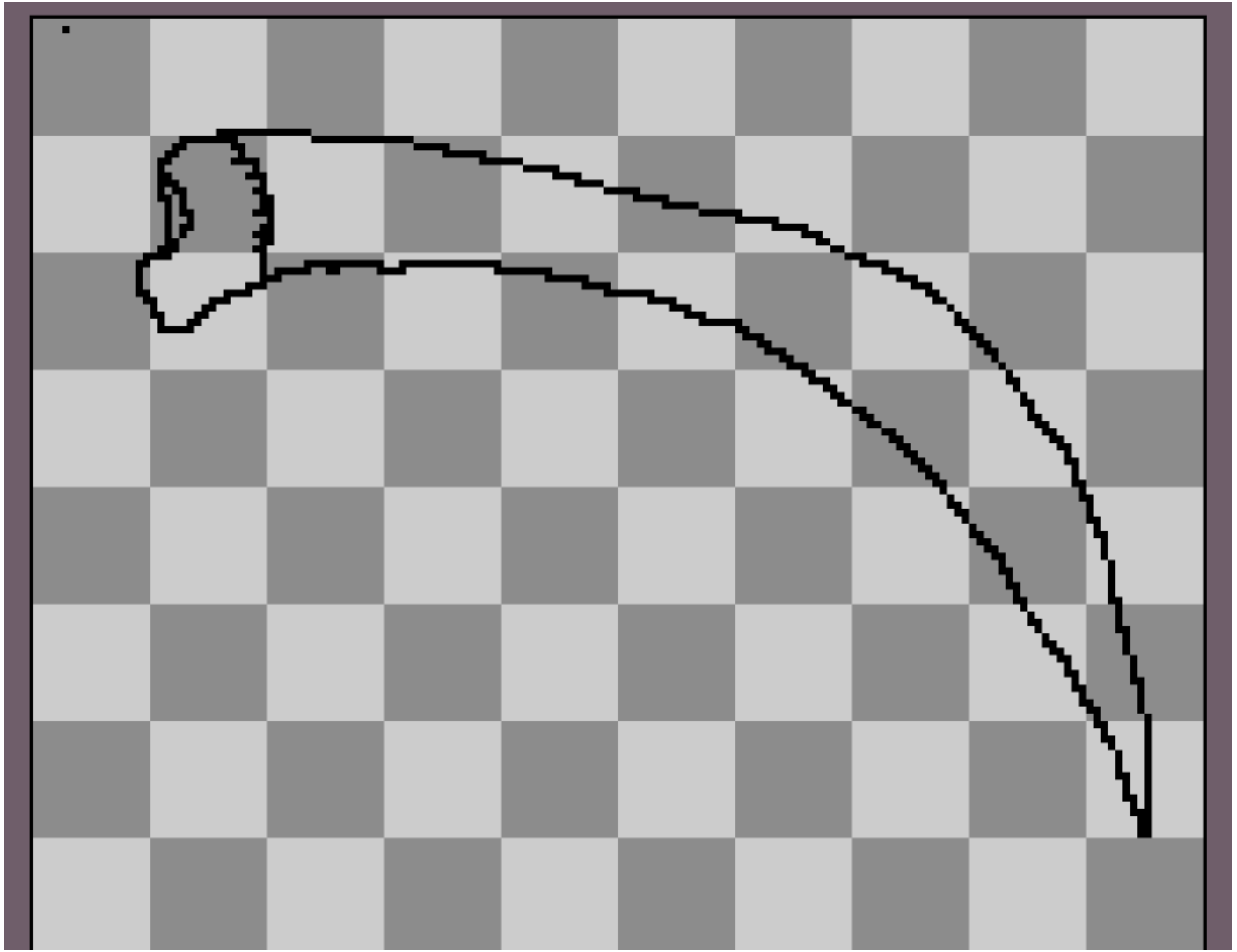
FOSSIL ART – DAY 53

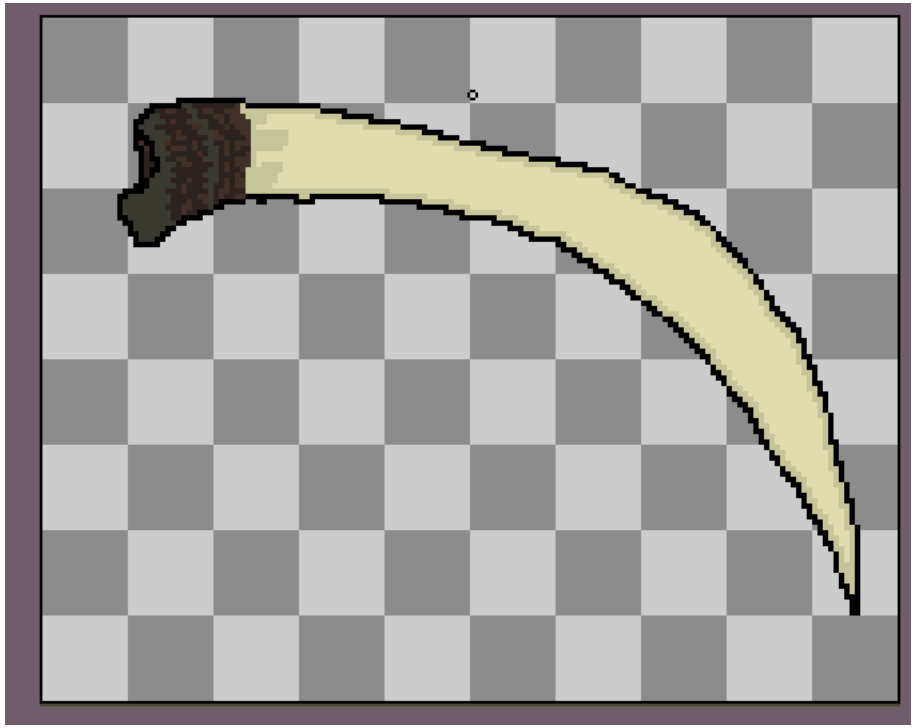
- I replaced the fossil sketch with the outline of the fully excavated fossil as I made some changes to the teeth of that compared to the original fossil sketch I saved.

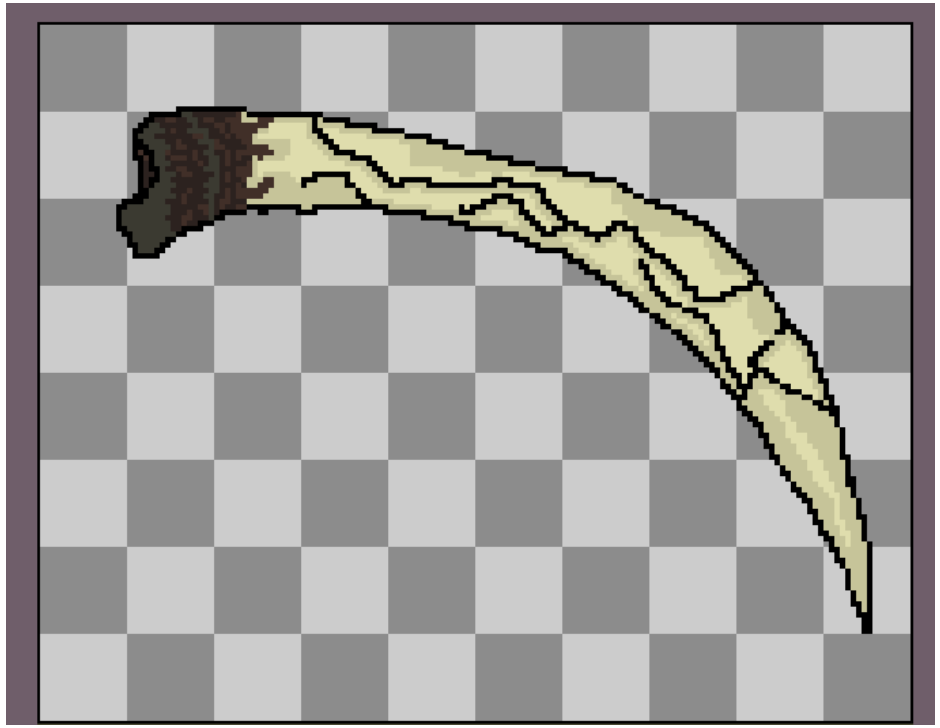


FOSSIL ART – DAY 53

- This is the fossil sketch for the claw of the Therizinosaurus – drawn with the pencil tool as always.







FOSSIL ART – DAY 53

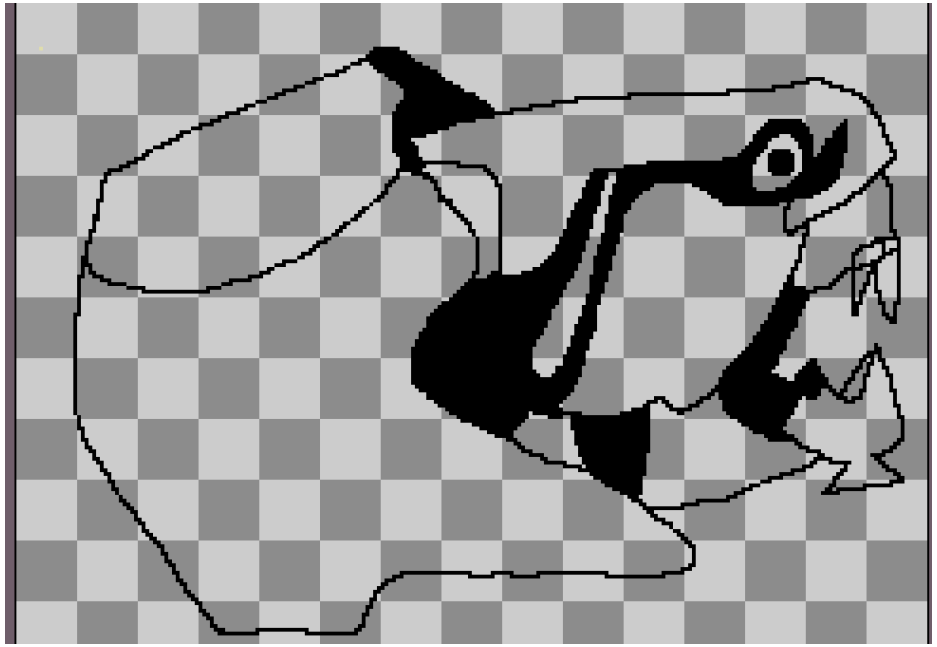
- I filled in the basic colors with the pencil and paint bucket tool as per usual, and then added cracks to the claw and gave them some depth by shading underneath them.

FOSSIL ART – DAY 53

- I finally added a significant amount of excess rock to the claw, as this will be a rarer fossil and require more chisel uses, and then surrounded it with the outer rock to complete each stage of the fossil.

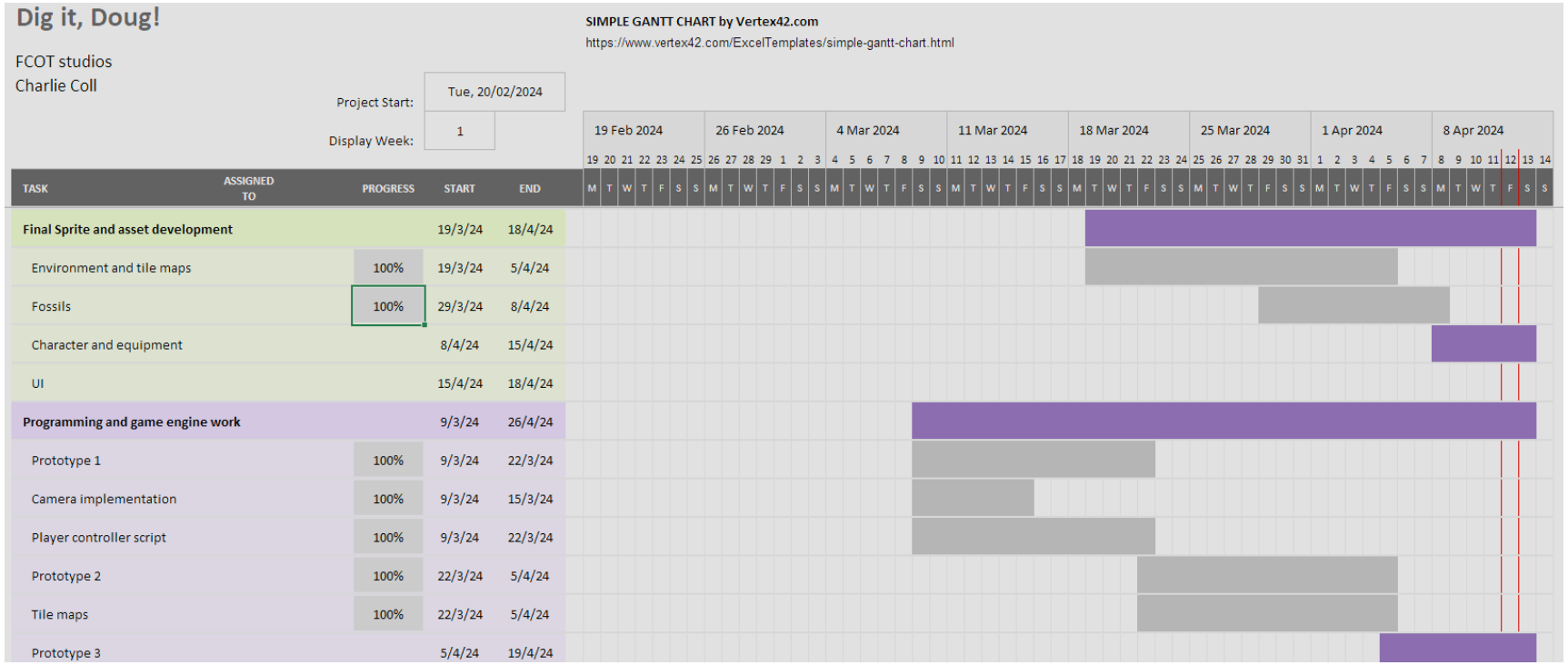
FOSSIL ART – DAY 53

- I made the Dunkleosteus with direct reference to my concept art of it, making the process a lot easier and faster, starting with the journal sketch – only consisting of black outlines and filled in areas.
- Following that, I filled in its basic colors.



FOSSIL ART – DAY 53

- I then went over the outlines with this lighter color, which is needed for the final fossil, but the journal sketch can only consist of the color black, so I originally drew this in that color.
- Finally, I added in the scratch/damage marks and the finer details to complete this fossil.



Dig it, Doug!

FCOT studios
Charlie Coll

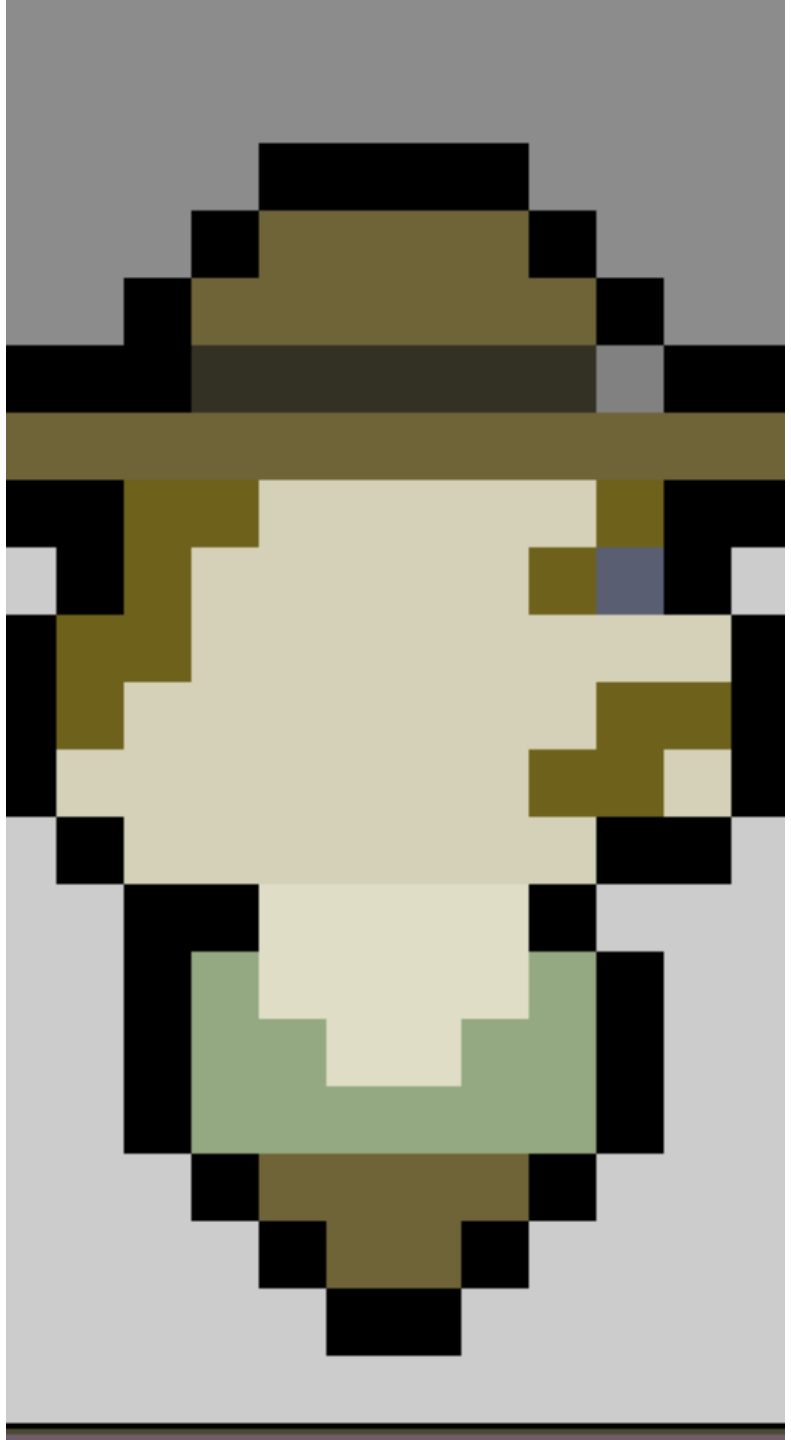
SIMPLE GANTT CHART by Vertex42.com
https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html

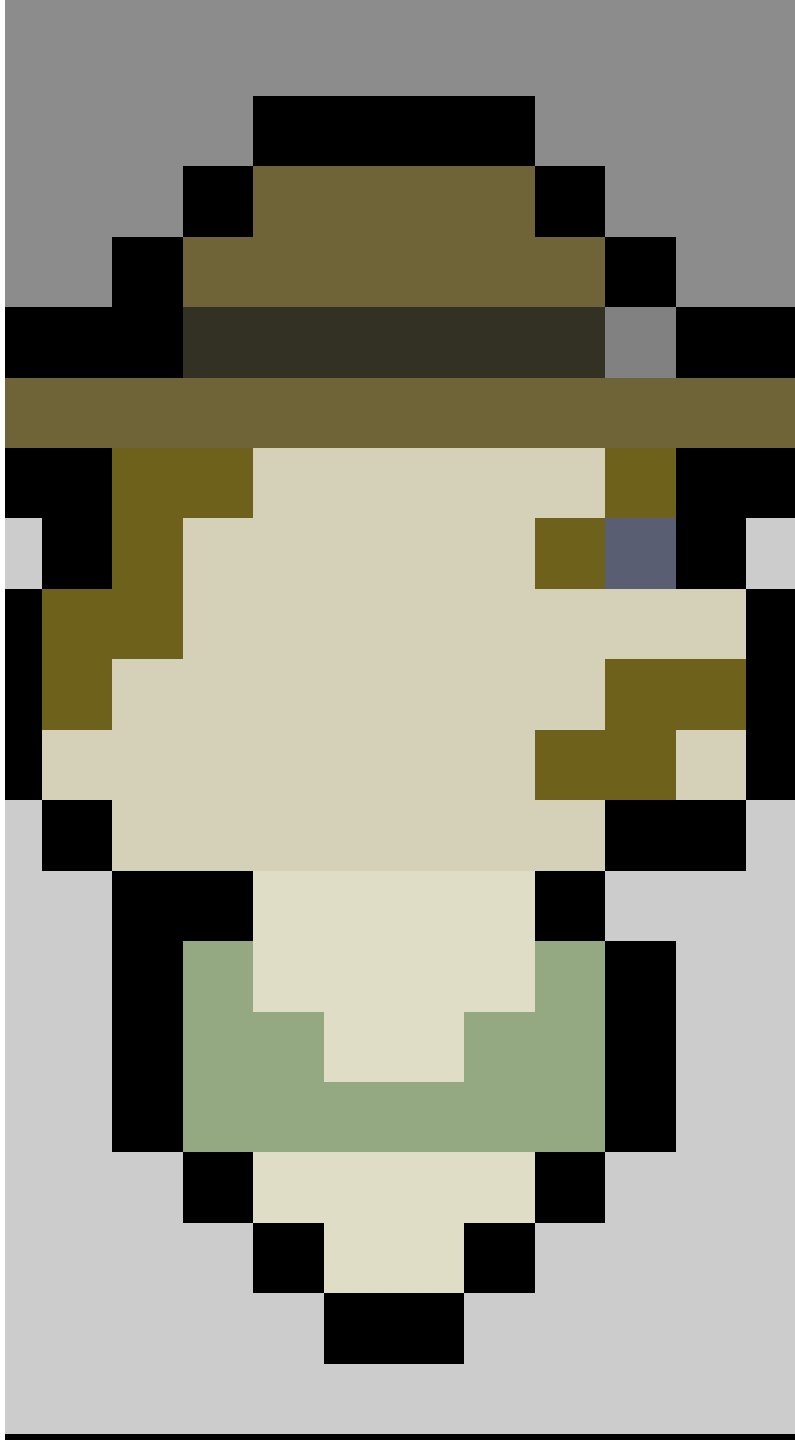
Project Start: Tue, 20/02/2024
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Final Sprite and asset development			19/3/24	18/4/24
Environment and tile maps		100%	19/3/24	5/4/24
Fossils		100%	29/3/24	8/4/24
Character and equipment			8/4/24	15/4/24
UI			15/4/24	18/4/24
Programming and game engine work			9/3/24	26/4/24
Prototype 1		100%	9/3/24	22/3/24
Camera implementation		100%	9/3/24	15/3/24
Player controller script		100%	9/3/24	22/3/24
Prototype 2		100%	22/3/24	5/4/24
Tile maps		100%	22/3/24	5/4/24
Prototype 3			5/4/24	19/4/24

MILESTONE– DAY 53

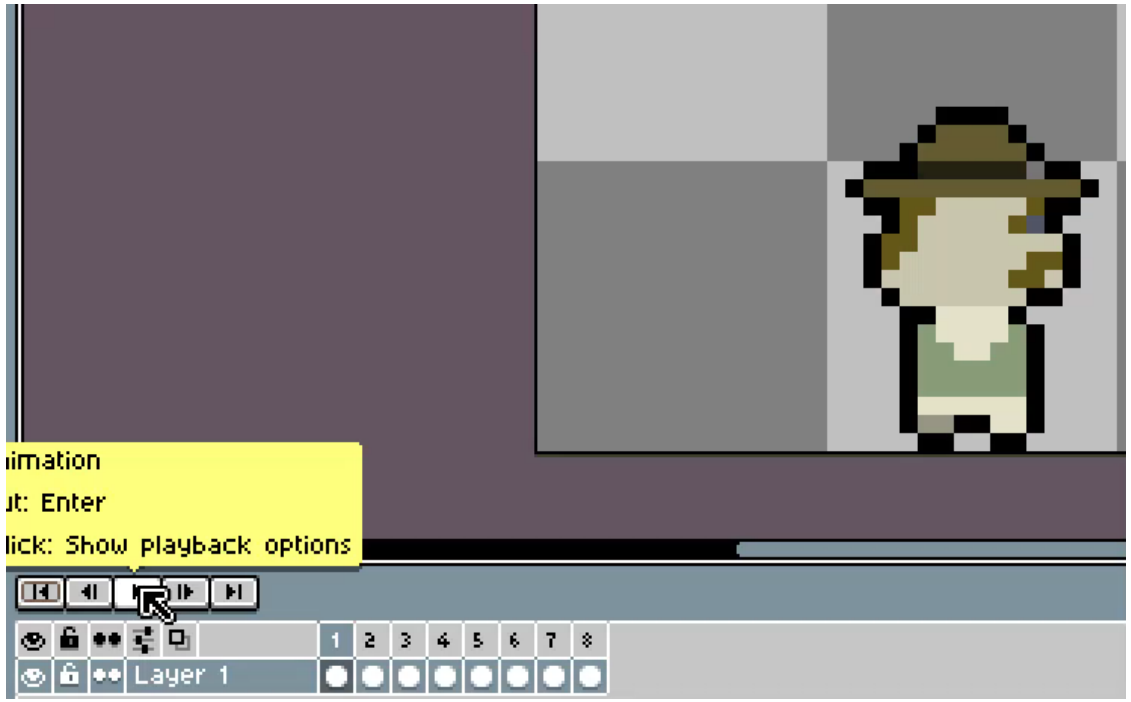
- I decided to mark the fossils subtask as complete and move on from making them, even though I've made four out of the six I'd hoped to make, as I can always make the remaining fossils if I have time after I've done the other essential tasks.





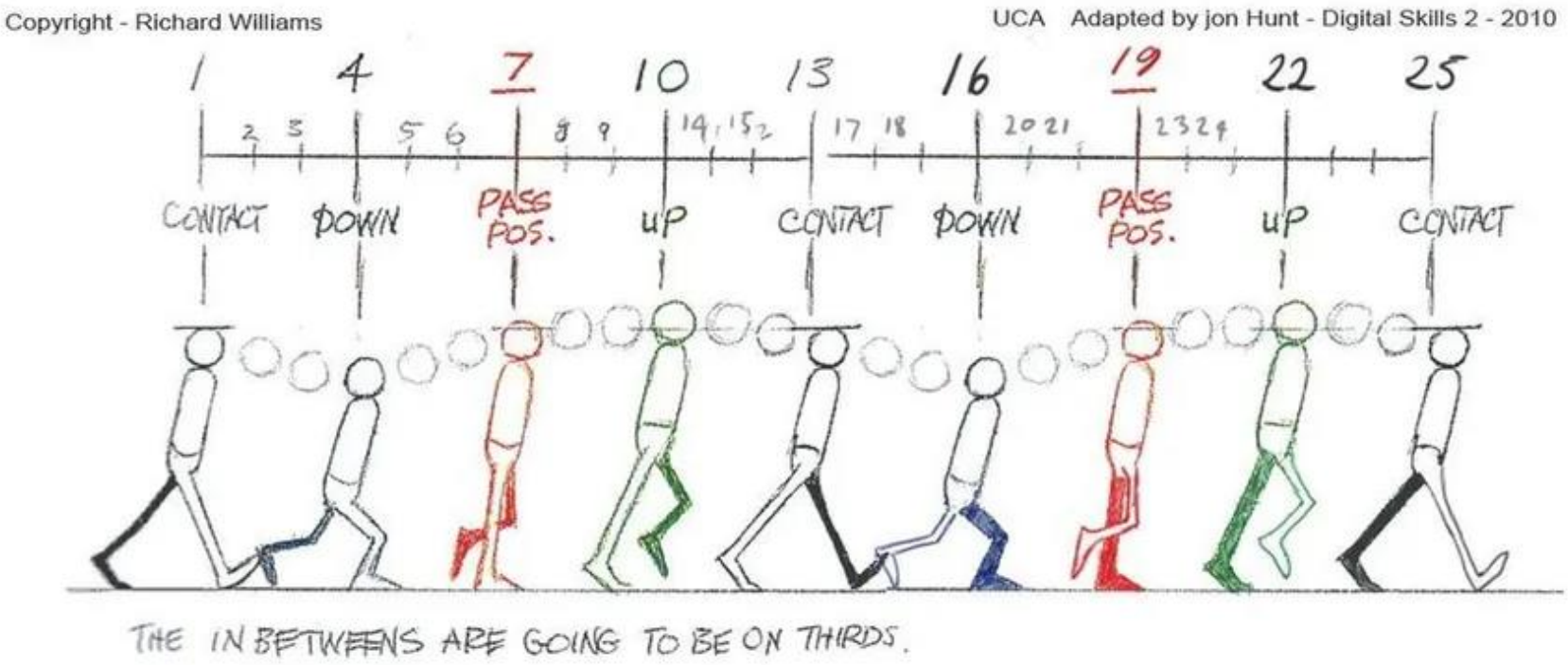
CHARACTER ART – DAY 54

- I started making this simple character from my concept, starting with the character without accessories and then trying out different colors for the trousers and adding the hat, the final version is on the far right.



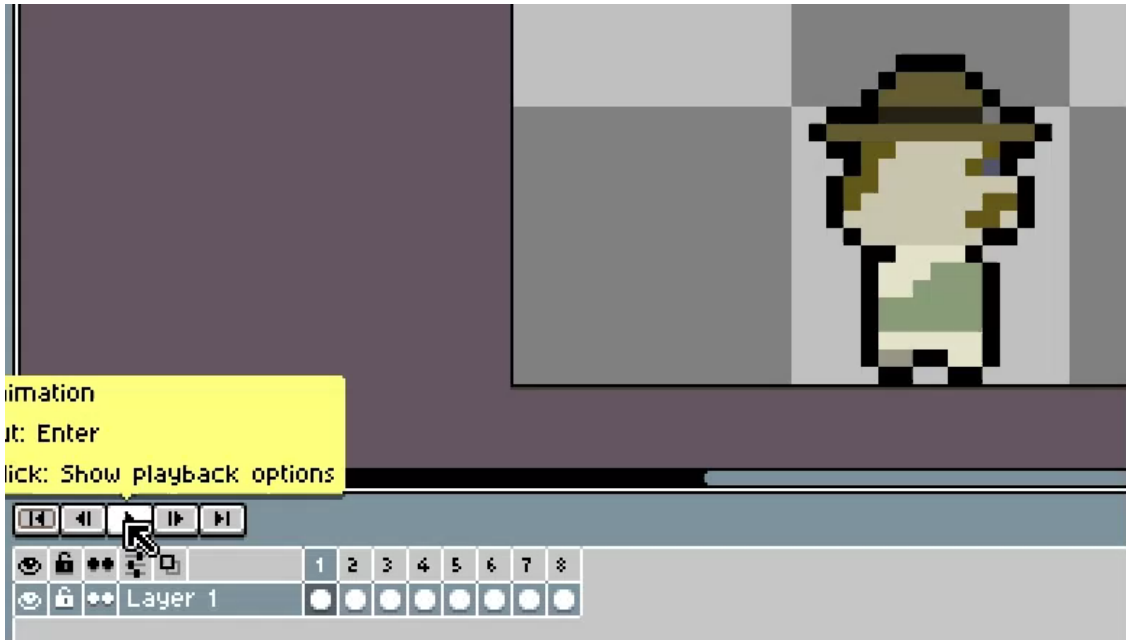
CHARACTER ANIMATION – DAY 54

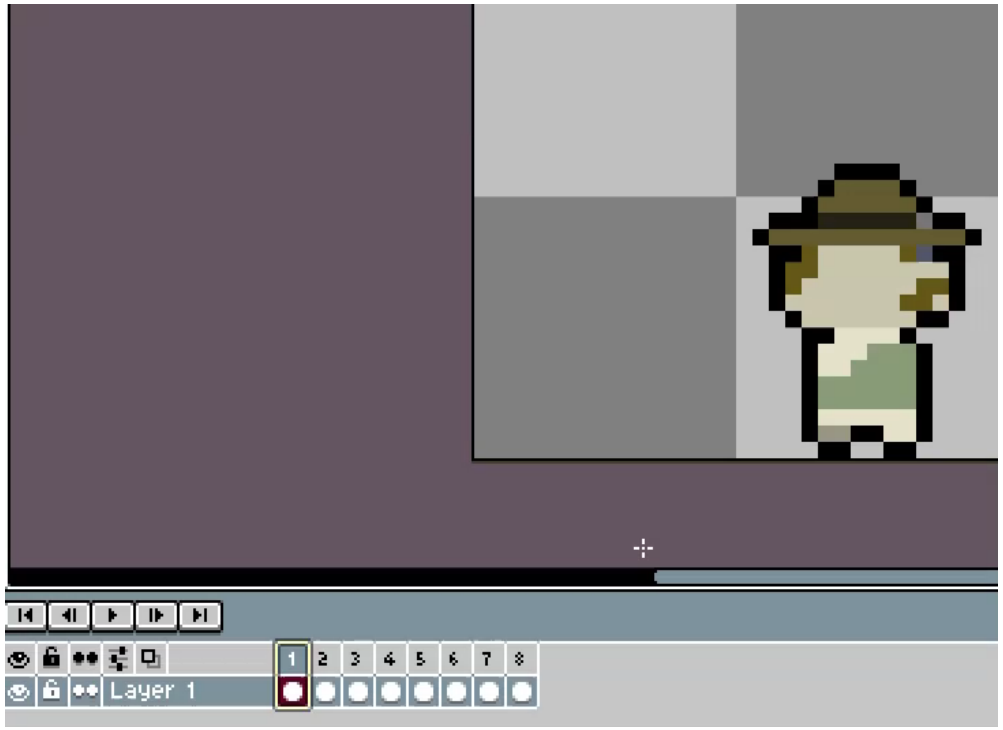
- To animate a walk cycle, I started by copying the character across 8 frames and moving it up and down on each frame according to the diagram shown on the next slide.
- Next, I animated the legs based off the diagram but in a much simpler fashion – using a slightly darker shade to represent the back leg.



ANIMATION – DAY 54

- The diagram I used to animate my own walk cycle, simplified to 8 frames in my animation, by Richard Williams – Animators survival kit.



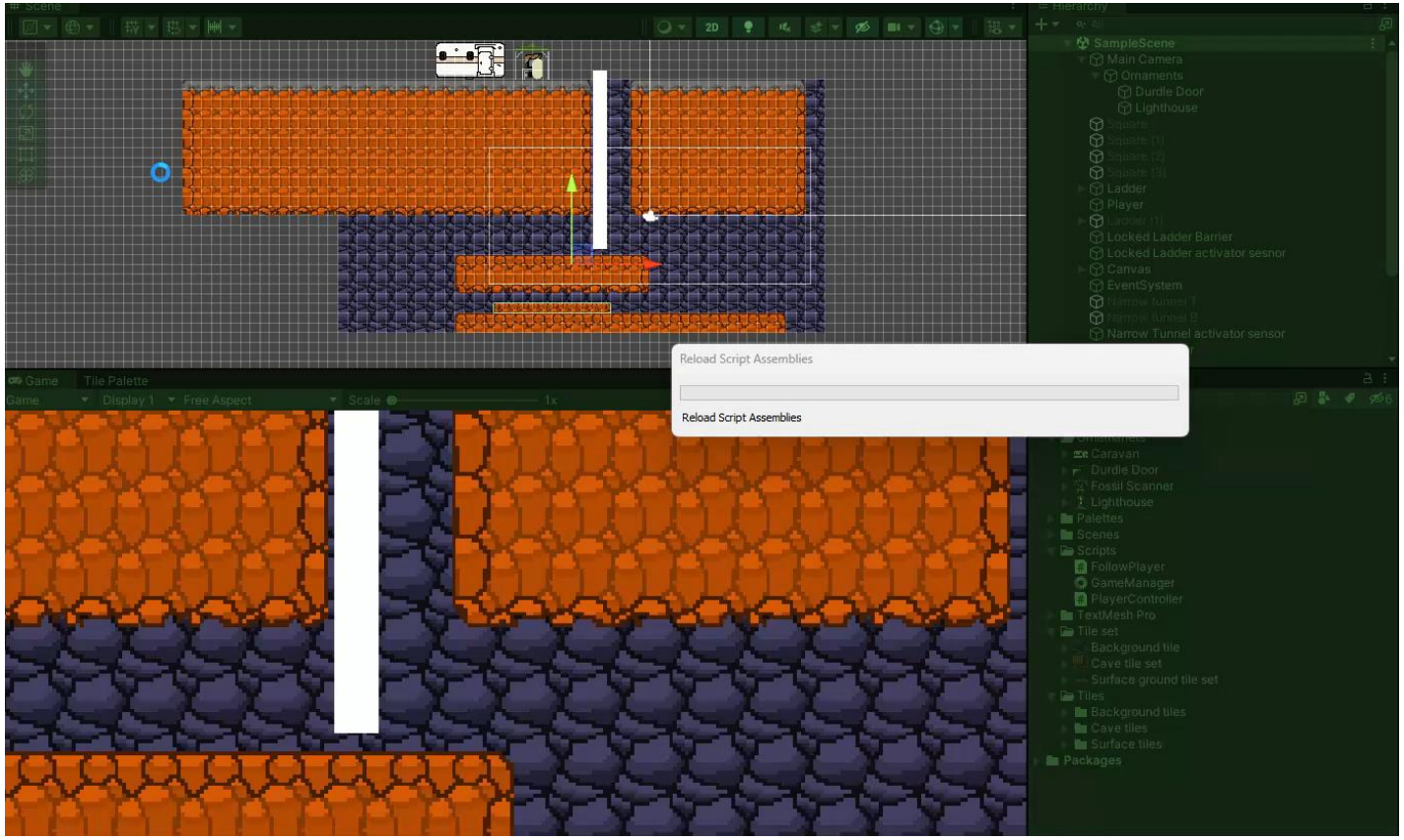


ANIMATION – DAY 54

- To animate the arms, I made them sway in the opposite direction to the leg that is on the same side of the character's body.
- I lowered the hat down the character's head by one pixel the frame after the character is at their highest point, giving it the effect that the hat is slightly bouncing on the character's head as they walk – shown on the right.

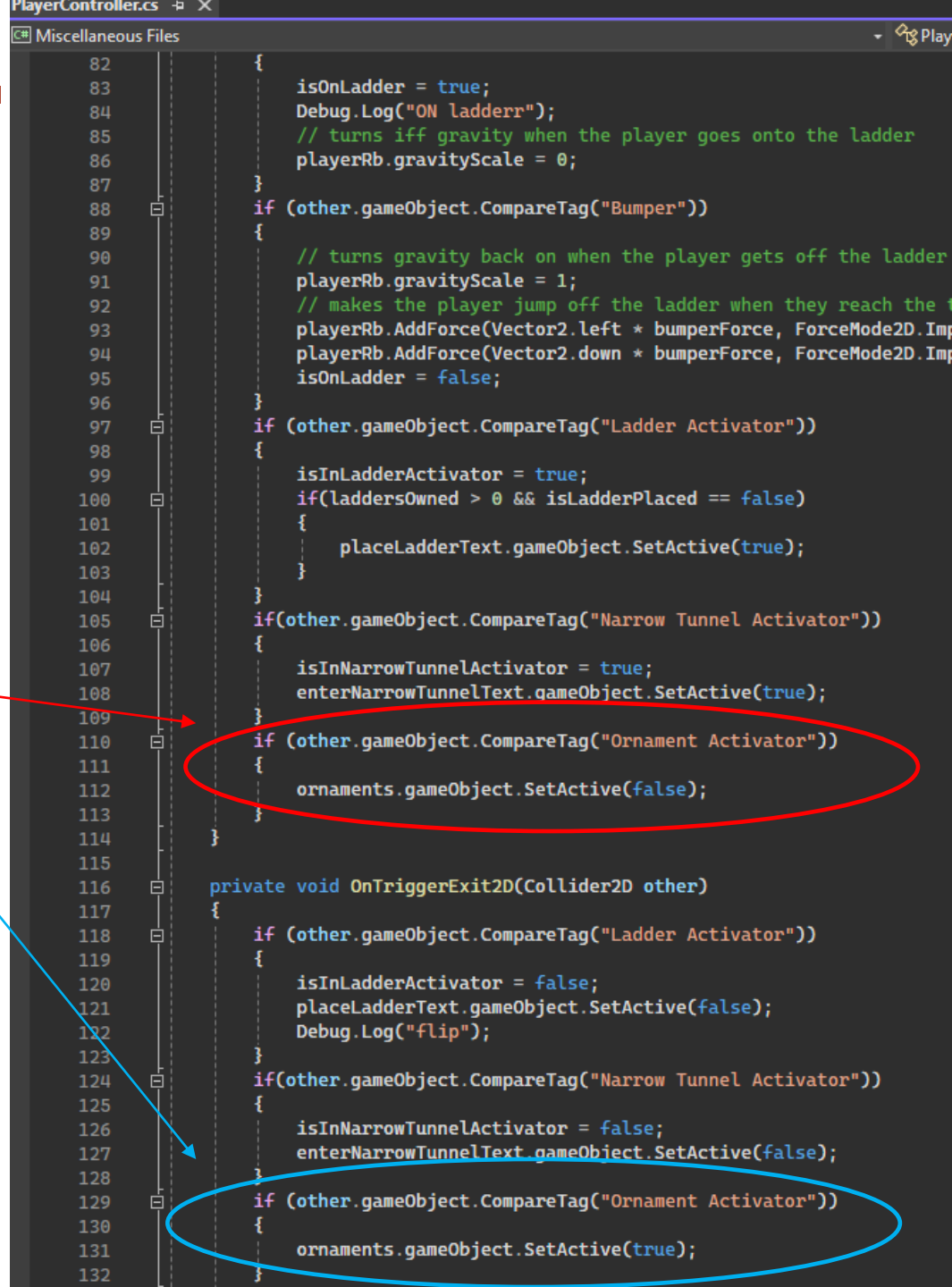
PROGRAMMING – DAY 55

- I implemented the environment ornaments (aesthetics) and replaced the top tiles with the tiles from the surface ground tile set.
- The caravan and scanner were dragged directly into the scene with a sprite renderer, whereas the lighthouse and Durdle door are child sprites of the main camera – once positioned they remain stationary, so the player doesn't walk past them, and they look as if they're in the background. The sprites attached to the camera are also sorted behind the tiles and other environment sprites in the layers system.
- I solved the issue of creating the narrow tunnel by creating a new tile map grid for it with half the scale of the other tile map grids, allowing me to place smaller cave tiles which became the bottom part of the narrow tunnel.



PROGRAMMING – DAY 55

- I added an is trigger box collider that surrounded the entirety of the undergrounds that sets the ornaments attached to the camera inactive when you enter it and active when you exit – meaning that the lighthouse and Durdle door sprites are only active when the player is in an area where they can be seen.

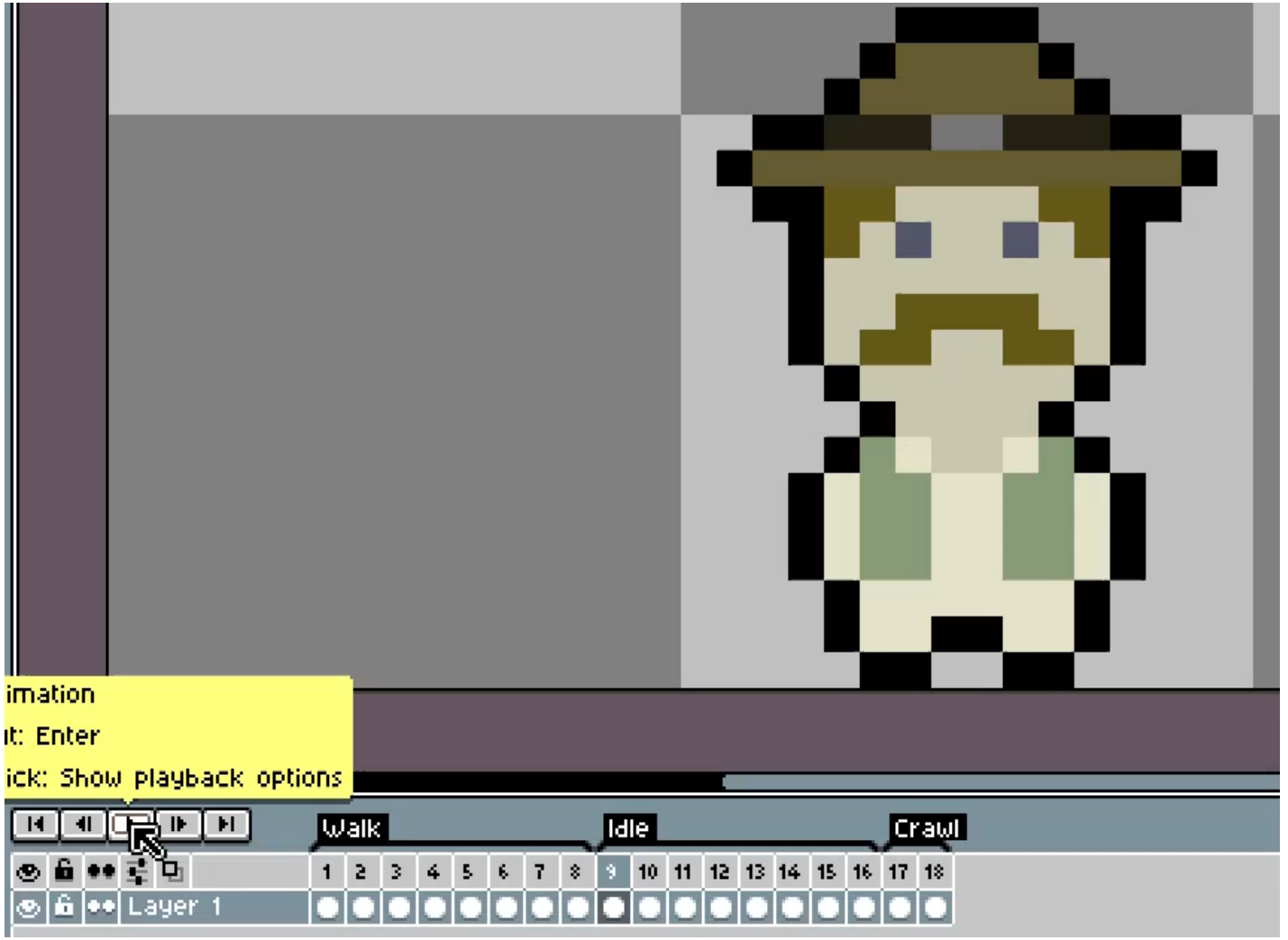


CONTENTS – WEEK 9

ANIMATION ART MILESTONE

CHARACTER ANIMATION – DAY 56

- I made this idle animation by taking one of the frames of the walk cycle and changing the shape of the head slightly, then changing the shirt and jacket to this angle, and moving the upper part of the body up and down over 8 frames.

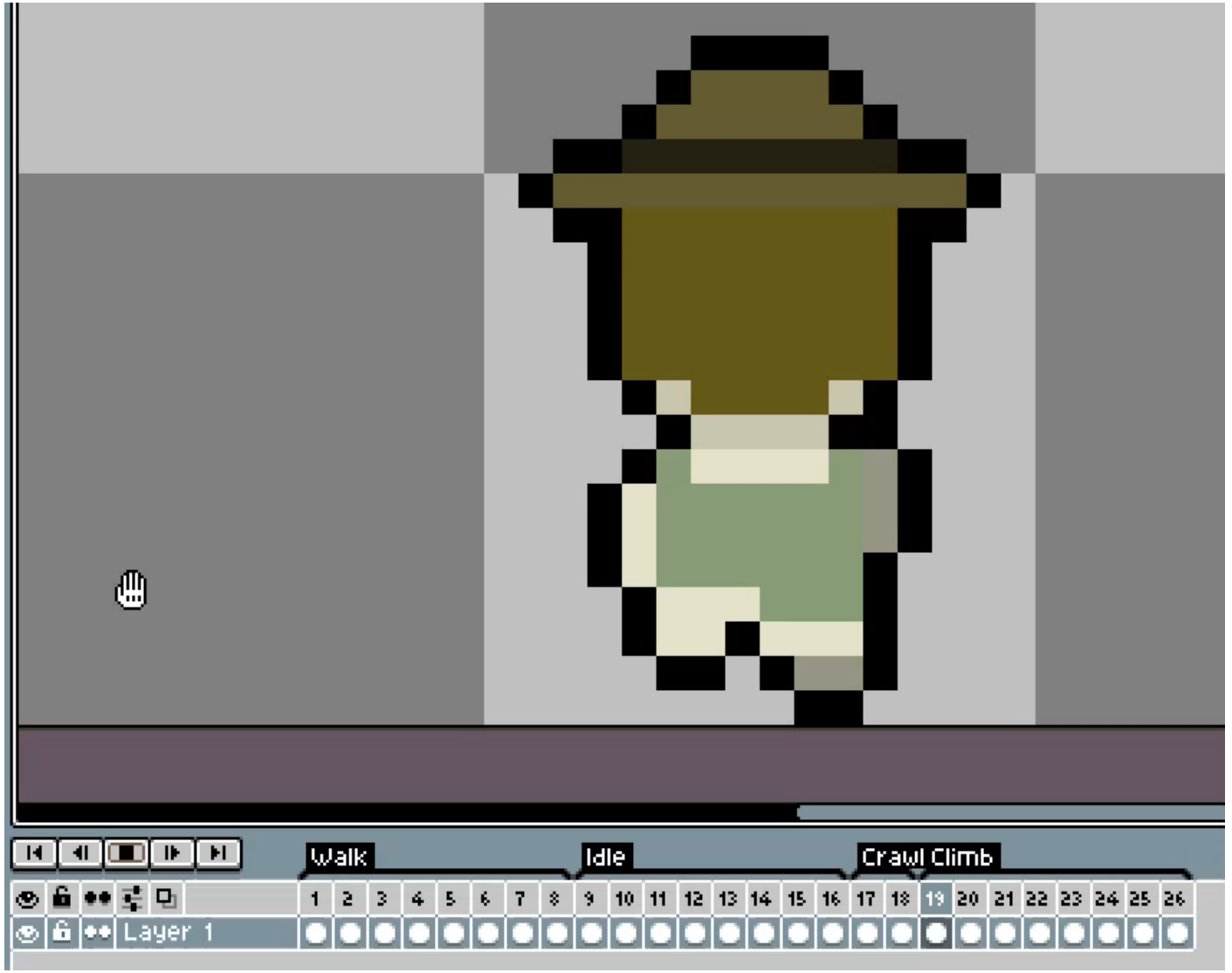


CHARACTER ANIMATION – DAY 56

- To make the crawl for the narrow tunnels, I first tried turning the limbs into squares but decided not to continue with that, I went with the legs of the walk animation and the arms of the idle animation in the end – alternating them up and down across 2 frames as this will only animate when the player completes a quick time event.

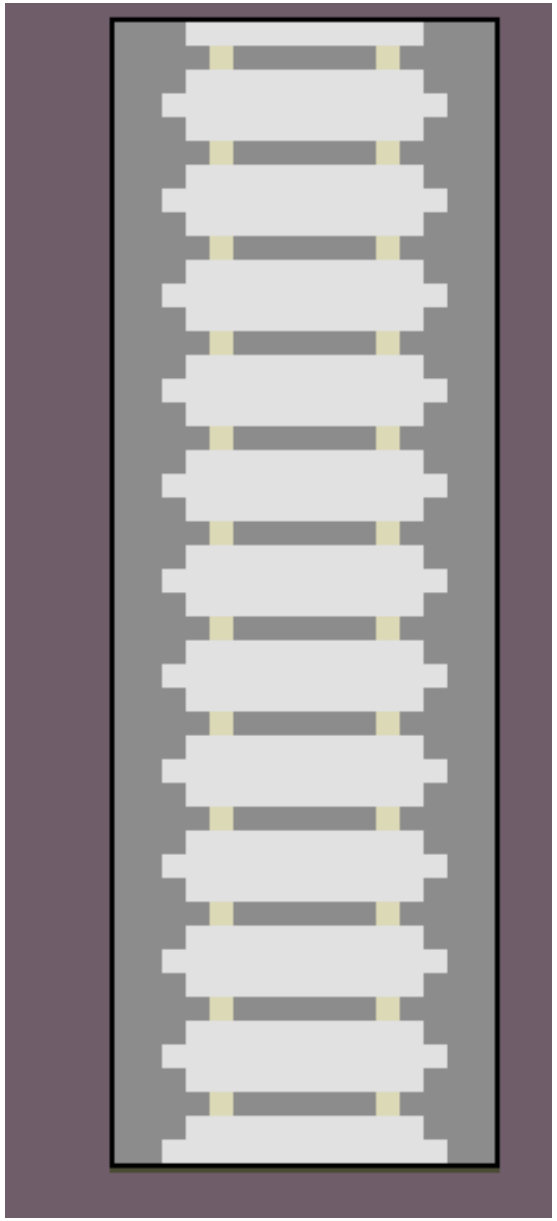
CHARACTER ANIMATION – DAY 56

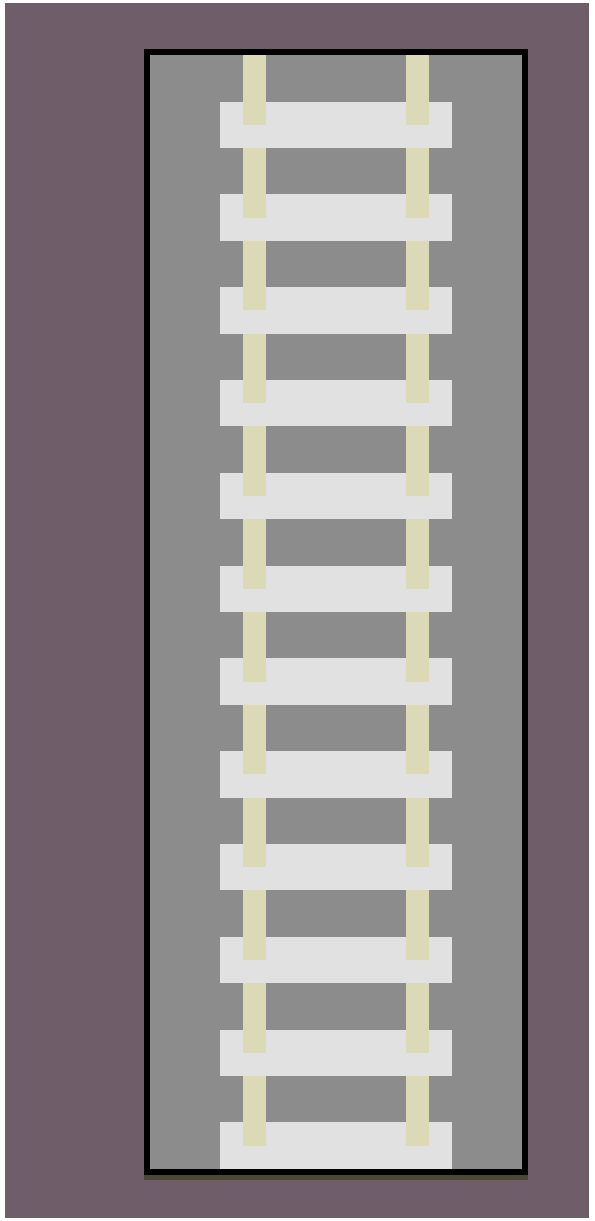
- I took the crawl animation and replaced the face with hair to change the angle to the back, then rotated the character back to being upright to create the animation for climbing ladders.



EQUIPMENT ART – DAY 56

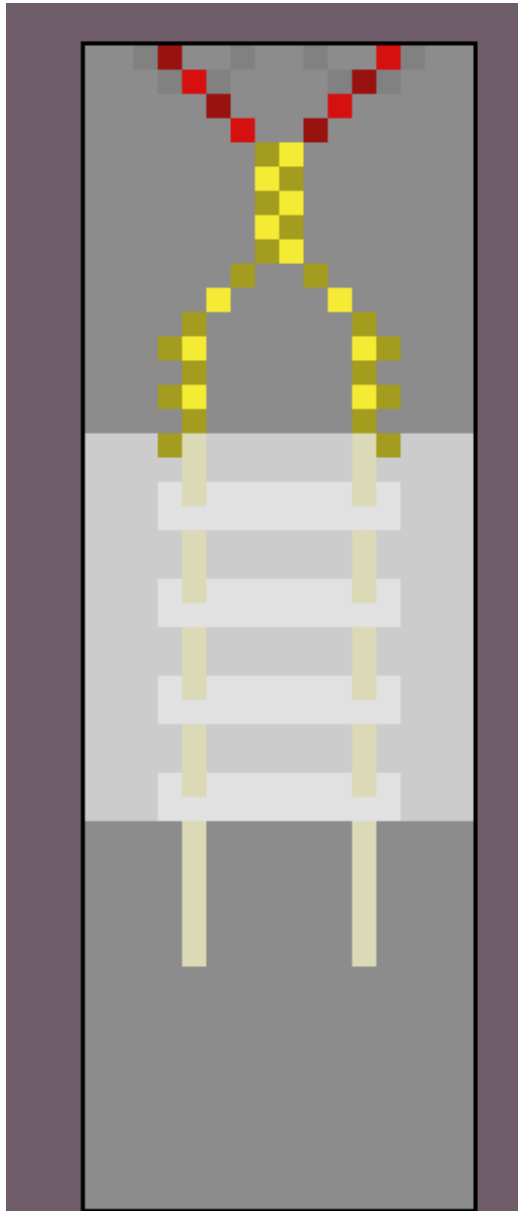
- I took a break from the character to make the ladders in tiled mode – with the first iteration on the left, and the final version on the right after making the ruts shorter and passing the rope through the top of them to give depth to the ladder.

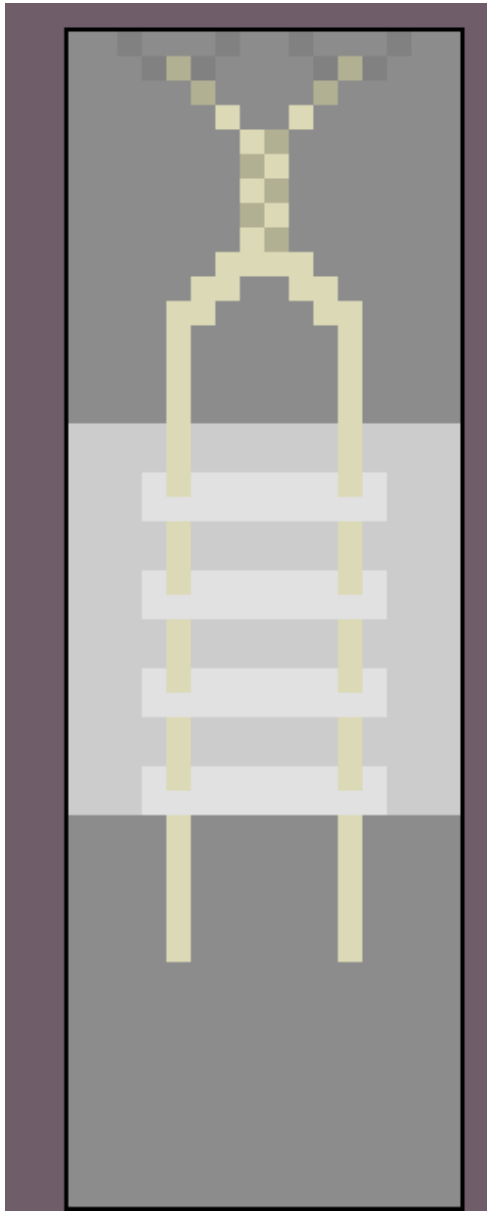




EQUIPMENT ART – DAY 56

- I made the ladder into a small tile set with the bottom piece being just hanging rope and the top piece being attached to clamps – I tried an industrial look first but then decided to switch this to the rope that the ladder is already made of to maintain consistency.



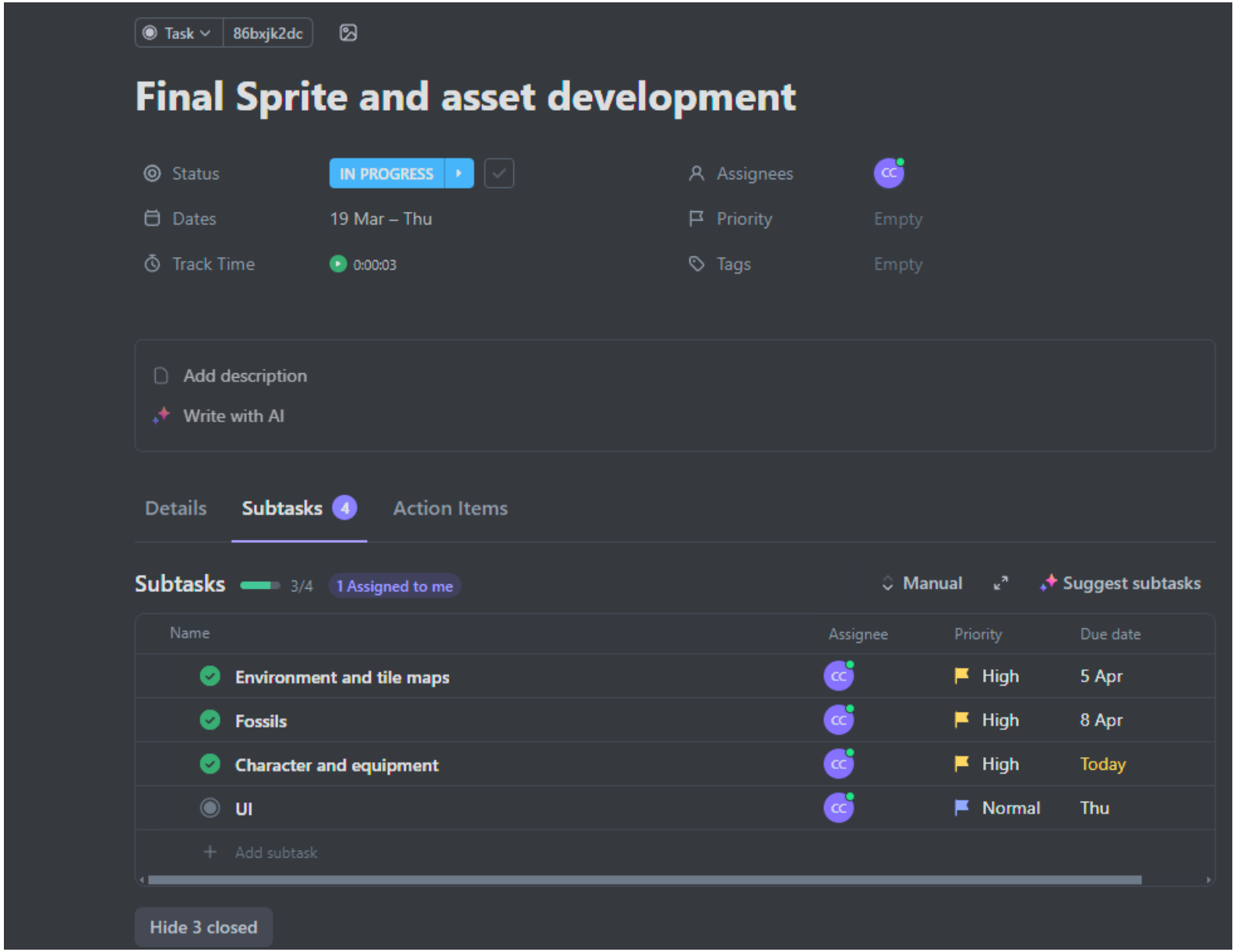


CHARACTER ANIMATION – DAY 56

- The final character animations were these tool actions – I animated the pickaxe with an anticipation movement followed by a large trail for the swing of the pickaxe, then the chisel was a simple jousting movement via the arm moving to the right and back.

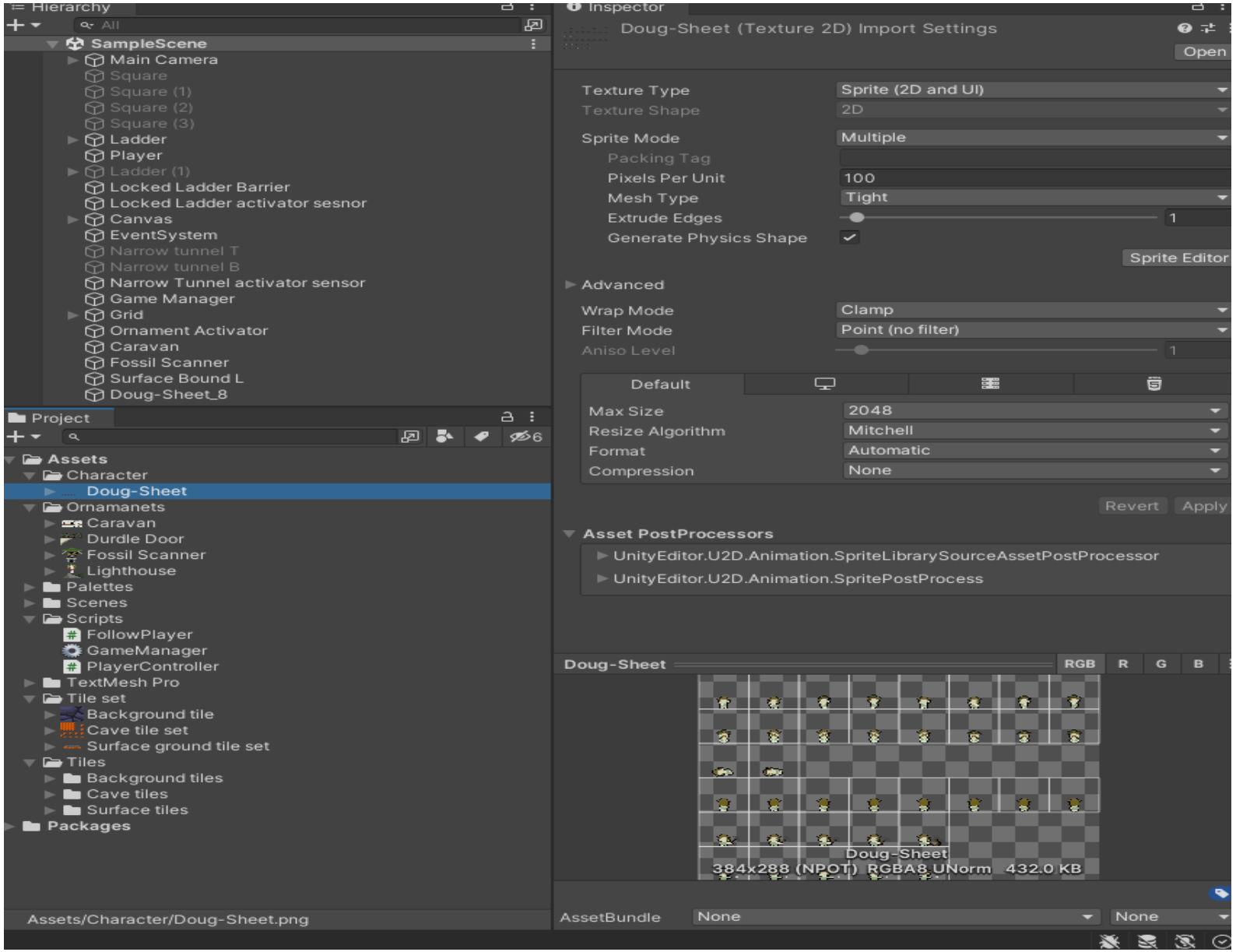
MILESTONE – DAY 56

- I met the deadline for the character and equipment subtask on time – cutting some corners with simple designs and animations to make sure I produce something complete and don't spend this remaining time perfecting minute details, but still a good motivator for the final stretch of the project



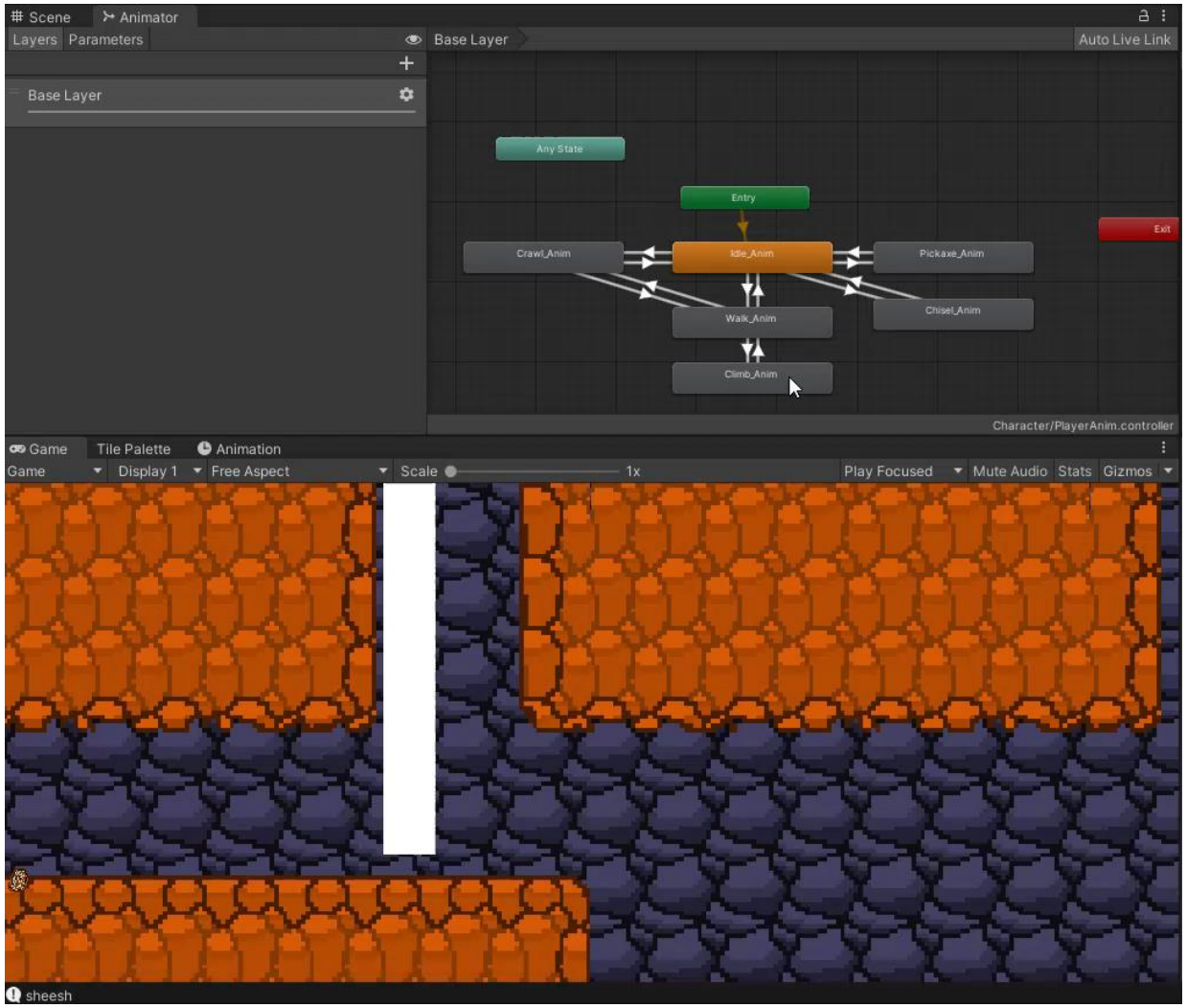
ASSET IMPLEMENTATION – DAY 56

- I exported the character as a sprite sheet from Aseprite and imported this into the unity project into its own character folder, I then removed the compression and sliced the sprite sheet by rows and columns – ready to be put into the scene and animated next time I work on the prototype. 6.54 -5.01



PROGRAMMING – DAY 57

- I then used the previously implemented sprite sheet with this animator to animate the character – switching between the states I created with keyframes based on different parameters that I setup.



```
ger.cs    PlayerController.cs
neous Files                                  PlayerCon
        }

        if(isInNarrowTunnelActivator == true && Input.GetKeyDown(KeyCode.Q))
        {
            transform.Rotate(0, 0, 90);
            transform.position = new Vector2(-2, -4.232444f);
            Debug.Log("Caving");
            enterNarrowTunnelText.gameObject.SetActive(false);
            isCrawling = true;
            animator.SetBool("IsCrawling", true);
        }

        if (Input.GetMouseButtonDown(0))
        {
            animator.SetBool("PickaxeSwung", true);
            StartCoroutine(ToolCoolDown());
        }

        if (Input.GetMouseButtonDown(1))
        {
            animator.SetBool("ChiselUsed", true);
            StartCoroutine(ToolCoolDown());
            fossilManagerScript.hasBeenExposed = true;
        }


    }

    void Flip()
    {
        Vector2 currentScale = gameObject.transform.localScale;
        currentScale.x *= -1;
        gameObject.transform.localScale = currentScale;

        facingRight = !facingRight;
    }

    IEnumerator ToolCoolDown()
    {
        yield return new WaitForSeconds(1);
        animator.SetBool("PickaxeSwung", false);
        animator.SetBool("ChiselUsed", false);
    }
```

```
    public Animator animator;
    private bool facingRight = true;
    private FossilManager fossilManagerScript;
    // Start is called before the first frame update
    void Start()
    {
        playerRb = GetComponent<Rigidbody2D>();
        fossilManagerScript = GameObject.Find("Ammonite Fossil").GetComponent<FossilManager>();
    }

    // Update is called once per frame
    void Update()
    {
        if(Input.GetKeyDown(KeyCode.Space))
        {
            isOnLadder = false;
            playerRb.gravityScale = 1;
            animator.SetBool("IsClimbing", false);
        }
        // assigns the horizontal axis input (A and D keys) to the horizontal axis variable
        horizontalInput = Input.GetAxis("Horizontal");
        // assigns the vertical axis input (W and S keys) to the vertical axis variable
        verticalInput = Input.GetAxis("Vertical");
        if(isOnLadder == false && isCrawling == false)
        {
            // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
            transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
            animator.SetFloat("Speed", Mathf.Abs(horizontalInput));
        }
        if (horizontalInput > 0 && !facingRight)
        {
            Flip();
        }
        if (horizontalInput < 0 && facingRight)
        {
            Flip();
```

PROGRAMMING – DAY 57

I UPDATED THE PARAMETERS IN THE SCRIPT WHEN A CHANGE OF ANIMATION STATE IS NEEDED, THEN CALLED A METHOD THAT FLIPS THE AXIS OF THE CHARACTER WHEN THE PLAYER MOVES IN A DIFFERENT DIRECTION – SO THEY FACE THE DIRECTION THEY ARE MOVING IN. THE TOOLS WORK WITH AN IE-NUMERATOR THAT ALLOWS YOU TO USE THE TOOLS AGAIN AFTER A SECOND HAS PASSED.

TUTORIAL – DAY 57

- I watched this tutorial by Dani Krossing to learn a method of flipping the character sprite in the direction that they move in.

```
void FixedUpdate()
{
    inputHorizontal = Input.GetAxisRaw("Horizontal");
    inputVertical = Input.GetAxisRaw("Vertical");

    if (inputHorizontal != 0)
    {
        rb.AddForce(new Vector2(inputHorizontal * speed, 0f));
    }

    if (inputHorizontal > 0 && !facingRight)
    {
        Flip();
    }
    if (inputHorizontal < 0 && facingRight)
    {
        Flip();
    }
}

void Flip()
{
    Vector3 currentScale = gameObject.transform.localScale;
    currentScale.x *= -1;
    gameObject.transform.localScale = currentScale;

    facingRight = !facingRight;
}
```

8:14 / 8:48 • Method 2: Flip player (the good way)

HOW TO FLIP YOUR SPRITE IN UNITY | Flip Your Player In Unity | Unity Tutorial

Dani Krossing
468K subscribers
Subscribe

932 Share Clip Save

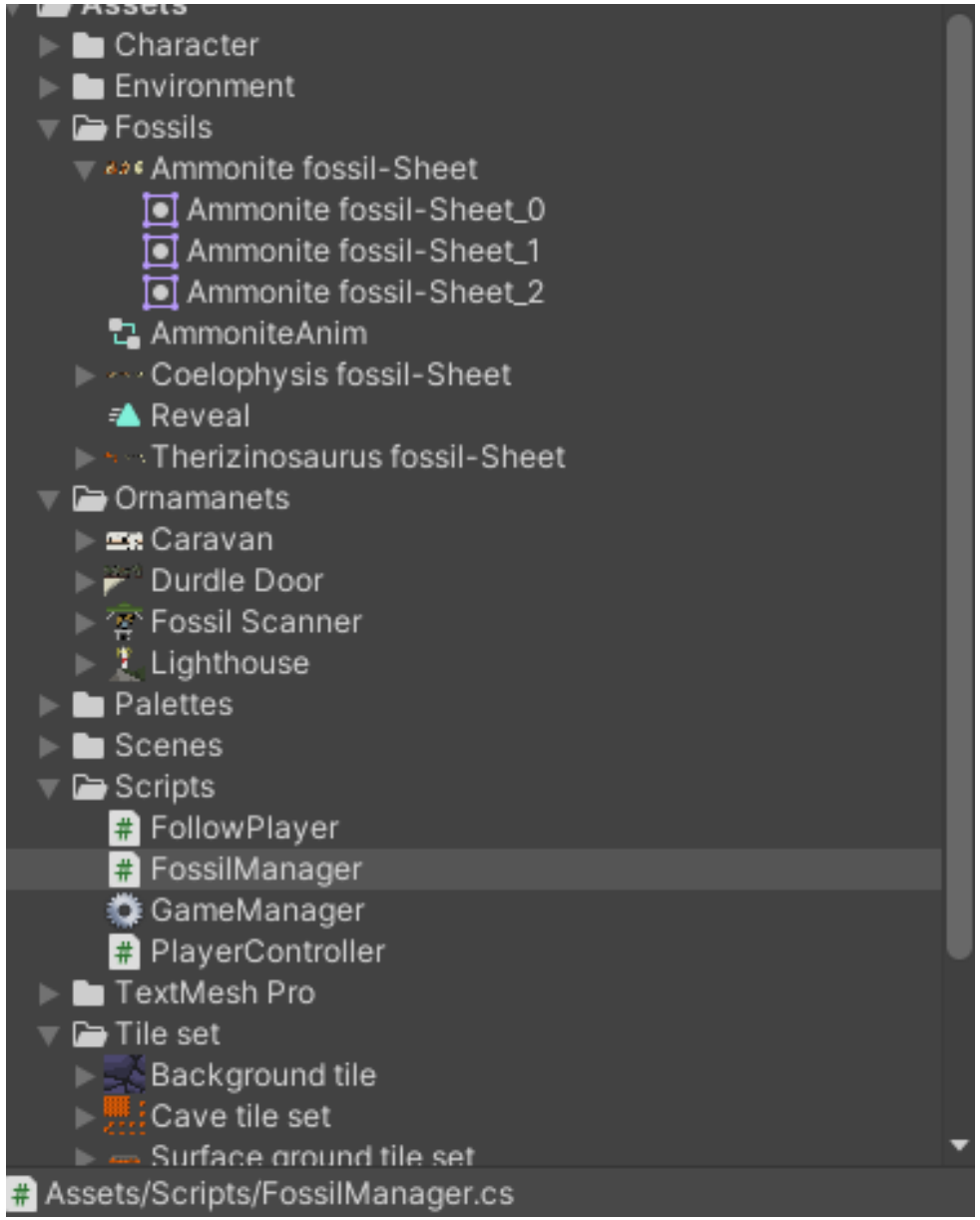
55K views 2 years ago

EQUIPMENT ART – DAY 58

- I created the final rendition of the fossil bag using the concept art I had made previously, making its normal and torn states.

ASSET IMPLEMENTATION – DAY 58

I IMPORTED MY FOSSILS INTO THE PROJECT AND CREATED A FOSSIL MANAGER SCRIPT – WHICH I KNEW I'D NEED WHEN PROGRAMMING THE FOSSILS INTO THE GAME

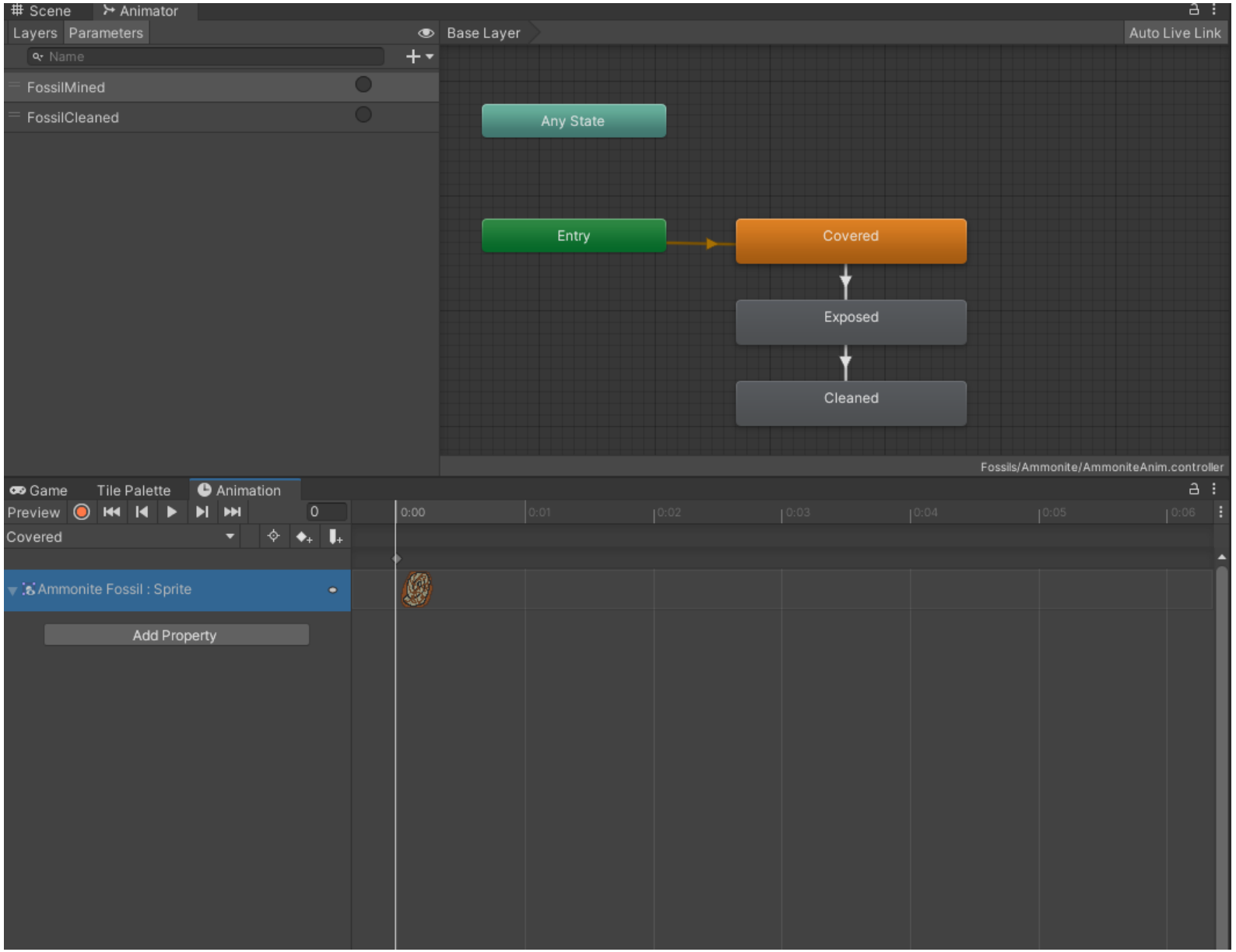


PROBLEM SOLVING AND PROGRAMMING – DAY 59

- The Ammonite fossil is in the scene and can be mined with the pickaxe and cleaned with the chisel, the Coelophysis fossil is in the scene without animation.

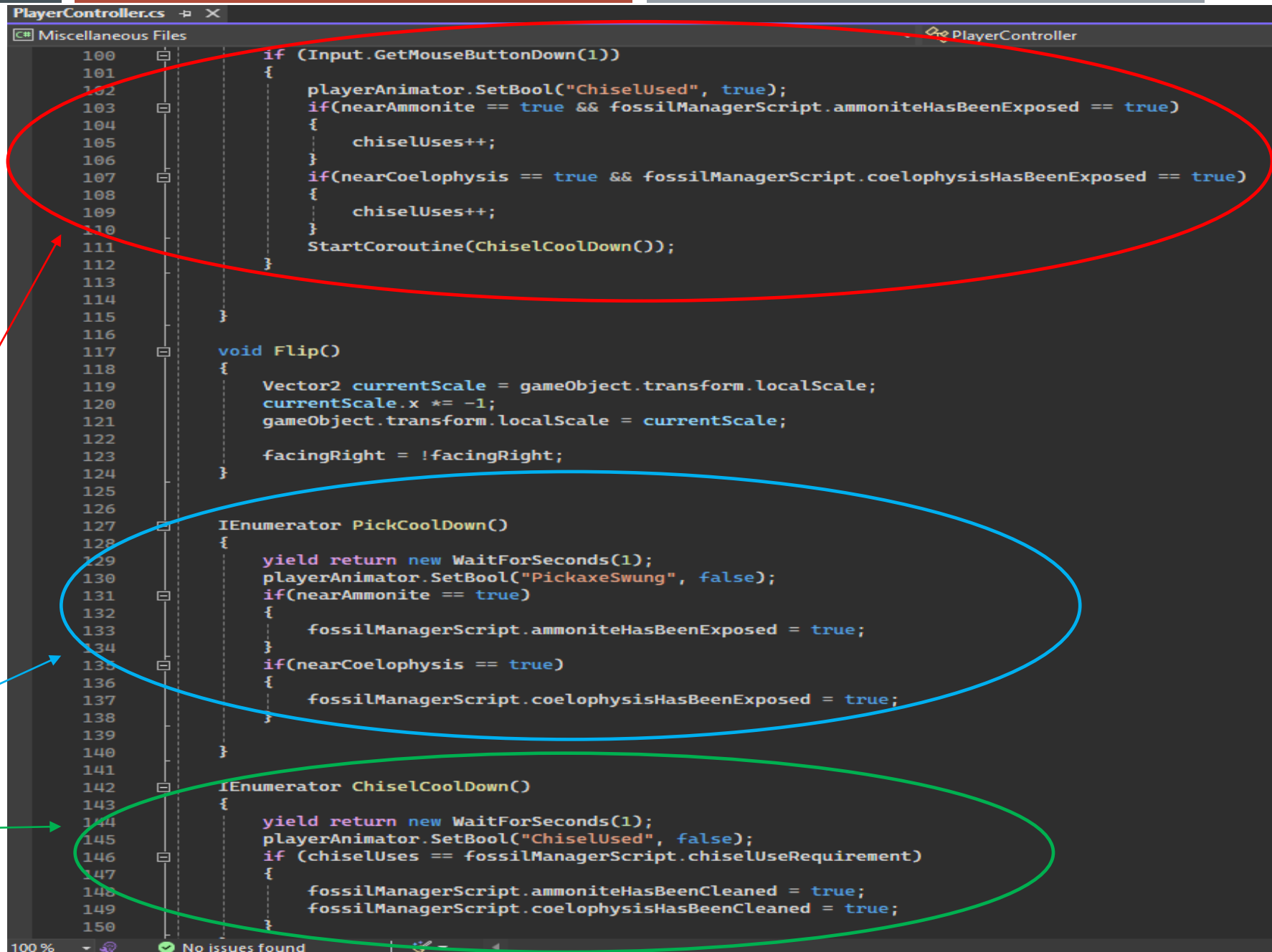
PROBLEM SOLVING AND PROGRAMMING – DAY 59

- I made each individual frame its own animation state (as the fossil only animates one frame at a time), which transition depending if the triggers I made are set.



PROGRAMMING AND PROBLEM SOLVING – DAY 59

- When the chisel is used, I added one to a chisel uses variable if this was done on a fossil that has been exposed.
- I separated the tool cool down IE numerator into two different methods for each type of tool – exposing the fossil if the pickaxe was used on it, and getting the fossil to its final stage if the chisel is used on it the right number of times.




```
100         if (Input.GetMouseButtonDown(1))
101         {
102             playerAnimator.SetBool("ChiselUsed", true);
103             if(nearAmmonite == true && fossilManagerScript.ammoniteHasBeenExposed == true)
104             {
105                 chiselUses++;
106             }
107             if(nearCoelophysis == true && fossilManagerScript.coelophysisHasBeenExposed == true)
108             {
109                 chiselUses++;
110             }
111             StartCoroutine(ChiselCoolDown());
112         }
113
114
115     }
116
117     void Flip()
118     {
119         Vector2 currentScale = gameObject.transform.localScale;
120         currentScale.x *= -1;
121         gameObject.transform.localScale = currentScale;
122
123         facingRight = !facingRight;
124     }
125
126
127     IEnumerator PickCoolDown()
128     {
129         yield return new WaitForSeconds(1);
130         playerAnimator.SetBool("PickaxeSwung", false);
131         if(nearAmmonite == true)
132         {
133             fossilManagerScript.ammoniteHasBeenExposed = true;
134         }
135         if(nearCoelophysis == true)
136         {
137             fossilManagerScript.coelophysisHasBeenExposed = true;
138         }
139
140     }
141
142     IEnumerator ChiselCoolDown()
143     {
144         yield return new WaitForSeconds(1);
145         playerAnimator.SetBool("ChiselUsed", false);
146         if (chiselUses == fossilManagerScript.chiselUseRequirement)
147         {
148             fossilManagerScript.ammoniteHasBeenCleaned = true;
149             fossilManagerScript.coelophysisHasBeenCleaned = true;
150         }
```

PROGRAMMING AND PROBLEM SOLVING – DAY 59

- I detect if the player is acting on the fossil with their tool via is-trigger box collider sensors that set the near fossil variables to true when you enter them and false when you exit them.

```
190         if(other.gameObject.CompareTag("Ammonite Sensor"))
191         {
192             nearAmmonite = true;
193         }
194         if(other.gameObject.CompareTag("Coelophysis Sesnor"))
195         {
196             nearCoelophysis = true;
197         }
198     }
199
200     private void OnTriggerExit2D(Collider2D other)
201     {
202         if (other.gameObject.CompareTag("Ladder Activator"))
203         {
204             isInLadderActivator = false;
205             placeLadderText.gameObject.SetActive(false);
206             Debug.Log("flip");
207         }
208         if(other.gameObject.CompareTag("Narrow Tunnel Activator"))
209         {
210             isInNarrowTunnelActivator = false;
211             enterNarrowTunnelText.gameObject.SetActive(false);
212         }
213         if (other.gameObject.CompareTag("Ornament Activator"))
214         {
215             ornaments.gameObject.SetActive(true);
216         }
217         if (other.gameObject.CompareTag("Ammonite Sensor"))
218         {
219             nearAmmonite = false;
220         }
221         if (other.gameObject.CompareTag("Coelophysis Sesnor"))
222         {
223             nearCoelophysis = false;
224         }
```

PROGRAMMING AND PROBLEM SOLVING – DAY 59

- In the fossil manager I created public Booleans that set the triggers in the fossil's animator, these bools are set to true in the player controller.
- I also made a public integer for the chisel use requirement, which will be changed to 3, 5, or 7 in the inspector – depending on the fossil the script is attached to.

AGE RATINGS AND CLASSIFICATION – DAY 60

- When considering the Pegi rating, I chose Pegi 3 as I believe my game is suitable for all age groups – as it doesn't contain any violence, bad language, or any sounds or pictures that are likely to frighten young children.
- I also chose this Pegi rating as the game is an educational experience that should be accessible to as wide of an audience as possible.

IP & COPYRIGHT LAW – DAY 60

- Copyright laws don’t need to be considered for this project as all the game assets are my own, and the game is my own IP as it is my own creation – this prevents my productions being stolen or copied.

CONSUMER PROTECTION AND ADVERTISING LAWS – DAY 60

- The game doesn’t contain anything involving transactions, privacy settings, currency, gambling, freemium models, or auto-renewal. The game could be played by children, but it is suitable for this as it doesn't contain any of the previously mentioned aspects or any content that may frighten children.
- There are no adverts or chance elements in this game, so the advertising laws don’t need to be considered.

DATA PROTECTION AND PRIVACY – DAY 60

- No personal information or data is stored from the player at any point, so this also doesn’t apply to this project.

	Negligible	Minor	Moderate	Significant	Severe
Very likely					
Likely					
Possible	Computer use posture		Illness	Loss of work/data	
Unlikely	Carpal tunnel	Back pain			
Very unlikely				Damaged hardware	

RISKS AND HAZARDS – DAY 60

I CREATED A RISK MATRIX FOR THE PROJECT, OUTLINING THE LIKELIHOOD AND IMPACT OF DIFFERENT RISKS THAT MAY OCCUR.

SAFE COMPUTER USE – DAY 60

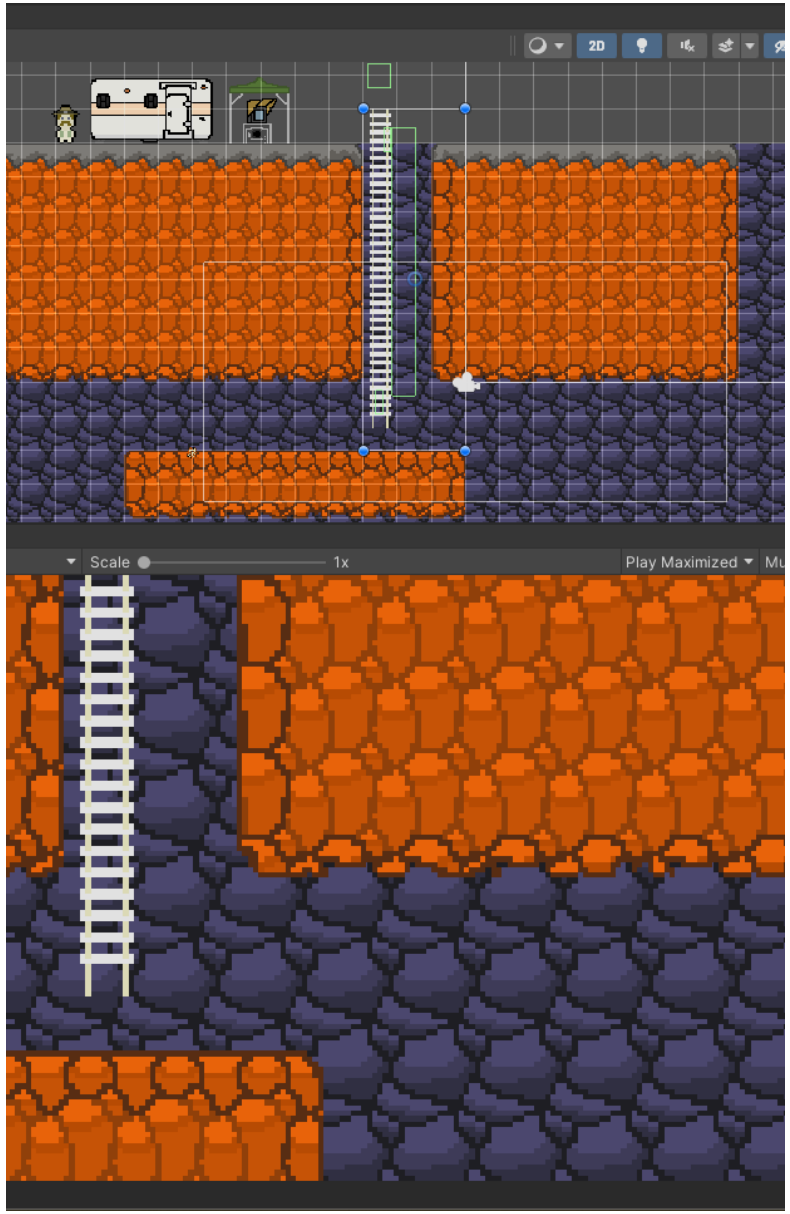
- To reduce the likelihood of risks such as eye strain, back pain, and carpal tunnel, I will take breaks from the computer.
- I also sit with a straight posture to reduce the chance of getting any back pain.

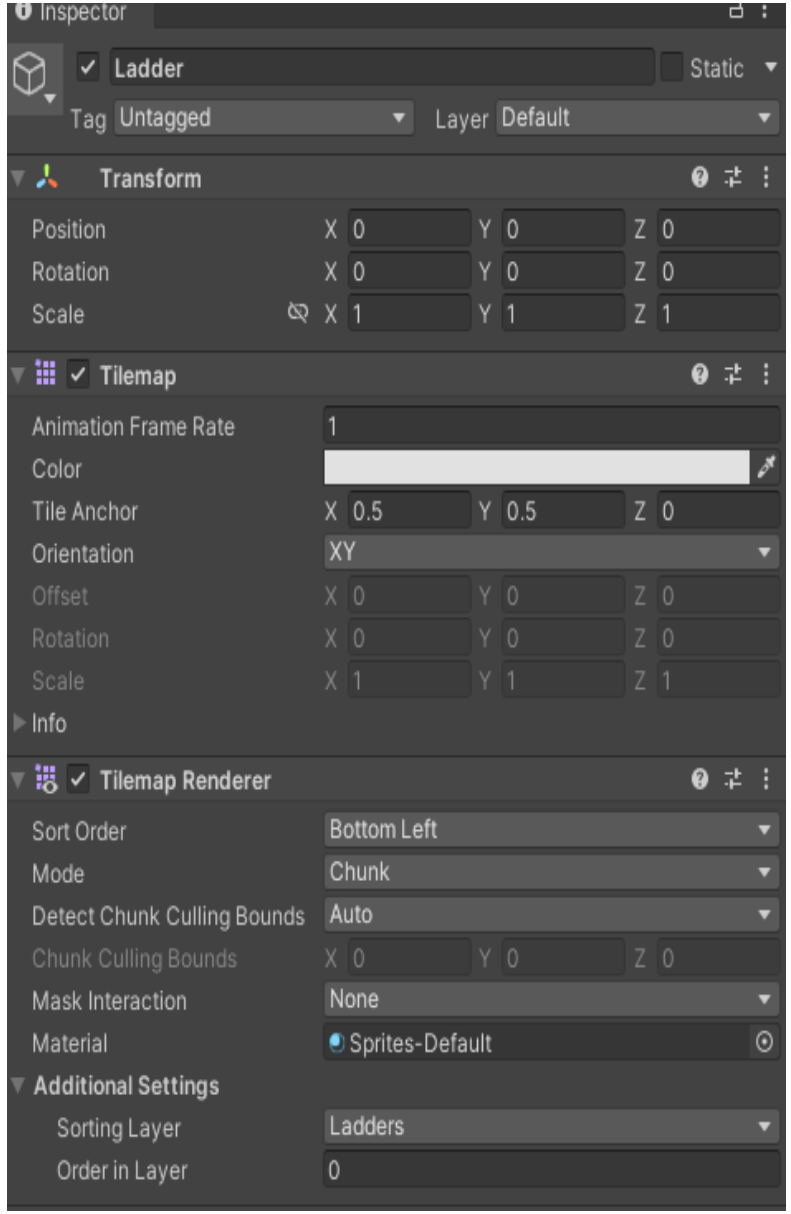
CONTENTS – WEEK 10

PROGRAMMING ART

PROBLEM SOLVING – DAY 63

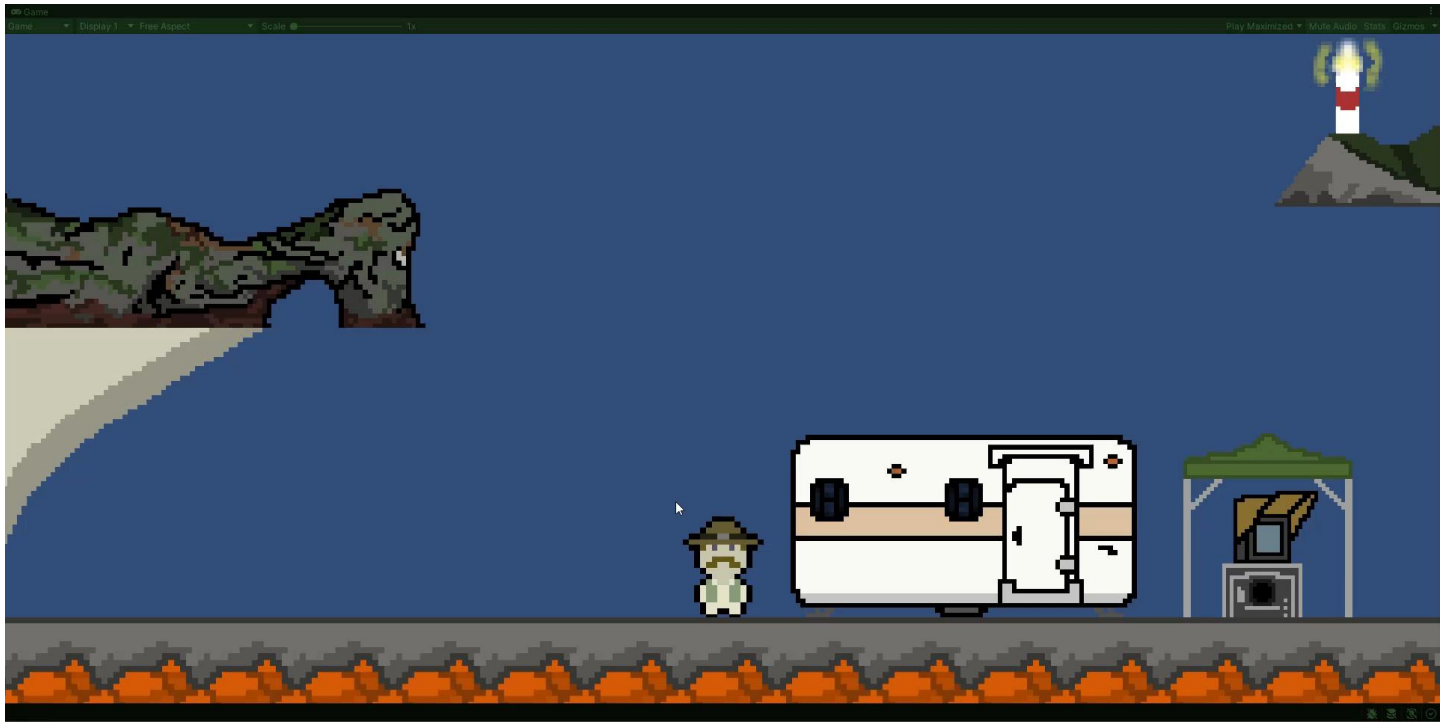
- I removed the sprite renderer that rendered the white rectangles from the ladder and replaced this component with a tile map renderer – then I made the ladder a child of the grid (like all the other tile maps) and painted the ladder tiles under this tile map renderer.





PROBLEM SOLVING – DAY 63

- Now both current fossils can be mined and then cleaned with the chisel, while the Ammonite sample (the entire thing, as its small) can also be extracted and scanned at the dig site – notice the debug log in the bottom left that appears once I return to the dig site after collecting the Ammonite.

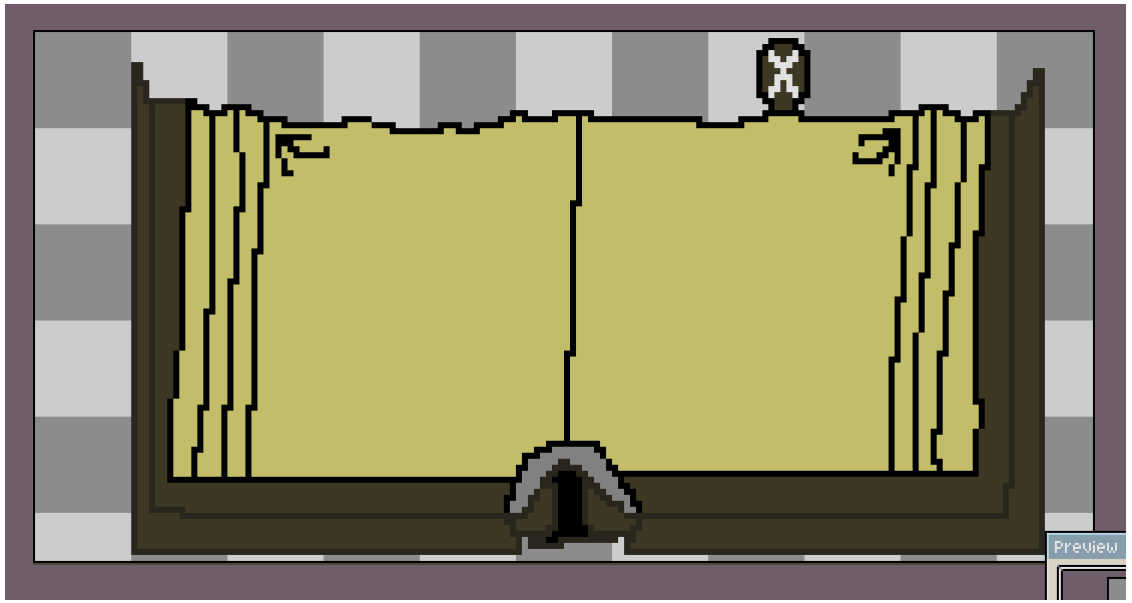


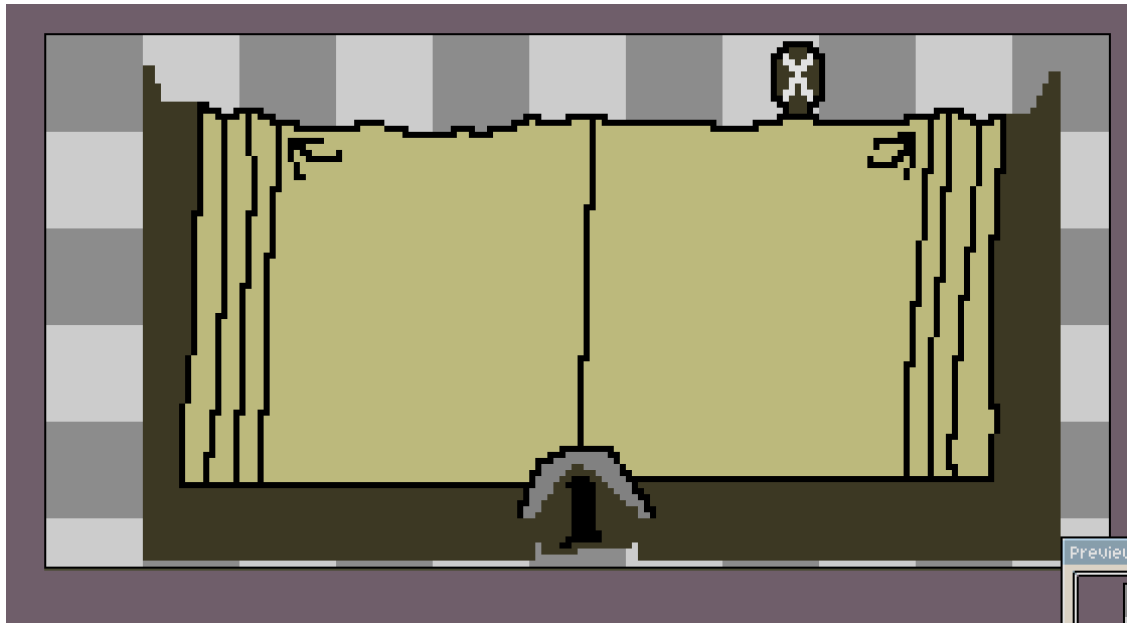
PROGRAMMING – DAY 63

- To make the Coelophysis fossil work with the tools and animate properly by itself, I made a separate variable that got its fossil manager script via the component of the Coelophysis game object and referenced the variables from this – meaning that still works when the ammonite game object is in-active, which the Coelophysis previously referenced.
- In update, I created an if statement for extracting the Ammonite when the player is standing by it, when the fossil is cleaned, and when the player presses the E key – the result sets the fossil to in-active and the has fossil sample variable to true, while resetting the chisel uses back to zero. I also made an if statement for scanning the fossil, which adds to the fossils obtained variable and sets the has fossil sample variable back to false, when the player with a fossil sample presses the S key at the scanner.

```
157
158     IEnumerator PickCoolDown()
159     {
160         yield return new WaitForSeconds(1);
161         playerAnimator.SetBool("PickaxeSwung", false);
162         pickaxeSwung = false;
163         if(nearAmmonite == true)
164         {
165             fossilManagerScriptA.ammoniteHasBeenExposed = true;
166         }
167         if(nearCoelophysis == true)
168         {
169             fossilManagerScriptC.coelophysisHasBeenExposed = true;
170         }
171
172     }
173
174     IEnumerator ChiselCoolDown()
175     {
176         yield return new WaitForSeconds(1);
177         playerAnimator.SetBool("ChiselUsed", false);
178         chiselUsed = false;
179         if (chiselUses == fossilManagerScriptA.chiselUseRequirement)
180         {
181             fossilManagerScriptA.ammoniteHasBeenCleaned = true;
182             ammoniteFossilSketch.gameObject.SetActive(true);
183         }
184         if (chiselUses == fossilManagerScriptC.chiselUseRequirement)
185         {
186             fossilManagerScriptC.coelophysisHasBeenCleaned = true;
187             coelophysisFossilSketch.gameObject.SetActive(true);
188         }
```

```
126         if(nearAmmonite == true && fossilManagerScriptA.ammoniteHasBeenCleaned == true && Input.GetKeyDown(KeyCode.E))
127         {
128             fossilManagerScriptA.gameObject.SetActive(false);
129             hasFossilSample = true;
130             chiselUses -= chiselUses;
131         }
132
133         if(atScanner == true && hasFossilSample == true && Input.GetKeyDown(KeyCode.S))
134         {
135             Debug.Log("journal info");
136             fossilsObtained++;
137             hasFossilSample = false;
138         }
```



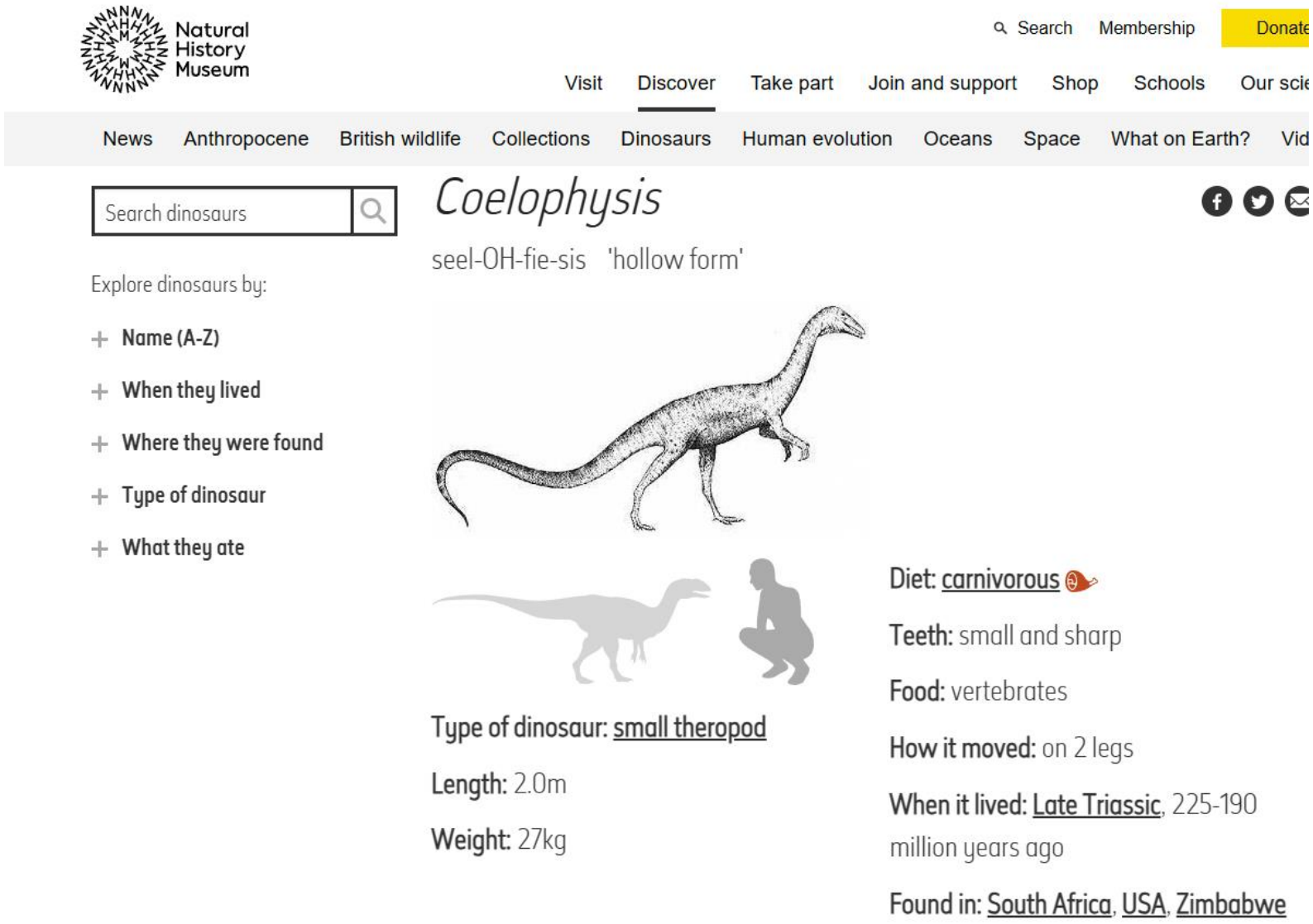


JOURNAL ART – DAY 63

- I transformed my concept art into the journal that will store the fossil sketches and info, iterating with the colors of the pages and the borders of the journal's hardback before reaching the final version on the left.

RESEARCH – DAY 63

- I Mainly used the natural history museum's website for my research on the prehistoric life in my game, to obtain the facts that I would add to the journal after the player has scanned the fossil – this slide shows some of my research on the Coelophysis.
- I included some other facts that I already knew, from series like walking with dinosaurs that I've watched previously, which I did fact-check on the internet before adding them into the game.



ASSET IMPLEMENTATION AND PROBLEM SOLVING – DAY 63

- Added journal and journal sketches
- Press j to open it and close it with a button
- First 2 fossil sketches appear under page 1-2 when fossils are cleaned

ASSET IMPLEMENTATION AND PROBLEM SOLVING – DAY 63

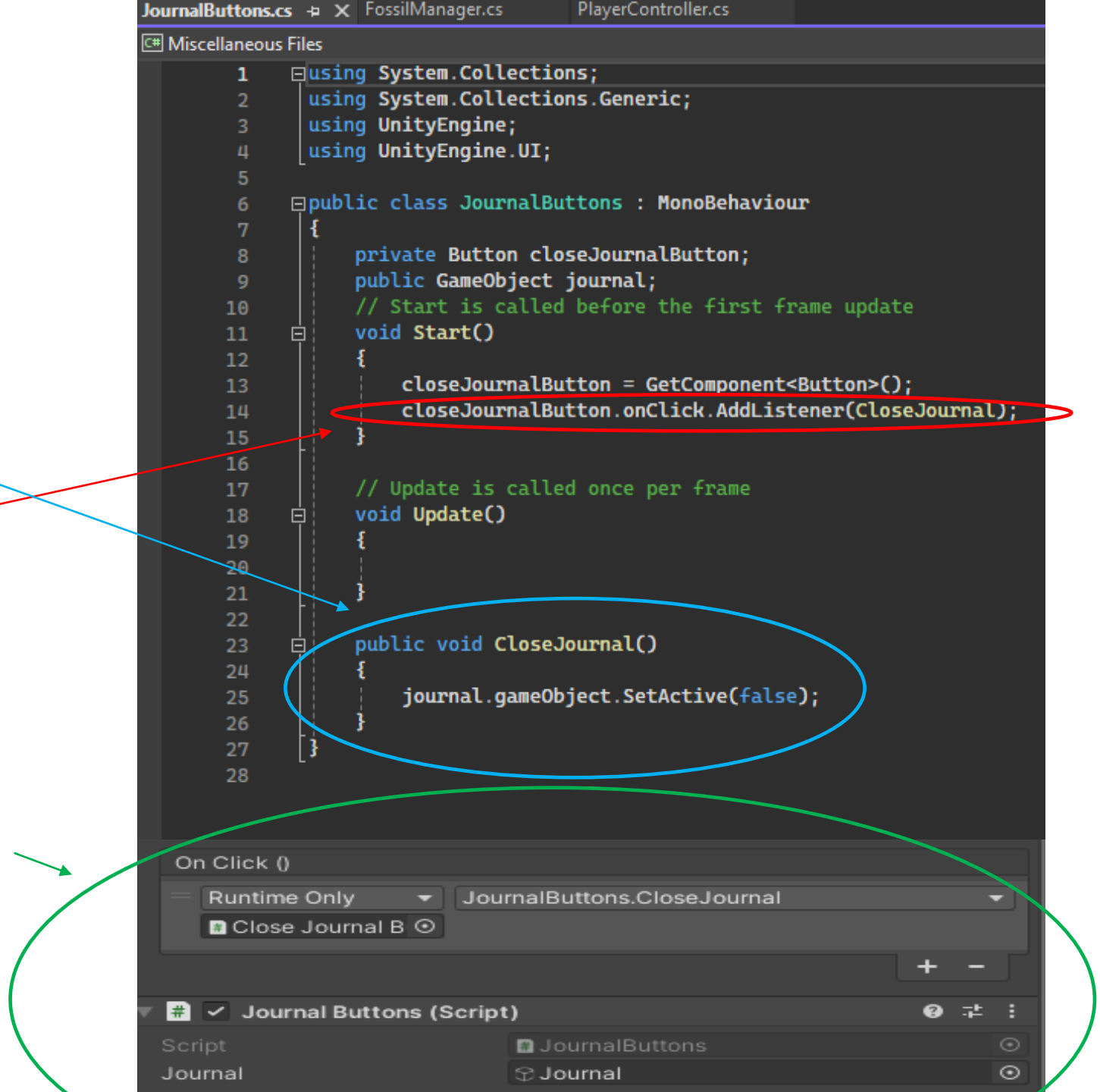
- I set the Journal active when J is pressed, and set the fossil sketches (which are children of the journal game object, and they're all part of the UI canvas) active at the same time as their respective fossil has been cleaned variable is set to true (when the fossil is bare)

```
IEnumerator ChiselCoolDown()
{
    yield return new WaitForSeconds(1);
    playerAnimator.SetBool("ChiselUsed", false);
    chiselUsed = false;
    if (chiselUses == fossilManagerScriptA.chiselUseRequirement)
    {
        fossilManagerScriptA.ammoniteHasBeenCleaned = true;
        ammoniteFossilSketch.gameObject.SetActive(true);
    }
    if (chiselUses == fossilManagerScriptC.chiselUseRequirement)
    {
        fossilManagerScriptC.coelophysisHasBeenCleaned = true;
        coelophysisFossilSketch.gameObject.SetActive(true);
    }
}
```

```
if(Input.GetKeyDown(KeyCode.J))
{
    journal.gameObject.SetActive(true);
}
```

PROGRAMMING – DAY 63

- In this new journal buttons script, I called a method that sets the journal back to being in-active when the close journal button is clicked – I set this button up in the scene view when I adjusted the scale of the button's borders to fit around the white cross of the journal, then I removed the text and the image along with its colors to make it transparent (the button art is already part of the journal's art), and finally I called the close journal method in the inspector with the on click function.

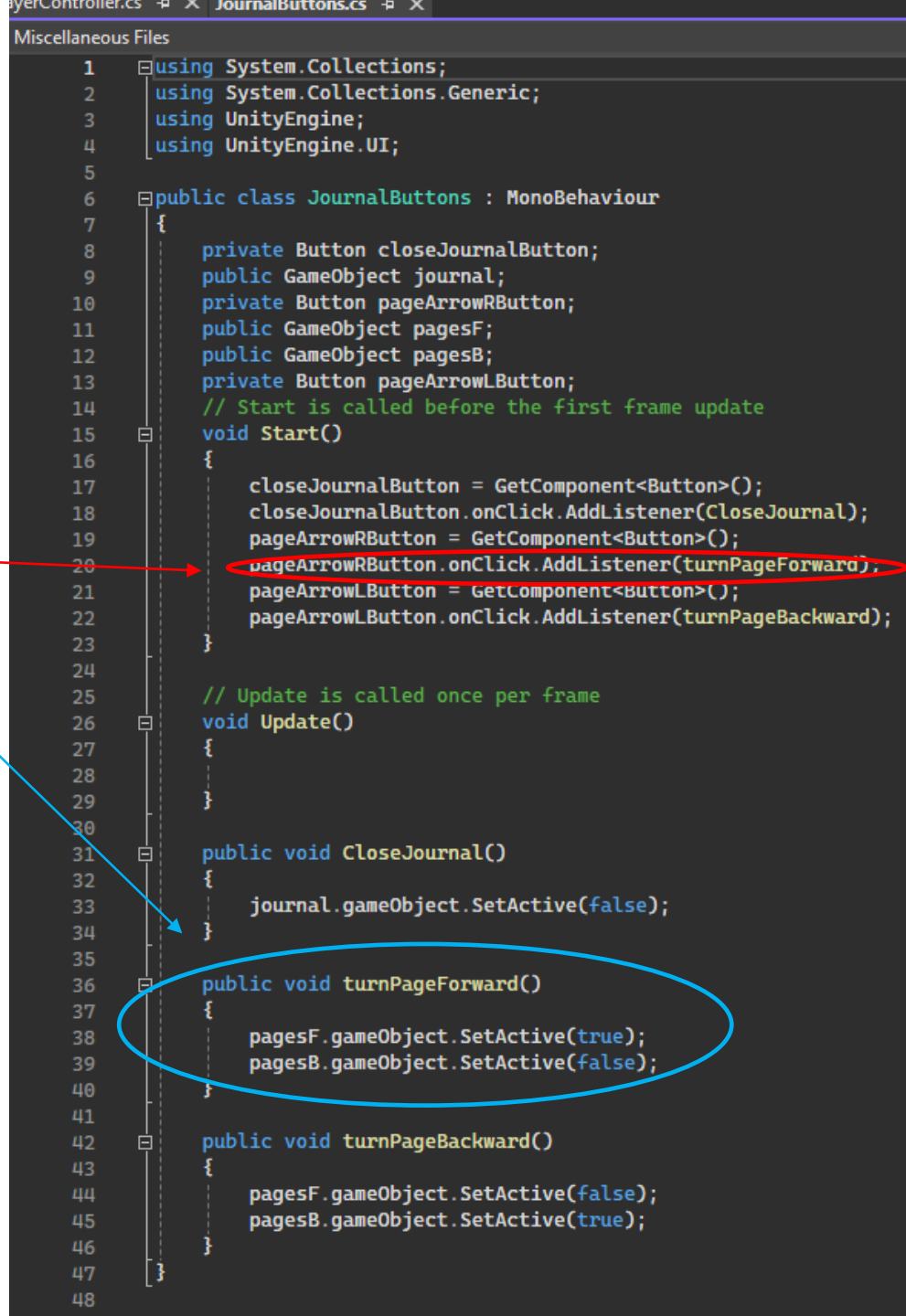


PROBLEM SOLVING – DAY 64

- The first two fossils now function fully in terms of retrieving them and viewing your findings in the journal, I just need to replace the extracted frame of the Coelophysis animation with a version of the fossil without a tooth.

PROBLEM SOLVING – DAY 64

- I repeated the same process as the close journal button for the buttons that turn the pages of the journal, except when the arrow on the right of the journal is clicked the "turn page forward" method is called – the last two pages are set to active while the first two pages are set in-active – and the opposite happens when the arrow on the left is clicked.
- The Ammonite and Coelophysis sketches are children of an empty game object called "pages1-2" and the other two fossils are children of the "pages3-4" empty game object – these empty game objects are set to active and in-active on each click of the different arrow buttons, showing pages3-4 and hiding pages 1-2 when the right arrow is clicked (and vice versa)



PROBLEM SOLVING – DAY 64

- I moved the “E” input part of the extracting fossils if statement into its own if statement within it, so text prompts can appear when you are only near the fossil that is ready to be extracted – and then the fossil gets extracted when you press E. I also added a variable to check if the fossil has been extracted, so this can be checked when scanning the fossil.
- I did the same for the input part of the scanning fossils if statement, while setting the information text true when its related fossil has been extracted.

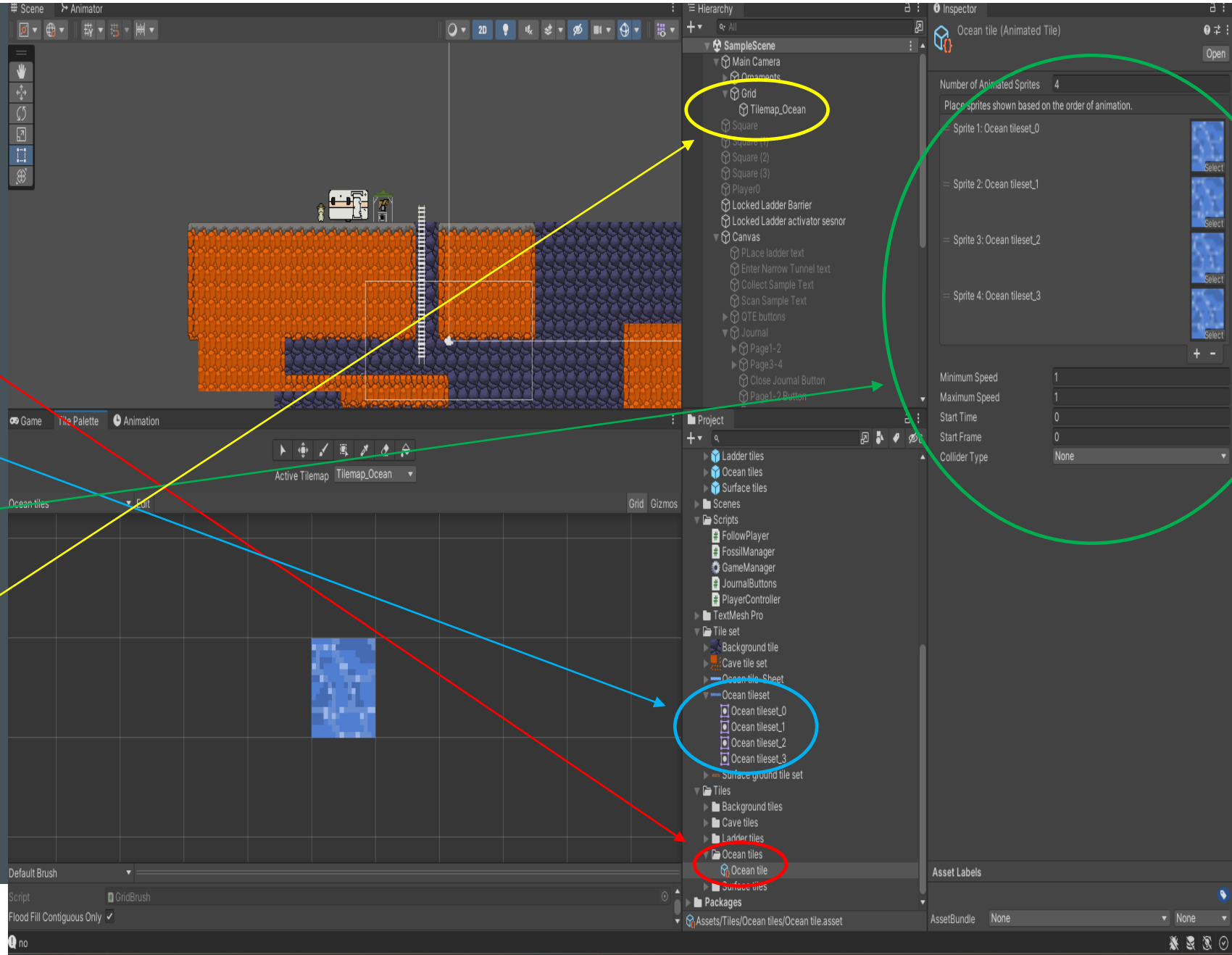
```
if(nearAmmonite == true && fossilManagerScriptA.ammoniteHasBeenCleaned == true)
{
    collectSampleText.gameObject.SetActive(true);
    if(Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptA.gameObject.SetActive(false);
        fossilManagerScriptA.ammoniteHasBeenExtracted = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        collectSampleText.gameObject.SetActive(false);
    }
}

if (nearCoelophysis == true && fossilManagerScriptC.coelophysisHasBeenCleaned == true && fossilManagerScriptC.coelophysisHasBeenExtracted == false)
{
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptC.coelophysisHasBeenExtracted = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        collectSampleText.gameObject.SetActive(false);
    }
}

if (atScanner == true && hasFossilSample == true)
{
    scanSampleText.gameObject.SetActive(true);
    if(Input.GetKeyDown(KeyCode.S))
    {
        Debug.Log("journal info");
        fossilsObtained++;
        hasFossilSample = false;
        scanSampleText.gameObject.SetActive(false);
        if(fossilManagerScriptA.ammoniteHasBeenExtracted == true)
        {
            ammoniteInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptC.coelophysisHasBeenExtracted == true)
        {
            coelophysisInfoText.gameObject.SetActive(true);
        }
    }
}
```

ASSET IMPLEMENTATION – DAY 65

- I imported my ocean tile that I had made (video on the left) but thought the water was too bright and harshly contrasted with the rest of the environment, so I went back into Aseprite briefly to darken all the elements of the ocean tile – then imported this updated version of the tile into the project (shown in the video on the right)

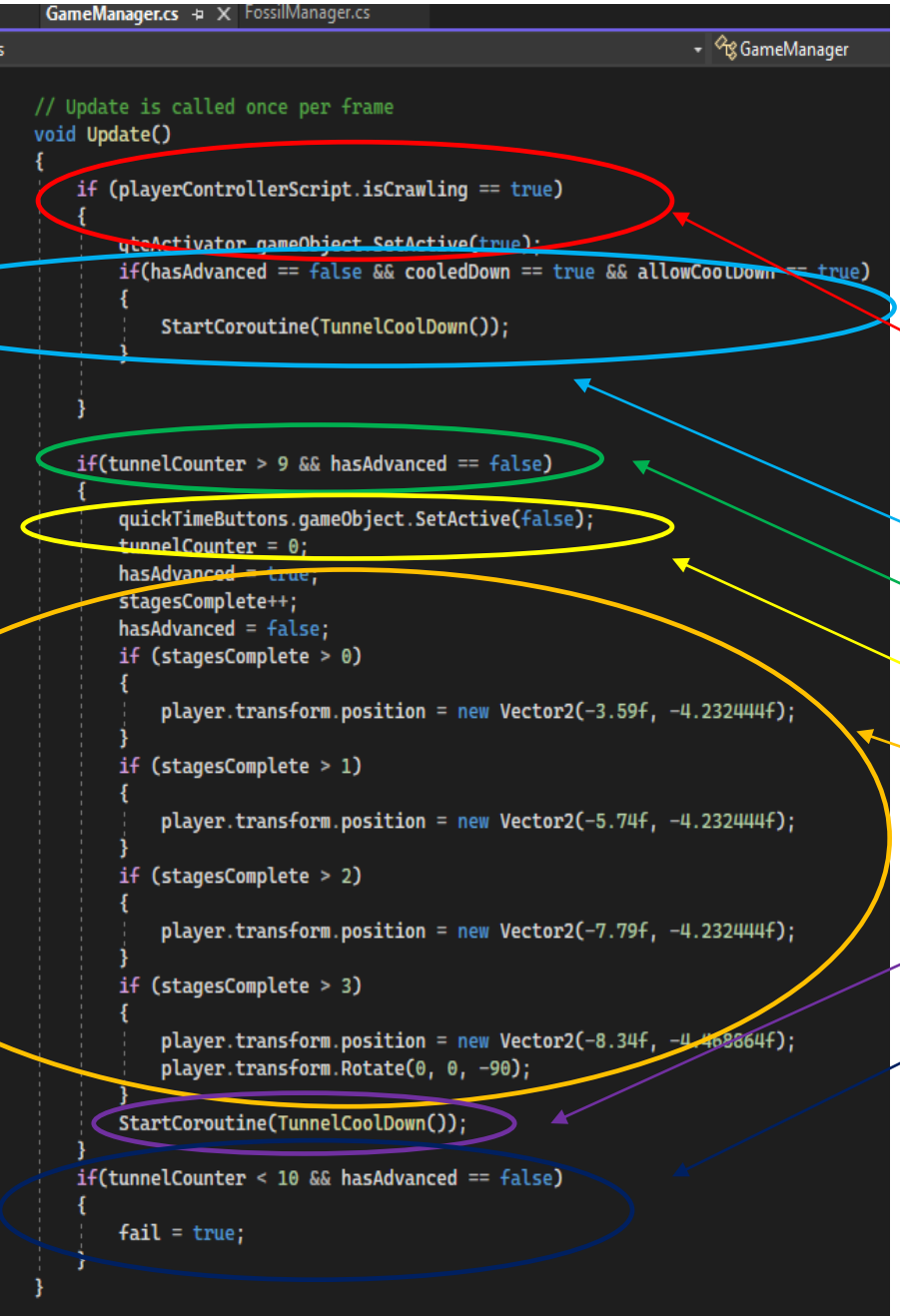
ASSET IMPLEMENTATION – DAY 65

- I created an animated tile in my new ocean tile folder, then added each frame as a sprite (which I sliced with the sprite editor of the ocean tile sprite sheet) into the four slots of the animation for the tile.
- I painted these tiles onto a new ocean tile map that is part of a grid which is a child to the main camera – making the ocean a stationary background piece, like the other coastal landforms – and ordered it into the ocean layer so its behind everything else.



PROBLEM SOLVING AND PROGRAMMING – DAY 66

- I began implementing the quick time event for crawling through the narrow tunnel when I added the button that the player will have to mash to progress, which does not spawn at a random position or animate the player yet.



PROBLEM SOLVING AND PROGRAMMING – DAY 66

- The game manager script contains the code for traversing the narrow tunnel – as it manages the player and UI elements.
- The QTE button is set active when the is crawling variable from the player controller is true, then the tunnel cool down is called every time the player hasn't moved and the cool down has finished.
- If the player clicks the button ten times before it disappears, the button is set in-active again and the player's clicks are reset to zero – also adding to how many stages the player has completed and moving the player to a new position, depending on how many stages have been completed – the cool down is called once again at the end of the if statement.
- Another if statement sets a failure variable to true when the player has clicked less than ten times and hasn't moved – although currently this doesn't give the effect I want.

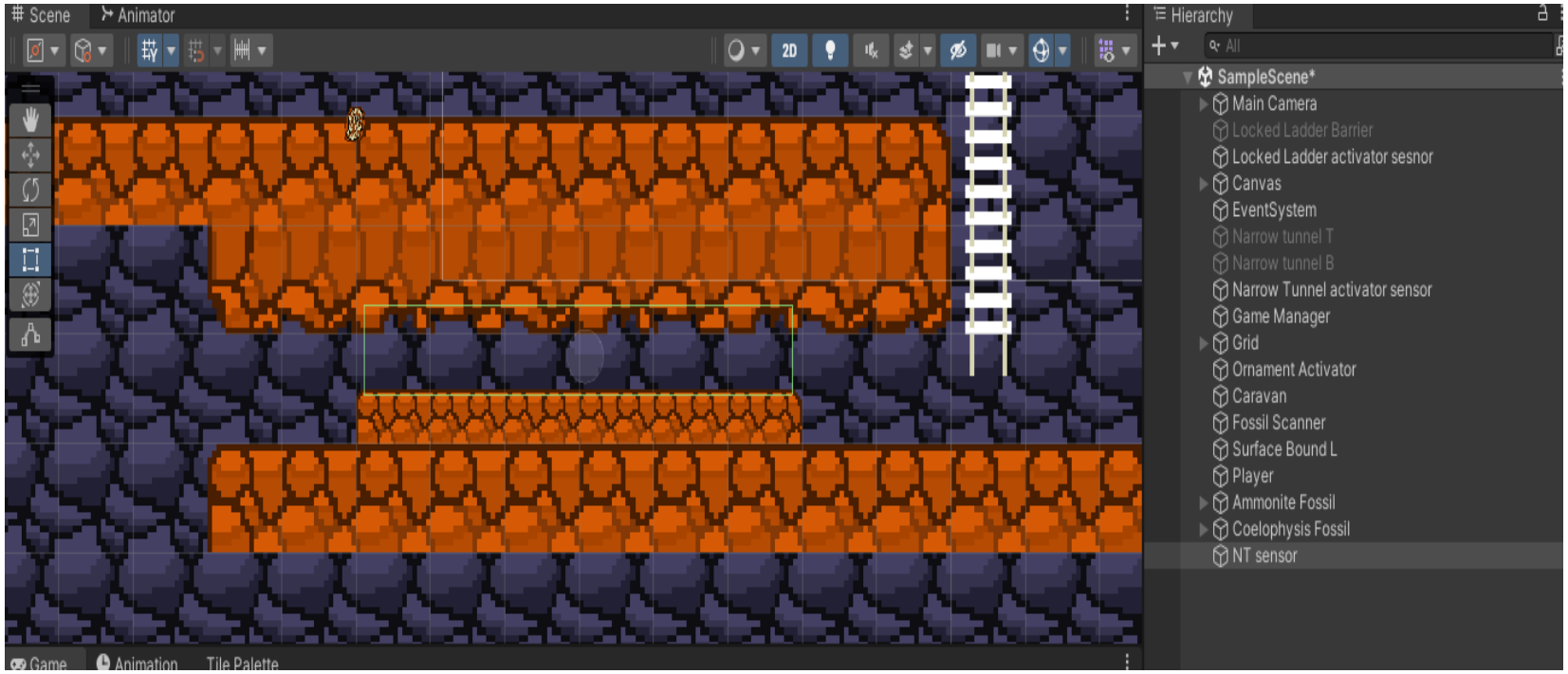
PROBLEM SOLVING AND PROGRAMMING – DAY 66

- The counter integer that stores the number of times the player has clicked the button is increased by one when the button's listener detects a click and calls this method via the on click feature in unity.
- The IE numerator for activating the quick time event waits two seconds before activating the button upon the first entry into the tunnel, then continues to deactivate and activate the button in two second intervals.

```
public void AdvancePlayer()
{
    tunnelCounter++;
    Debug.Log(tunnelCounter);
}

IEnumerator TunnelCoolDown()
{
    Debug.Log("Yo");
    cooledDown = false;
    if (stagesComplete == 0)
    {
        yield return new WaitForSeconds(2);
    }

    quickTimeButtons.gameObject.SetActive(true);
    yield return new WaitForSeconds(2);
    quickTimeButtons.gameObject.SetActive(false);
    yield return new WaitForSeconds(2);
    cooledDown = true;
    if(fail == true)
    {
        playerControllerScript.bagTorn = true;
        Debug.Log("Torn");
    }
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 66

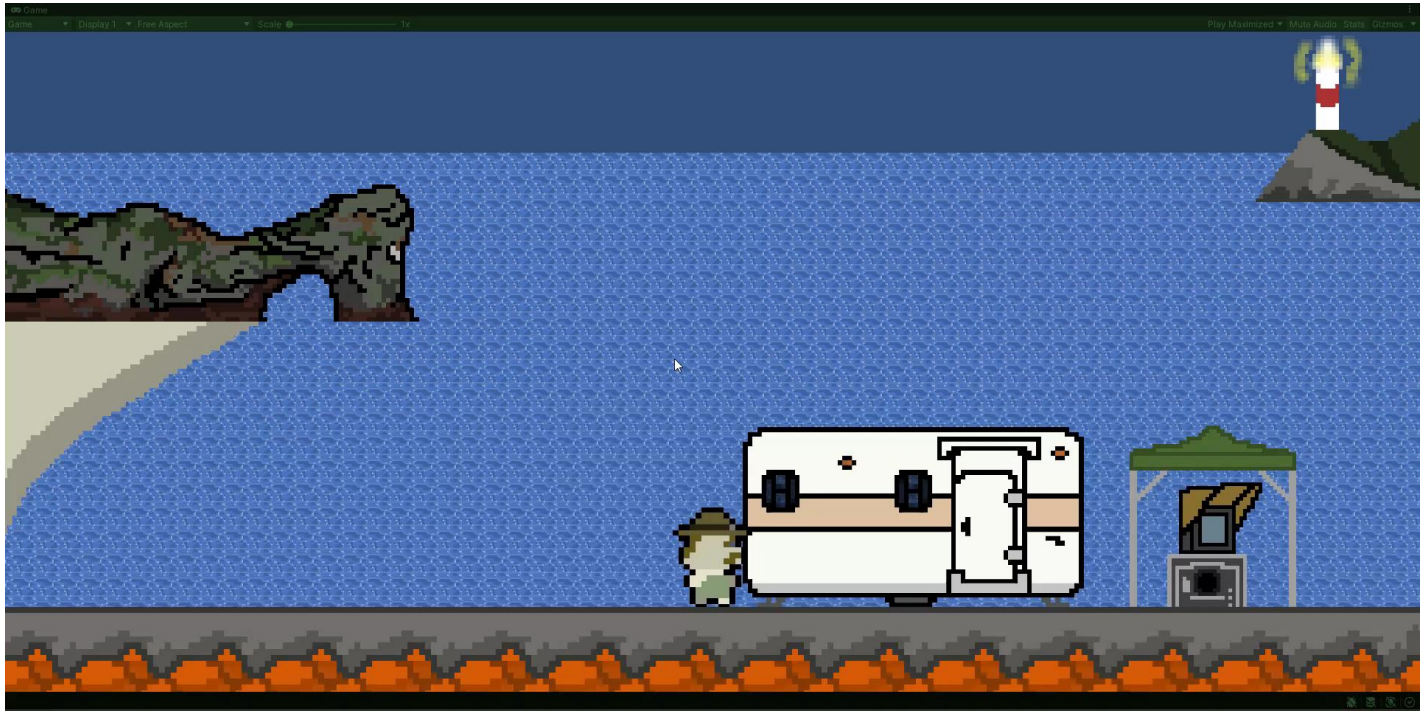
I SET THE IS CRAWLING VARIABLE TRUE WHEN THE PLAYER ENTERS THE SENSOR FOR THE NARROW TUNNEL, SO I CAN CHANGE THE VARIABLE BACK TO FALSE WHEN THE PLAYER LEAVES THIS IS-TRIGGER BOX COLLIDER.




```
if (other.gameObject.CompareTag("NT sensor"))
{
    isCrawling = true;
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 67

- The QTE button now spawns at a random position on the screen, and the player can enter the narrow tunnel from both sides.
- The Therizinosaurus claw is also now present in the scene and can be obtained.



PROBLEM SOLVING AND PROGRAMMING – DAY 67

- I created bools that state when the player has entered from the right or left side of the tunnel (crawling forwards or backwards), these are both set back to false when the player leaves the narrow tunnel sensor.

```
if(isInNarrowTunnelActivator == true && Input.GetKeyDown(KeyCode.Q))
{
    crawlingF = true;
    transform.Rotate(0, 0, 90);
    transform.position = new Vector2(-2, -4.232444f);
    Debug.Log("Caving");
    enterNarrowTunnelText.gameObject.SetActive(false);
    //playerAnimator.SetBool("IsCrawling", true);
}

if (isInNarrowTunnelActivatorL == true && Input.GetKeyDown(KeyCode.Q))
{
    crawlingB = true;
    transform.Rotate(0, 0, -90);
    transform.position = new Vector2(-7.79f, -4.232444f);
    playerAnimator.SetBool("CrawlMove", true);
    Debug.Log("Caving");
    enterNarrowTunnelText.gameObject.SetActive(false);
    //playerAnimator.SetBool("IsCrawling", true);
}
```

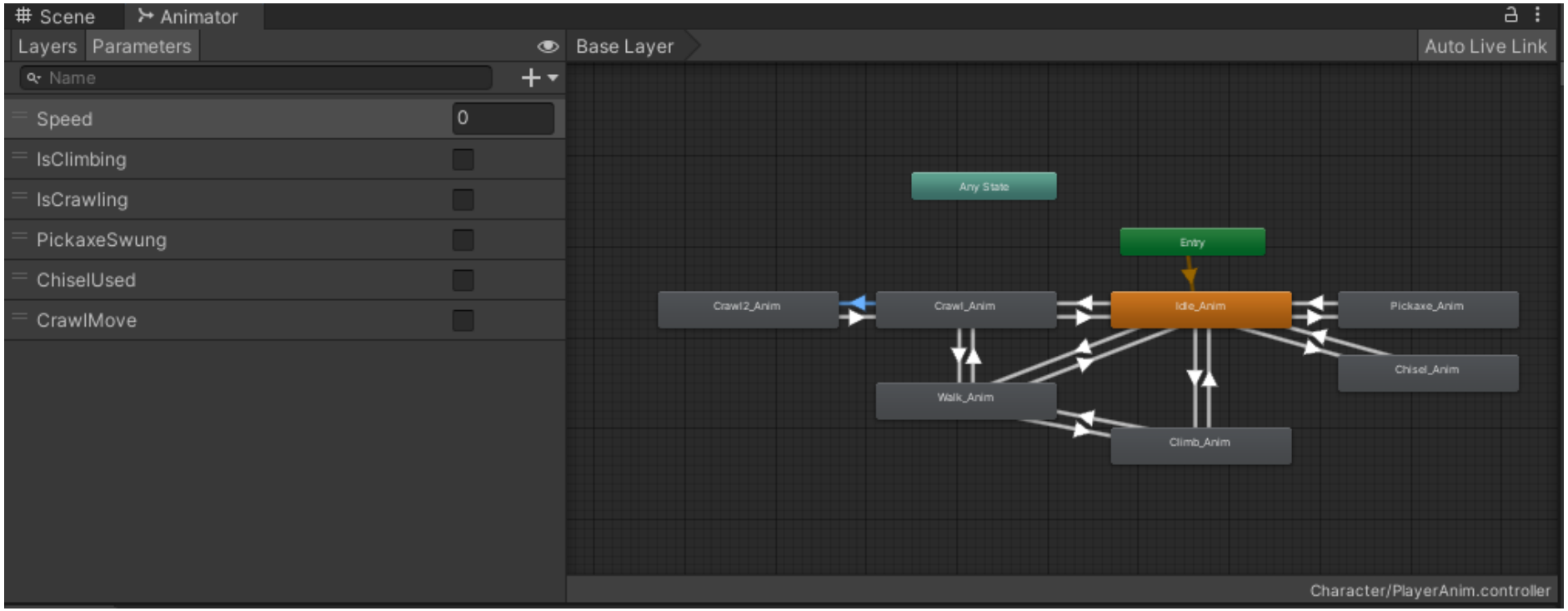
PROBLEM SOLVING AND PROGRAMMING – DAY 67

- I either added or subtracted from the stage complete variable, depending on what side of the tunnel the player entered from, so the player advances through the different positions from the right to left or they advance through the positions in the opposite order from the left to right.
- I then alternated the player between the two crawling animation frames each time they advance.

GameManager.cs | FossilManager.cs

GameManager

```
if(tunnelCounter > 9 && hasAdvanced == false)
{
    quickTimeButtons.gameObject.SetActive(false);
    tunnelCounter = 0;
    hasAdvanced = true;
    if(playerControllerScript.crawlingF == true)
    {
        stagesComplete++;
    }
    if(playerControllerScript.crawlingB == true)
    {
        stagesComplete--;
    }
    hasAdvanced = false;
    if (stagesComplete == -1)
    {
        player.transform.position = new Vector2(-1.558282f, -4.468854f);
        player.transform.Rotate(0, 0, 90);
        playerAnim.SetBool("CrawlMove", false);
        stagesComplete++;
    }
    if (stagesComplete == 0)
    {
        player.transform.position = new Vector2(-2, -4.232444f);
        playerAnim.SetBool("CrawlMove", false);
    }
    if (stagesComplete == 1 )
    {
        player.transform.position = new Vector2(-3.59f, -4.232444f);
        playerAnim.SetBool("CrawlMove", true);
    }
    if (stagesComplete == 2 )
    {
        player.transform.position = new Vector2(-5.74f, -4.232444f);
        playerAnim.SetBool("CrawlMove", false);
    }
    if (stagesComplete == 3)
    {
        player.transform.position = new Vector2(-7.79f, -4.232444f);
        playerAnim.SetBool("CrawlMove", true);
    }
    if (stagesComplete == 4)
    {
        player.transform.position = new Vector2(-8.34f, -4.468864f);
        player.transform.Rotate(0, 0, -90);
        playerAnim.SetBool("CrawlMove", false);
        stagesComplete--;
    }
    StartCoroutine(TunnelCoolDown());
```



PROBLEM SOLVING AND ANIMATION – DAY 67

- I added the two frames of the crawling animation as separate animations (the same way the fossils animate) into the player's animator and set the condition for switching between them as the crawl move Boolean being true one way and false the other – then the player returns to idling or walking if the is crawling bool returns to being false.

PROBLEM SOLVING AND PROGRAMMING – DAY 67

- An issue that arose previously was that the cool down would be called by two if statements at the same time – I resolved this by only allowing the contents of the coroutine to happen if the cooled down variable is true – which only is true once the coroutine finishes.
- I added the initial two second delay to when you enter the tunnel on the left side by checking if the stages complete is three and the player has entered on the left side (via the crawling backwards variable)
- I transformed the position of the button to a new random position each time it is set active again – giving the element of challenge to the quick time event. The random range value is the canvas size (of the game in focused view) multiplied by a value that matches the full screen canvas size.

```
IEnumerator TunnelCoolDown()
{
    if (cooledDown == true)
    {
        cooledDown = false;
        if (stagesComplete == 0 && playerControllerScript.crawlingF == true)
        {
            yield return new WaitForSeconds(2);
        }
        if(stagesComplete == 3 && playerControllerScript.crawlingB == true)
        {
            yield return new WaitForSeconds(2);
        }

        quickTimeButtons.gameObject.SetActive(true);
        float randomX = Random.Range(0, 1057*1.67f);
        float randomY = Random.Range(0, 483*1.55f);
        quickTimeButtons.gameObject.transform.position = new Vector2(randomX, randomY);
        yield return new WaitForSeconds(2.5f);
        quickTimeButtons.gameObject.SetActive(false);
        tunnelCounter = 0;
        yield return new WaitForSeconds(2);
        cooledDown = true;
    }

    if(fail == true)
    {
        playerControllerScript.bagTorn = true;
        Debug.Log("Torn");
    }
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 67

```
if (nearTheri == true && fossilManagerScriptT.tHBC == true)
{
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptT.gameObject.SetActive(false);
        fossilManagerScriptT.tHBExt = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        collectSampleText.gameObject.SetActive(false);
    }
}

if (atScanner == true && hasFossilSample == true)
{
    scanSampleText.gameObject.SetActive(true);
    if(Input.GetKeyDown(KeyCode.S))
    {
        Debug.Log("journal info");
        fossilsObtained++;
        hasFossilSample = false;
        scanSampleText.gameObject.SetActive(false);
        if(fossilManagerScriptA.ammoniteHasBeenExtracted == true)
        {
            ammoniteInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptC.coelophysisHasBeenExtracted == true)
        {
            coelophysisInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptT.tHBExt == true)
        {
            theriInfoText.gameObject.SetActive(true);
        }
    }
}
```

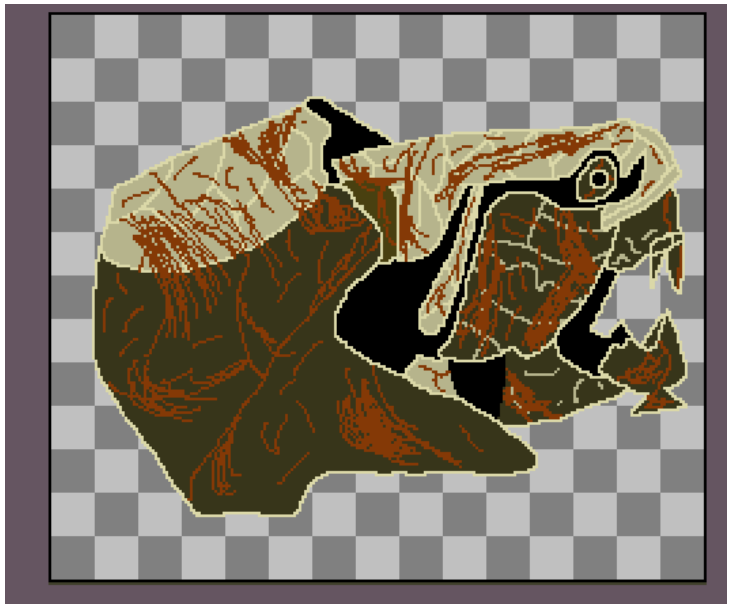
```
IEnumerator PickCoolDown()
{
    yield return new WaitForSeconds(1);
    playerAnimator.SetBool("PickaxeSwung", false);
    pickaxeSwung = false;
    if(nearAmmonite == true)
    {
        fossilManagerScriptA.ammoniteHasBeenExposed = true;
    }
    if(nearCoelophysis == true)
    {
        fossilManagerScriptC.coelophysisHasBeenExposed = true;
    }
    if(nearTheri == true)
    {
        fossilManagerScriptT.tHBExp = true;
    }
}

IEnumerator ChiselCoolDown()
{
    yield return new WaitForSeconds(1);
    playerAnimator.SetBool("ChiselUsed", false);
    chiselUsed = false;
    if (chiselUses == fossilManagerScriptA.chiselUseRequirement && nearAmmonite == true)
    {
        fossilManagerScriptA.ammoniteHasBeenCleaned = true;
        ammoniteFossilSketch.gameObject.SetActive(true);
    }
    if (chiselUses == fossilManagerScriptC.chiselUseRequirement && nearCoelophysis == true)
    {
        fossilManagerScriptC.coelophysisHasBeenCleaned = true;
        coelophysisFossilSketch.gameObject.SetActive(true);
    }
    if(chiselUses == fossilManagerScriptT.chiselUseRequirement && nearTheri == true)
    {
        fossilManagerScriptT.tHBC = true;
        theriFossilSketch.gameObject.SetActive(true);
    }
}
```

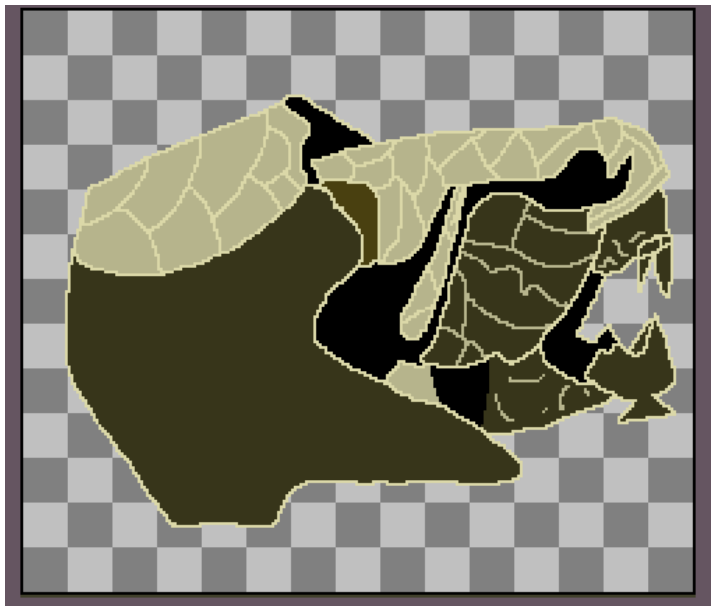
- I added the therizinosaurus with the exact same code as the other fossils, just replacing the variable names to the ones relevant to this fossil.

PROBLEM SOLVING AND PROGRAMMING – DAY 67

- I did the same when animating the claw (in the script and the Animator), except I shortened and simplified the variable names.

```
public bool coelophysisHasBeenExposed = false;
public bool coelophysisHasBeenCleaned = false;
public bool coelophysisHasBeenExtracted = false;
public bool tHBExp = false;
public bool tHBC = false;
public bool tHBExt = false;
public Animator ammoniteAnimator;
public Animator coelophysisAnimator;
public Animator therizinosaurusAnimator;
public int chiselUseRequirement;
// Start is called before the first frame update
void Start()
{

}

// Update is called once per frame
void Update()
{
    if(ammoniteHasBeenExposed == true)
    {
        ammoniteAnimator.SetTrigger("FossilMined");
    }
    if(ammoniteHasBeenCleaned == true)
    {
        ammoniteAnimator.SetTrigger("FossilCleaned");
    }

    if (coelophysisHasBeenExposed == true)
    {
        coelophysisAnimator.SetTrigger("FossilMined");
    }
    if (coelophysisHasBeenCleaned == true)
    {
        coelophysisAnimator.SetTrigger("FossilCleaned");
    }
    if(coelophysisHasBeenExtracted == true)
    {
        coelophysisAnimator.SetTrigger("FossilExtracted");
    }
    if(tHBExp == true)
    {
        therizinosaurusAnimator.SetTrigger("FossilMined");
    }
    if(tHBC == true)
    {
        therizinosaurusAnimator.SetTrigger("FossilCleaned");
    }
}
```



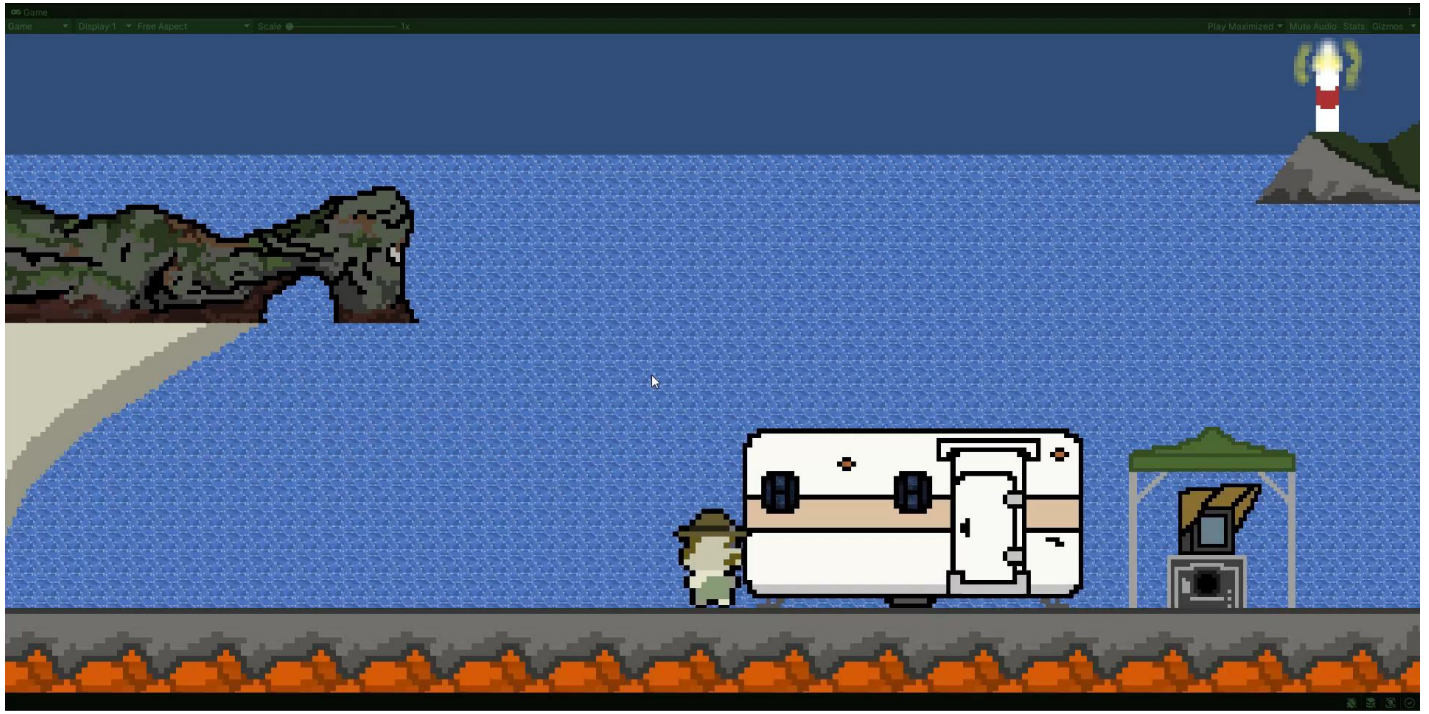


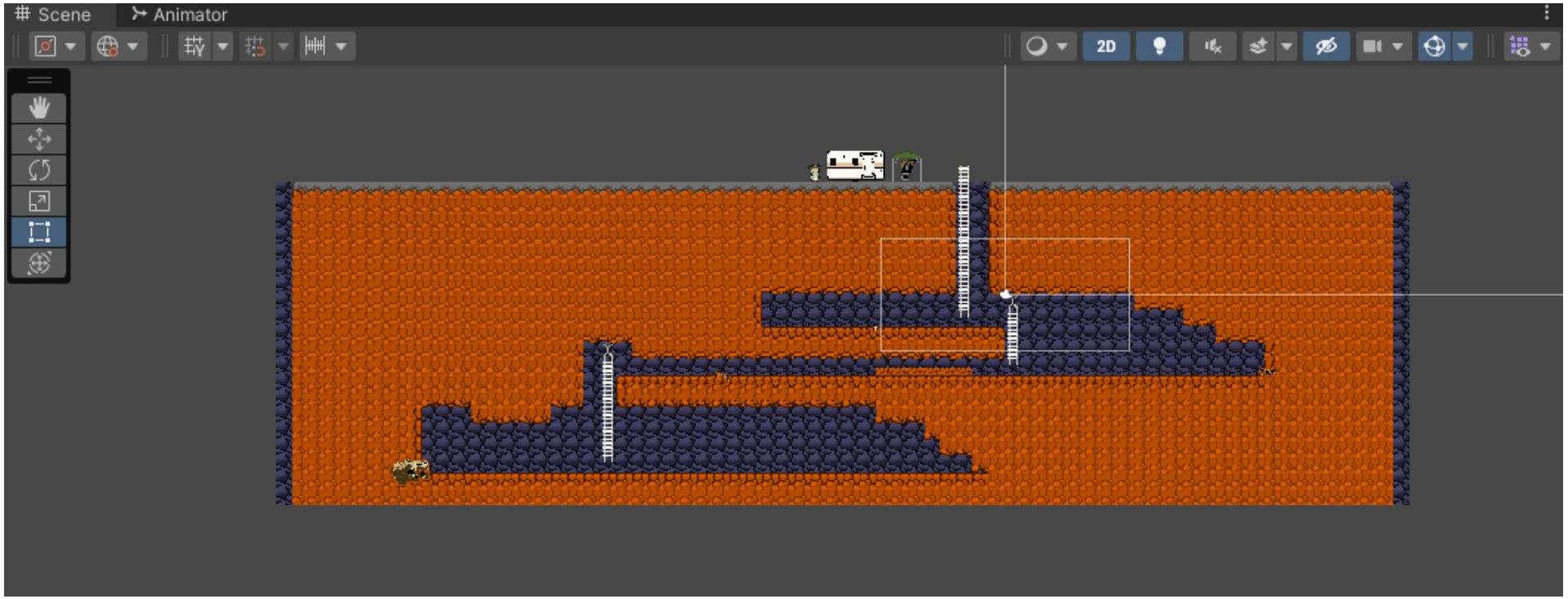
FOSSIL ANIMATION ART – DAY 68

- I drew the 1st, 2nd, and 4th frame of the animation for the Dunkleosteus as I had not done so already – using the pencil tool to draw the excess rock, then the pencil and bucket tool to make the outer rock, and finally erasing the eye socket (as this will be the fossil sample that the player will extract)

LEVEL DESIGN AND ASSET IMPLEMENTATION – DAY 68

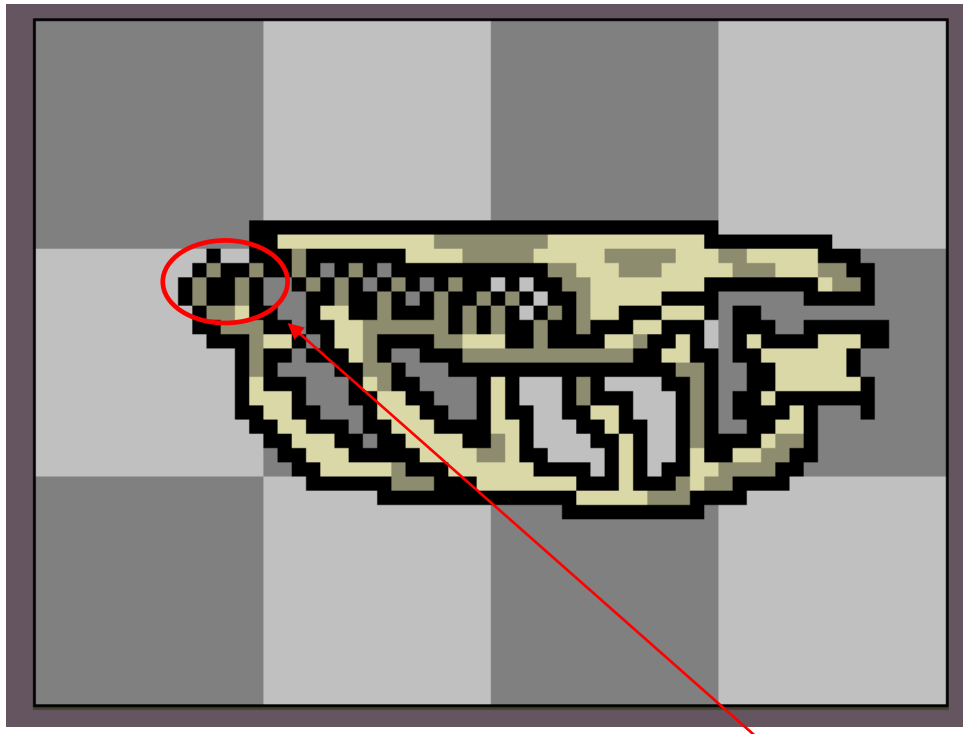
- Today I built the rest of the cave environment and imported the Dunkleosteus fossil into the depths of the cave, which isn't animated yet (the ladders are all active for testing purposes)





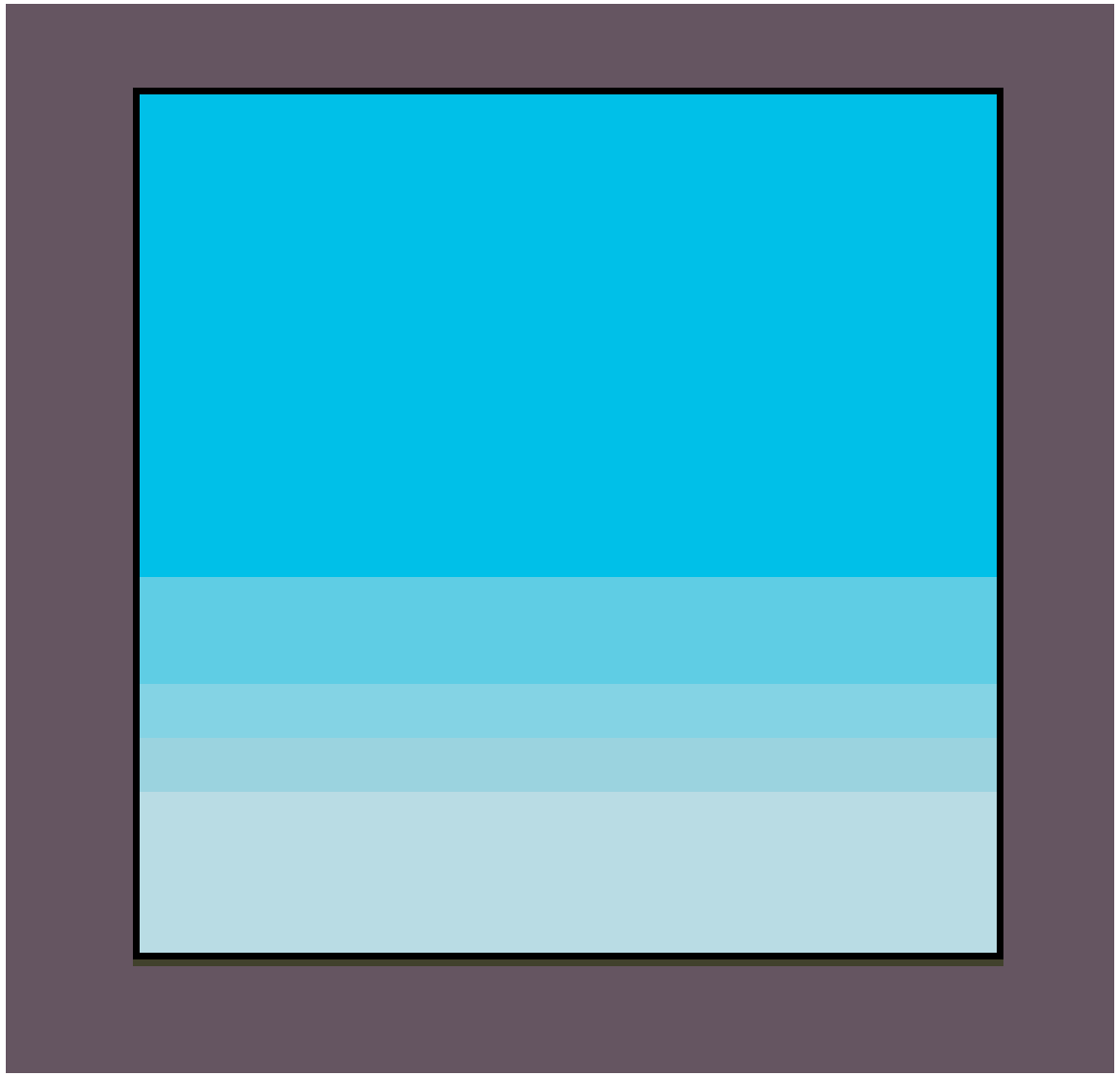
LEVEL DESIGN AND ASSET IMPLEMENTATION – DAY 68

- Here is the entire cave in the scene view, after today's work.



FOSSIL ANIMATION ART – DAY 69

- I removed the front two teeth from the Coelophysis fossil to create the final frame for after the player extracts the fossil sample – the new frame is shown on the right.

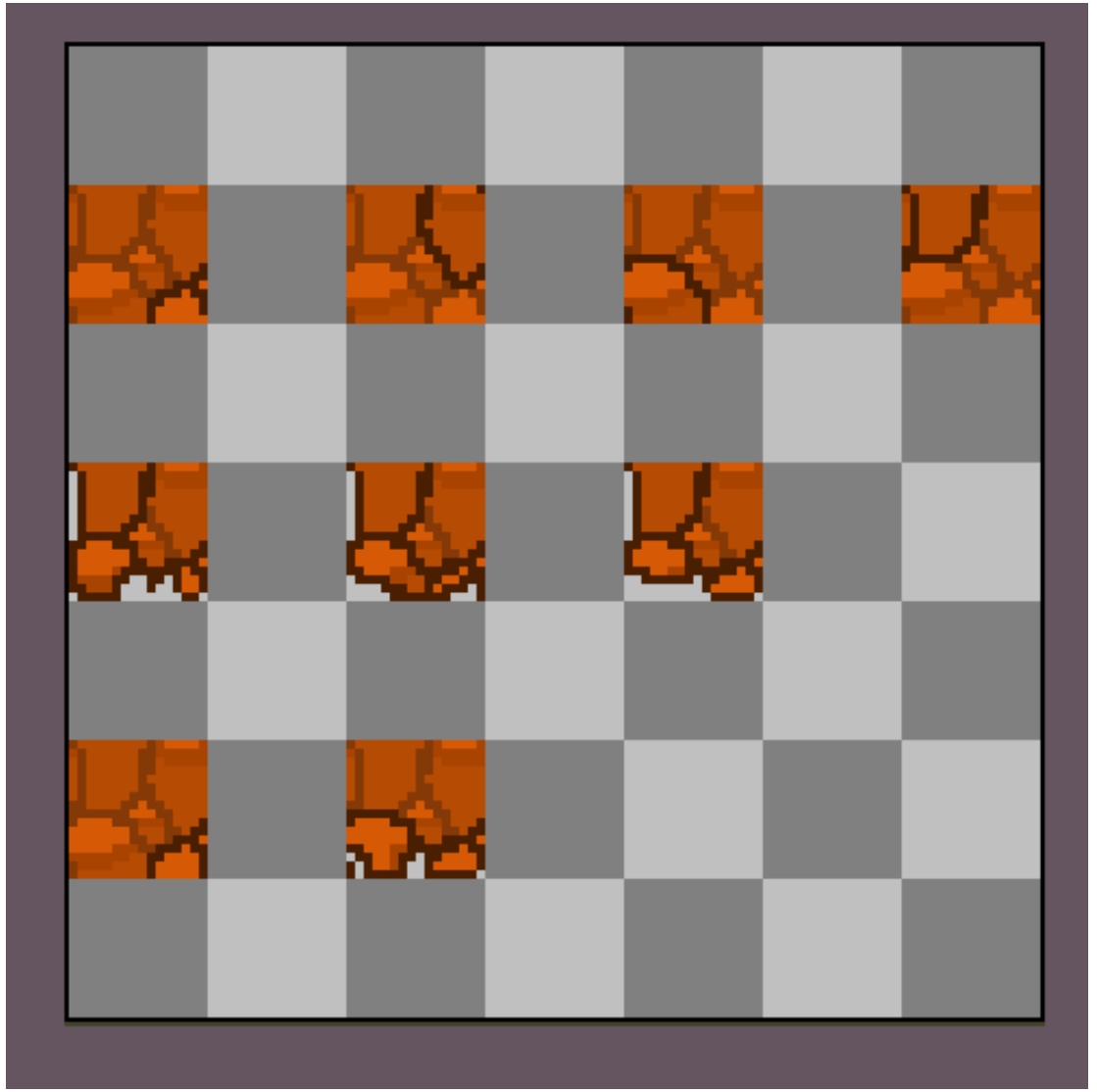


SKY TILE ART – DAY 69

- I made the sky tile with a blue lighter than the ocean tile, then faded the blue into white to give the effect of the horizon.

CAVE TILE ART – DAY 69

- After looking at what tiles don't connect correctly, I copied some of the existing tiles over to this new canvas and drew over some of the outlines with the outer border rock color – while changing the shape of some of the bottom corner tiles.



PROBLEM SOLVING AND ANIMATION – DAY 69

- I added and animated the fossil bag onto the UI canvas – it tears when you fail the QTE, but the code doesn't work properly yet as you can get a new bag at the caravan that doesn't work.
- I animated the Dunkleosteus (not shown in this video to keep it shorter) and updated the final frame of the Coelophysis extracted animation to the fossil without its front teeth.
- The scan fossil text still shows up once when the bag is torn, this shouldn't happen.
- I cleaned up the corners of the cave tile set by adding in the new tiles.

PROBLEM SOLVING AND PROGRAMMING – DAY 69

- I got the fossil bag to tear at the correct time by doing so when the cool downs variable is greater than the wins variable – each time the player advances the wins variable increases by one, and the same goes for the cool down variable when a coroutine ends – which happens when the coroutine is called and the player has not completed a QTE successfully. The cool down and win variables are reset when the player goes up to the caravan, the bag is also returned to its original state.

```
    if (coolDowns > wins)
    {
        playerControllerScript.bagTorn = true;
        bagAnim.SetBool("BagTorn", true);
        Debug.Log("Torn");
    }

    if(playerControllerScript.atCaravan == true)
    {
        coolDowns = 0;
        wins = 0;
        bagAnim.SetBool("BagTorn", false);
    }

}

public void AdvancePlayer()
{
    tunnelCounter++;
    Debug.Log(tunnelCounter);

}

IEnumerator TunnelCoolDown()
{
   if (cooledDown == true)
    {
        cooledDown = false;
        if (stagesComplete == 0 && playerControllerScript.crawlingF == true)
        {
            yield return new WaitForSeconds(2);
        }
        if(stagesComplete == 3 && playerControllerScript.crawlingB == true)
        {
            yield return new WaitForSeconds(2);
        }

        quickTimeButtons.gameObject.SetActive(true);
        float randomX = Random.Range(0, 1057*1.67f);
        float randomY = Random.Range(0, 483*1.55f);
        quickTimeButtons.gameObject.transform.position = new Vector2(randomX, randomY);
        yield return new WaitForSeconds(2.5f);
        quickTimeButtons.gameObject.SetActive(false);
        tunnelCounter = 0;
        yield return new WaitForSeconds(2);
        cooledDown = true;
        coolDowns++;


    }
}
```

PROGRAMMING – DAY 69

- The final fossil was implemented the same ways as the previous fossils were implemented, with its own variables.

```
s Files
    public bool tHBC = false;
    public bool tHBExt = false;
    public bool dHBExp = false;
    public bool dHBC = false;
    public bool dHBExt = false;
    public Animator ammoniteAnimator;
    public Animator coelophysisAnimator;
    public Animator therizinosaurusAnimator;
    public Animator dunkleosteusAnimator;
    public int chiselUseRequirement;
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if(ammoniteHasBeenExposed == true)
        {
            ammoniteAnimator.SetTrigger("FossilMined");
        }
        if(ammoniteHasBeenCleaned == true)
        {
            ammoniteAnimator.SetTrigger("FossilCleaned");
        }

        if (coelophysisHasBeenExposed == true)
        {
            coelophysisAnimator.SetTrigger("FossilMined");
        }
        if (coelophysisHasBeenCleaned == true)
        {
            coelophysisAnimator.SetTrigger("FossilCleaned");
        }
        if(coelophysisHasBeenExtracted == true)
        {
            coelophysisAnimator.SetTrigger("FossilExtracted");
        }
        if(tHBExp == true)
        {
            therizinosaurusAnimator.SetTrigger("FossilMined");
        }
        if(tHBC == true)
        {
            therizinosaurusAnimator.SetTrigger("FossilCleaned");
        }
        if(dHBExp == true)
        {
            dunkleosteusAnimator.SetTrigger("FossilMined");
        }
        if(dHBC == true)
        {
            dunkleosteusAnimator.SetTrigger("FossilCleaned");
        }
        if(dHBExt == true)
        {
            dunkleosteusAnimator.SetTrigger("FossilExtracted");
        }
    }
}
```

```
if (nearDunk == true && fossilManagerScriptD.dHBC == true && fossilManagerScriptD.dHBExt == false)
{
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptD.dHBExt = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        collectSampleText.gameObject.SetActive(false);
    }
}

if (atScanner == true && hasFossilSample == true)
{
    scanSampleText.gameObject.SetActive(true);
    if(Input.GetKeyDown(KeyCode.S))
    {
        Debug.Log("journal info");
        fossilsObtained++;
        hasFossilSample = false;
        scanSampleText.gameObject.SetActive(false);
        if(fossilManagerScriptA.ammoniteHasBeenExtracted == true)
        {
            ammoniteInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptC.coelophysisHasBeenExtracted == true)
        {
            coelophysisInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptT.tHBExt == true)
        {
            theriInfoText.gameObject.SetActive(true);
        }
        if(fossilManagerScriptD.dHBExt == true)
        {
            dunkInfoText.gameObject.SetActive(true);
        }
    }
```

```
// assigns the horizontal axis input (A and D keys) to the horizontal axis variable
horizontalInput = Input.GetAxis("Horizontal");
// assigns the vertical axis input (W and S keys) to the vertical axis variable
verticalInput = Input.GetAxis("Vertical");
if(isOnLadder == false && isCrawling == false && journalClosed == true)
{
    // move horizontally (x axis) with the A and D keys over time each second at a speed of 5
    transform.Translate(Vector2.right * horizontalInput * Time.deltaTime * speed);
    playerAnimator.SetFloat("Speed", Mathf.Abs(horizontalInput));
}
if (horizontalInput > 0 && !facingRight)
{
    Flip();
}
if (horizontalInput < 0 && facingRight)
{
    Flip();
}

if(isOnLadder == true && journalClosed == true)
{
    // move vertically (y axis) with the W and S keys over time each second at a speed of 10
    transform.Translate(Vector2.up * verticalInput * Time.deltaTime * climbSpeed);
}
```

PROGRAMMING AND PROBLEM SOLVING – DAY 69

- I also set a journal closed variable true when you click on the button that closes it, and false when you press J to open it – and now only allow the player to move if the journal is not open, I've also only allowed the tools to be used if the journal is closed.

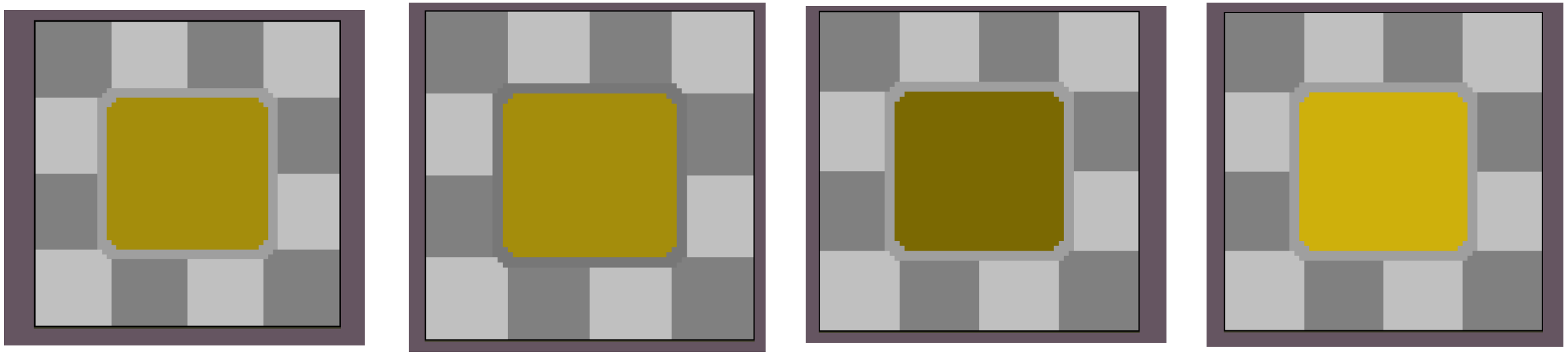
CONTENTS – WEEK 11

ART

PROGRAMMING

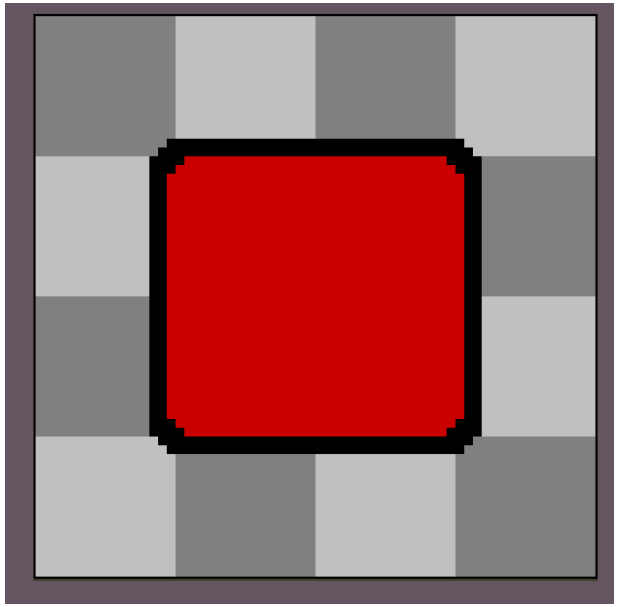
FEEDBACK

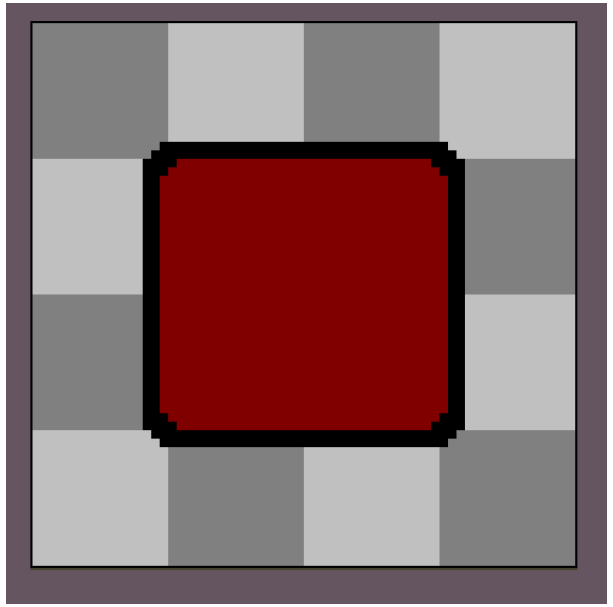
MILESTONE

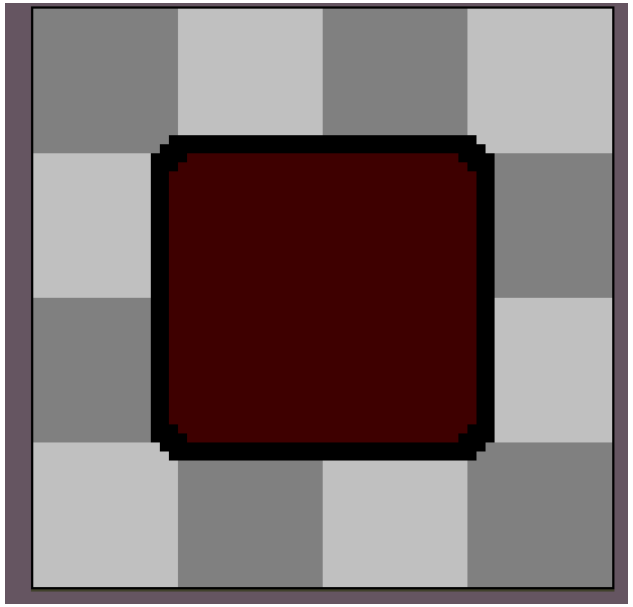


UI ART – DAY 70

- I went through multiple color iterations of the UI text panel, trying out different border and interior colors, before coming to the final asset – on the far left.





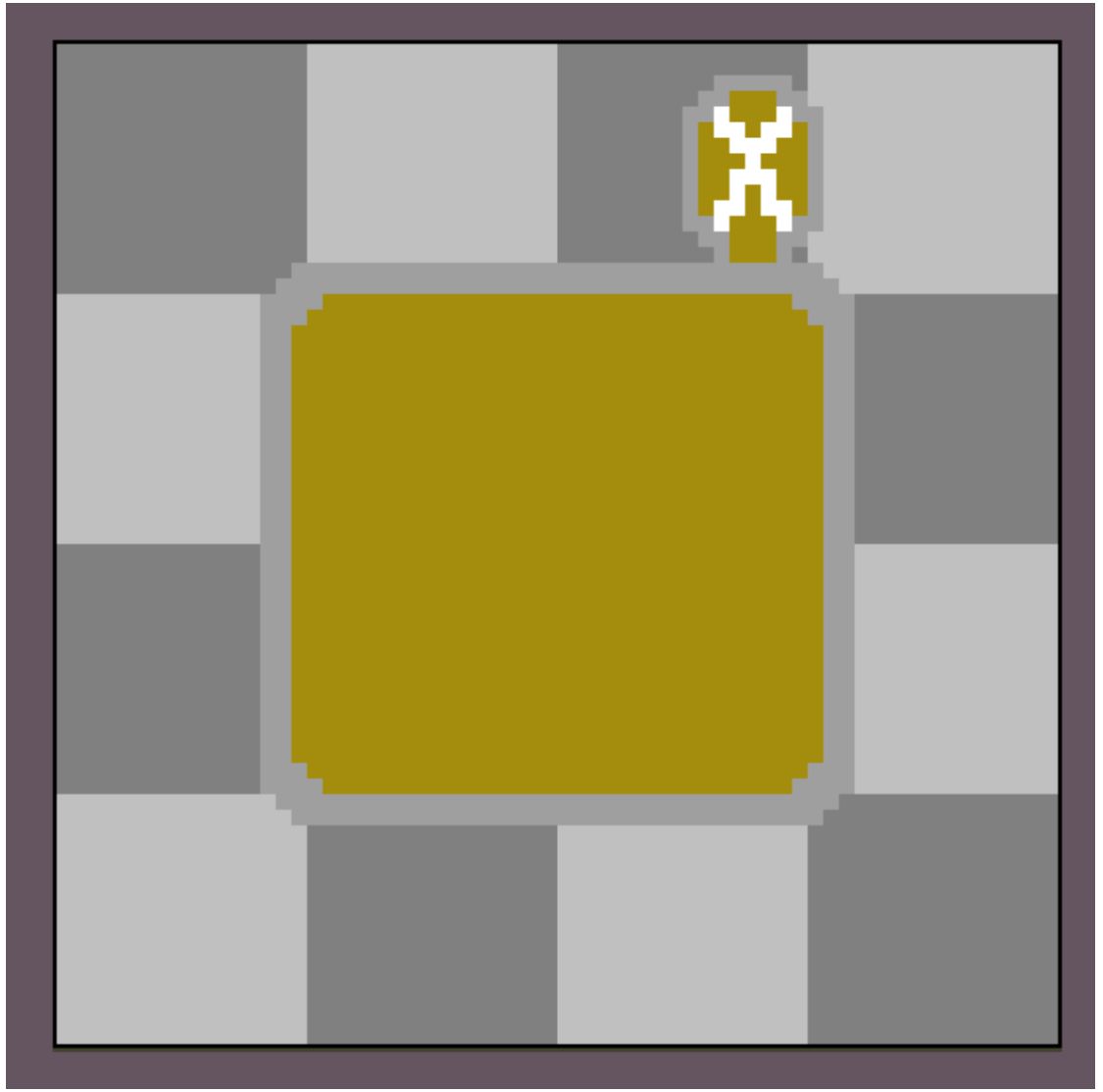


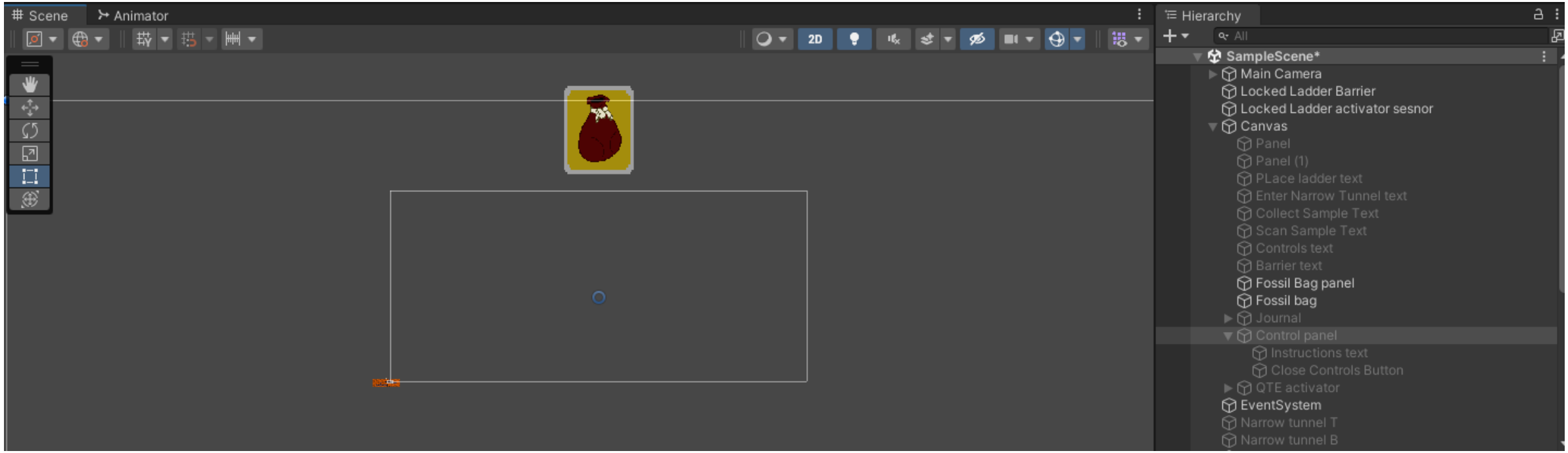
UI ART – DAY 70

- I did the same for this UI panel, that is for the quick time event button and the text that lets the player know they don't have enough fossils to proceed, only changing the interior color as I knew the border would be black – the panel shown in the middle is the final version that I decided on.

UI ART – DAY 70

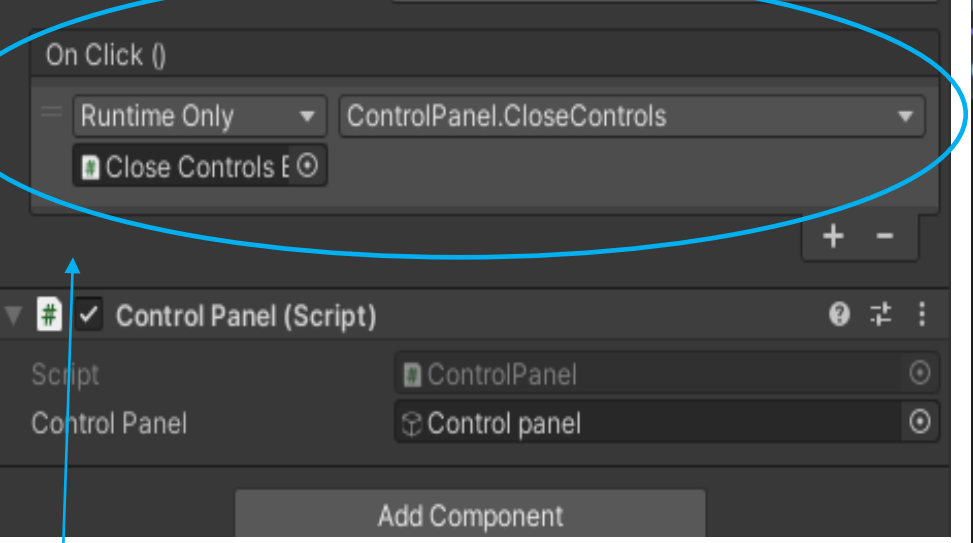
- I made the controls panel by tweaking the original UI panel by adding a button for closing it, as the controls will be accessible via an input at the caravan.





UI CANVAS – DAY 70

- I positioned the active fossil bag and its own panel in the middle and top of the screen, all the other UI elements are in-active by default as they pop up when the player does something – which are also all centered on the screen.
- I added these UI artworks into the canvas as UI images, which I replaced with my own sprites (panels, fossil bag, journal, etc)



PROGRAMMING – DAY 70

- I did the same as the journal for closing the controls panel – calling a method that sets the controls in-active when the button I positioned on the cross is clicked and detected by the listener.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class ControlPanel : MonoBehaviour
{
    private Button closeControlsButton;
    public GameObject controlPanel;
    private PlayerController playerControllerScript;
    // Start is called before the first frame update
    void Start()
    {
        closeControlsButton = GetComponent<Button>();
        closeControlsButton.onClick.AddListener(CloseControls);
        playerControllerScript = GameObject.Find("Player").GetComponent<PlayerController>();
    }

    // Update is called once per frame
    void Update()
    {

    }

    public void CloseControls()
    {
        controlPanel.gameObject.SetActive(false);
        playerControllerScript.controlsClosed = true;
    }
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 70

- When the bag is torn in the game manager script, I set a “can be extracted” variable false (which can never be set back to being true) along with the has been extracted variable for each fossil for when each related fossil has been extracted and the bag is torn.

GameManager.cs

```
if (coolDowns > wins)
{
    playerControllerScript.bagTorn = true;
    playerControllerScript.hasFossilSample = false;
    bagAnim.SetBool("BagTorn", true);
    if (fossilManagerScriptC.coelophysisHasBeenExtracted == true)
    {
        fossilManagerScriptC.coelophysisHasBeenExtracted = false;
        playerControllerScript.cCanBeExtracted = false;
    }
    if (fossilManagerScriptT.tHBExt == true)
    {
        fossilManagerScriptT.tHBExt = false;
        playerControllerScript.tCanBeExtracted = false;
    }
    if (fossilManagerScriptD.dHBExt == true)
    {
        fossilManagerScriptD.dHBExt = false;
        playerControllerScript.dCanBeExtracted = false;
    }
    Debug.Log("Torn");
}
```

```
}

if (nearCoelophysis == true && fossilManagerScriptC.coelophysisHasBeenCleaned == true && fossilManagerScriptC.coelophysisHasBeenExtracted == false && cCanBeExtracted == true && hasFossilSample == false && bagTorn == false)
{
    panel.gameObject.SetActive(true);
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptC.coelophysisHasBeenExtracted = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        panel.gameObject.SetActive(false);
        collectSampleText.gameObject.SetActive(false);
    }
}

if (nearTheri == true && fossilManagerScriptT.tHBC == true && tCanBeExtracted == true && hasFossilSample == false && bagTorn == false)
{
    panel.gameObject.SetActive(true);
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptT.gameObject.SetActive(false);
        fossilManagerScriptT.tHBExt = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        panel.gameObject.SetActive(false);
        collectSampleText.gameObject.SetActive(false);
    }
}

if (nearDunk == true && fossilManagerScriptD.dHBC == true && fossilManagerScriptD.dHBExt == false && dCanBeExtracted == true && hasFossilSample == false && bagTorn == false)
{
    panel.gameObject.SetActive(true);
    collectSampleText.gameObject.SetActive(true);
    if (Input.GetKeyDown(KeyCode.E))
    {
        fossilManagerScriptD.dHBExt = true;
        hasFossilSample = true;
        chiselUses -= chiselUses;
        panel.gameObject.SetActive(false);
        collectSampleText.gameObject.SetActive(false);
    }
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 70

- I added three variables on the end of each if statement for extracting the fossil samples, now only allowing extraction when the bag has not been torn with the fossil in it, when you don't currently have another fossil sample in possession, and when the bag isn't torn in general.
- I've also activated and deactivated the UI panel for every activation and deactivation of any UI text.

```
if(Input.GetKeyDown(KeyCode.J))
{
    journal.gameObject.SetActive(true);
    journalClosed = false;
}

if(Input.GetKeyDown(KeyCode.C) && atCaravan == true)
{
    panel.gameObject.SetActive(false);
    controlsText.gameObject.SetActive(false);
    controlPanel.gameObject.SetActive(true);
    controlsClosed = false;
}
```

PROGRAMMING – DAY 70

- I activated the controls panel when the player is at the caravan and presses the C key, while also deactivating the text prompt that tells the player to press C – along with its panel.

```
if (Input.GetMouseButtonDown(0) && pickaxeSwung == false && journalClosed == true && controlsClosed == true)
{
    playerAnimator.SetBool("PickaxeSwung", true);
    pickaxeSwung = true;
    StartCoroutine(PickCoolDown());
}

if (Input.GetMouseButtonDown(1) && chiselUsed == false && journalClosed == true && controlsClosed == true)
{
    playerAnimator.SetBool("ChiselUsed", true);
    chiselUsed = true;
    if(nearAmmonite == true && fossilManagerScriptA.ammoniteHasBeenExposed == true)
    {
        chiselUses++;
    }
    if(nearCoelophysis == true && fossilManagerScriptC.coelophysisHasBeenExposed == true)
    {
        chiselUses++;
    }
    if(nearTheri == true && fossilManagerScriptT.tHBExp == true)
    {
        chiselUses++;
    }
    if(nearDunk == true && fossilManagerScriptD.dHBExp == true)
    {
        chiselUses++;
    }
    StartCoroutine(ChiselCoolDown());
}
```

PROBLEM SOLVING AND PROGRAMMING – DAY 70

- I now also only allow the player to move or use tools when the controls are closed, done the same way as the journal closed variable.

PROBLEM SOLVING AND PROGRAMMING – DAY 70

- I made an IE numerator that waits for half a second and sets the gravity scale to one and sets everything involving climbing false, this is called after the player is dismounted from the ladder by the bumper – just to make sure the player isn't sill in the climbing state.
- The bumper adds an impulse force right instead of left for the bumper tagged "Bumper 2", which is for the ladder after the narrow tunnel – where you dismount in the opposite direction to the other two ladders.

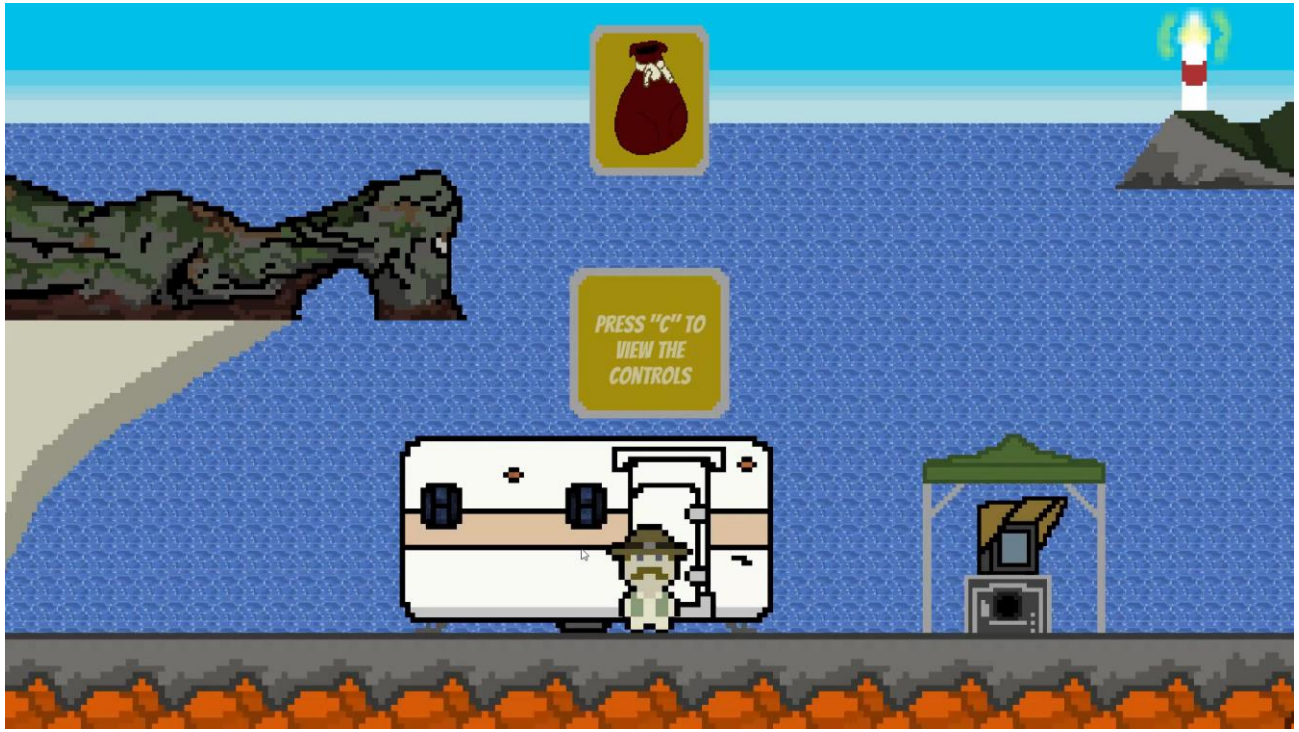
```
IEnumerator DismountCoolDown()
{
    yield return new WaitForSeconds(0.5f);
    playerRb.gravityScale = 1;
    isOnLadder = false;
    playerAnimator.SetBool("IsClimbing", false);
}

// method for entering trigger colliders
private void OnTriggerEnter2D(Collider2D other)
{
    if(other.gameObject.CompareTag("Ladder"))
    {
        isOnLadder = true;
        playerAnimator.SetBool("IsClimbing", true);
        Debug.Log("ON ladderr");
        // turns iff gravity when the player goes onto the ladder
        playerRb.gravityScale = 0;
    }
    if (other.gameObject.CompareTag("Bumper"))
    {
        // turns gravity back on when the player gets off the ladder
        playerRb.gravityScale = 1;
        // makes the player jump off the ladder when they reach the top
        playerRb.AddForce(Vector2.left * bumperForce, ForceMode2D.Impulse);
        playerRb.AddForce(Vector2.down * bumperForce, ForceMode2D.Impulse);
        isOnLadder = false;
        playerAnimator.SetBool("IsClimbing", false);
        StartCoroutine(DismountCoolDown());
    }
    if (other.gameObject.CompareTag("Bumper2"))
    {
        // turns gravity back on when the player gets off the ladder
        playerRb.gravityScale = 1;
        // makes the player jump off the ladder when they reach the top
        playerRb.AddForce(Vector2.right * bumperForce, ForceMode2D.Impulse);
        playerRb.AddForce(Vector2.down * bumperForce, ForceMode2D.Impulse);
        isOnLadder = false;
        playerAnimator.SetBool("IsClimbing", false);
        StartCoroutine(DismountCoolDown());
    }
```

```
if (other.gameObject.CompareTag("Ladder Activator B"))
{
    isInLadderActivatorB = true;
    if (isLadderBPlaced == false && fossilsObtained < 3)
    {
        barrierPanel.SetActive(true);
        barrierText.gameObject.SetActive(true);
    }
    if (laddersOwned > 0 && isLadderBPlaced == false && fossilsObtained > 2)
    {
        panel.gameObject.SetActive(true);
        placeLadderText.gameObject.SetActive(true);
    }
}
```

```
if (other.gameObject.CompareTag("Ladder Activator B"))
{
    isInLadderActivatorB = false;
    barrierPanel.SetActive(false);
    barrierText.gameObject.SetActive(false);
    panel.gameObject.SetActive(false);
    placeLadderText.gameObject.SetActive(false);
    Debug.Log("+1ip");
}
```

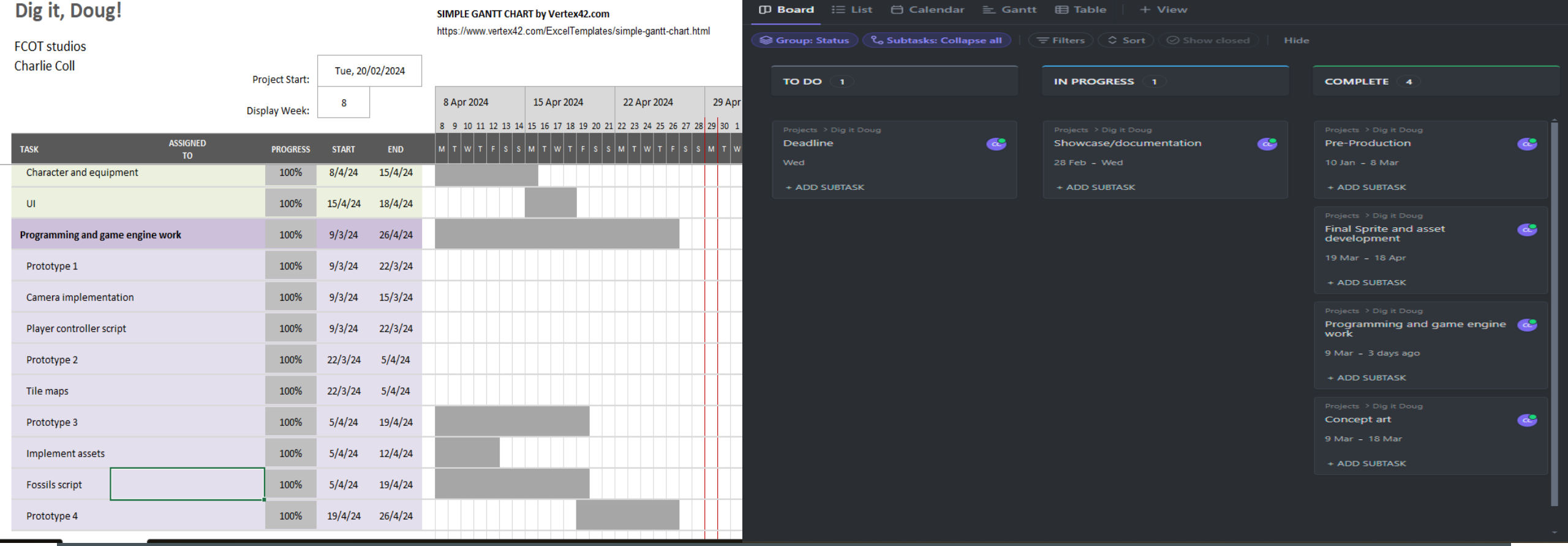
PROBLEM SOLVING AND PROGRAMMING – DAY 70

- I only set the barrier UI active when the player is at the final ladder with less than three fossils and set the UI to prompt the player to place a ladder when they are by the final ladder with more than two fossils .
- I set all these variables and UI elements false again when the player leaves the sensor for this ladder – the rest of the ladder functions as the other two ladders do.



FINAL GAMEPLAY – DAY 70

- Here is 4 and a half minutes of the final gameplay from the completed vertical slice of this game – a playthrough where all the fossils are discovered and retrieved.
- I'm also glad to say that the game plays exactly how I had planned at the start of this project – when I first thought of the idea and vision for the art, game mechanics, and educational experience.

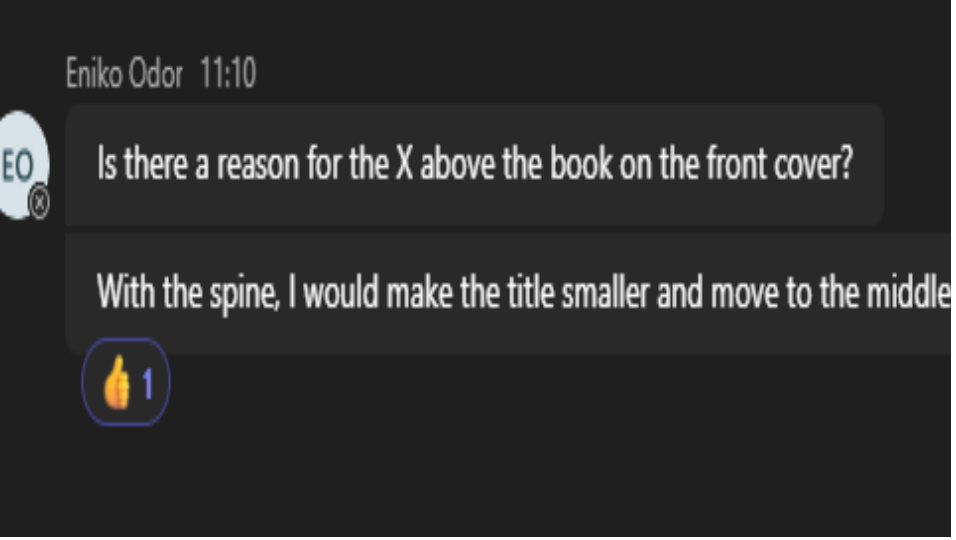


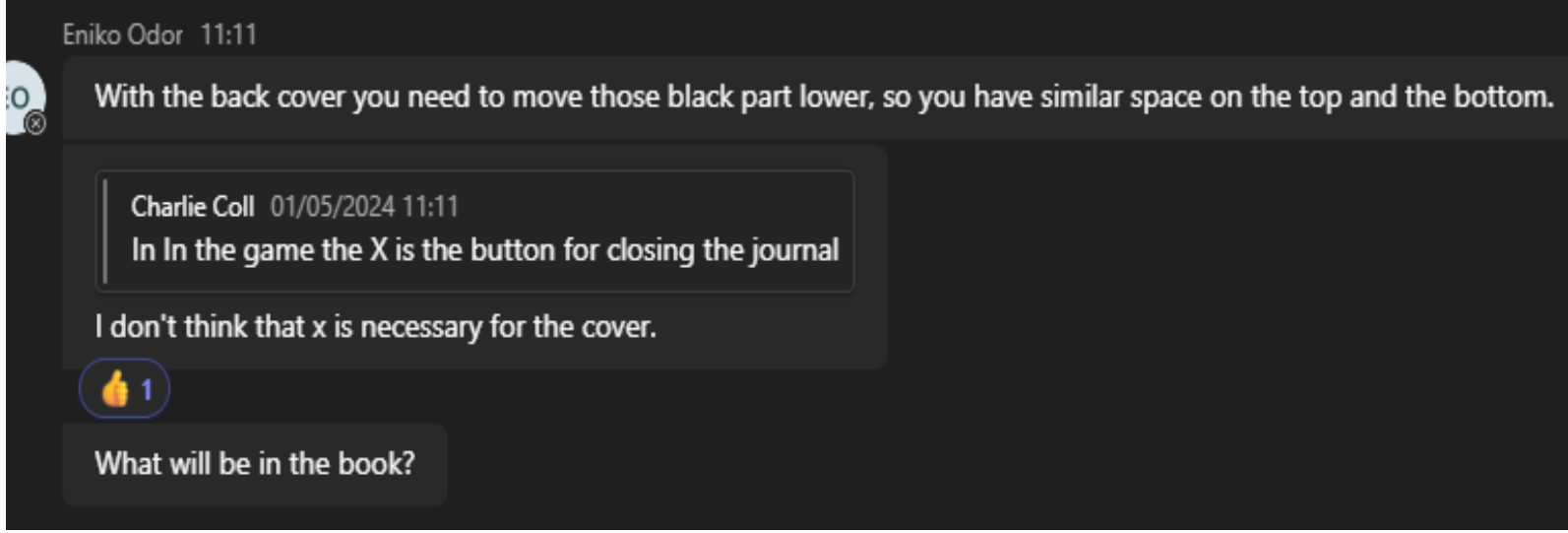
MILESTONE – DAY 70

- The final build of the game has been run and all the subtasks involving the development of the project are now marked complete.
- The deadlines for the programming and sprite phases are past due, as at the time I realized that I was going to do three prototypes instead of four (as I didn't need a whole 4th prototype for the UI) and worked on finishing the project as soon as possible – instead of setting more deadlines.

BOX ART – DAY 71

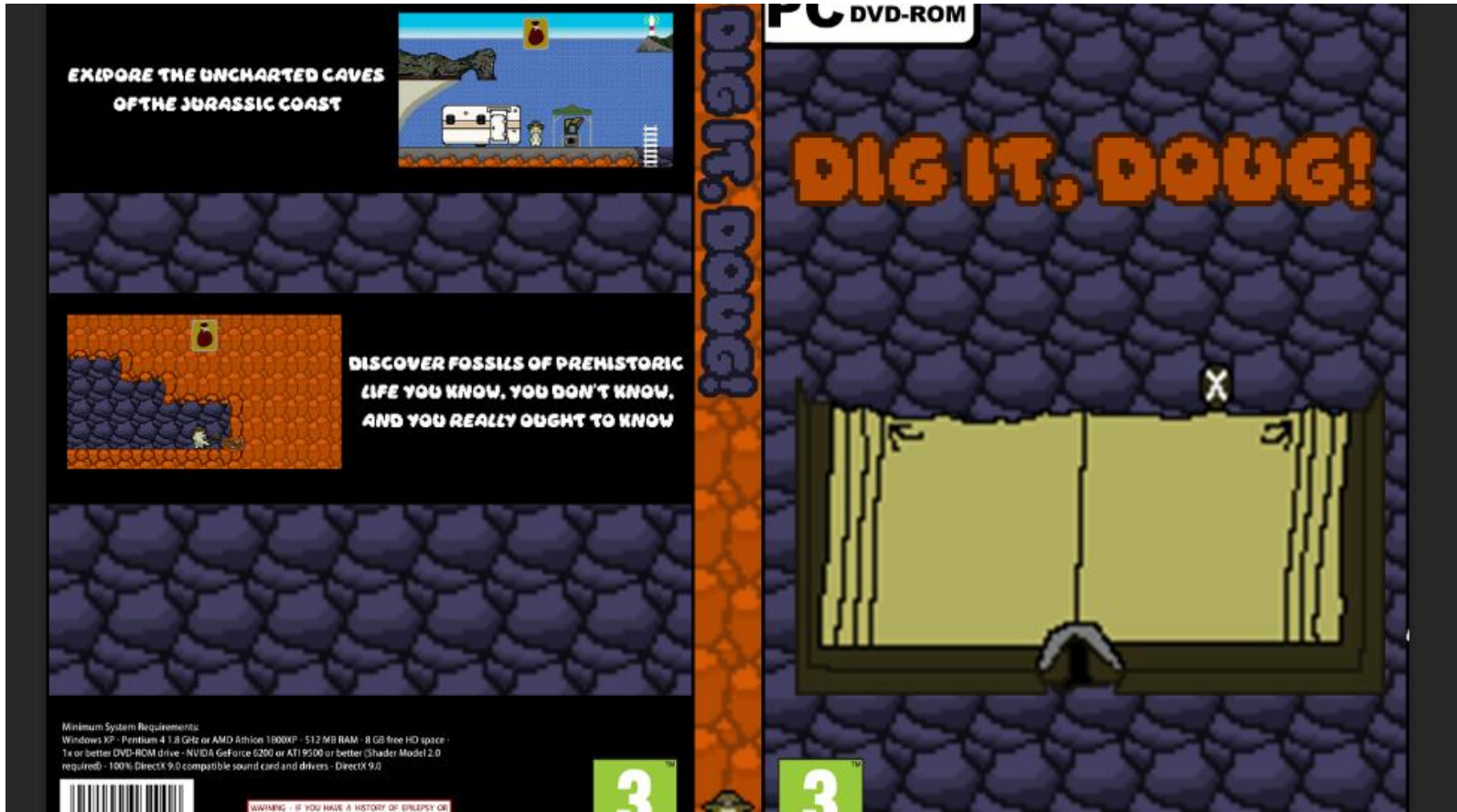
- I began the box art with a template provided by my lecturer, then I replaced the age ratings on it with the Pegi 3 rating that I had previously chosen for this game.
- I made my rock tiles into repeating pattens in photoshop, then I created new layers that I added a pattern fill mask to and used my rock tile repeating patterns with them – filling in the spine and the rest of the box art with these patterns.
- I imported my character and journal PNG files and scaled them up, however when I positioned them the quality had been reduced slightly.
- Finally, I downloaded this “Daydream” font by “DoubleGum” from “Dafont” and added the color of the rock to the text with an outline that I added with the stroke effect in photoshop.





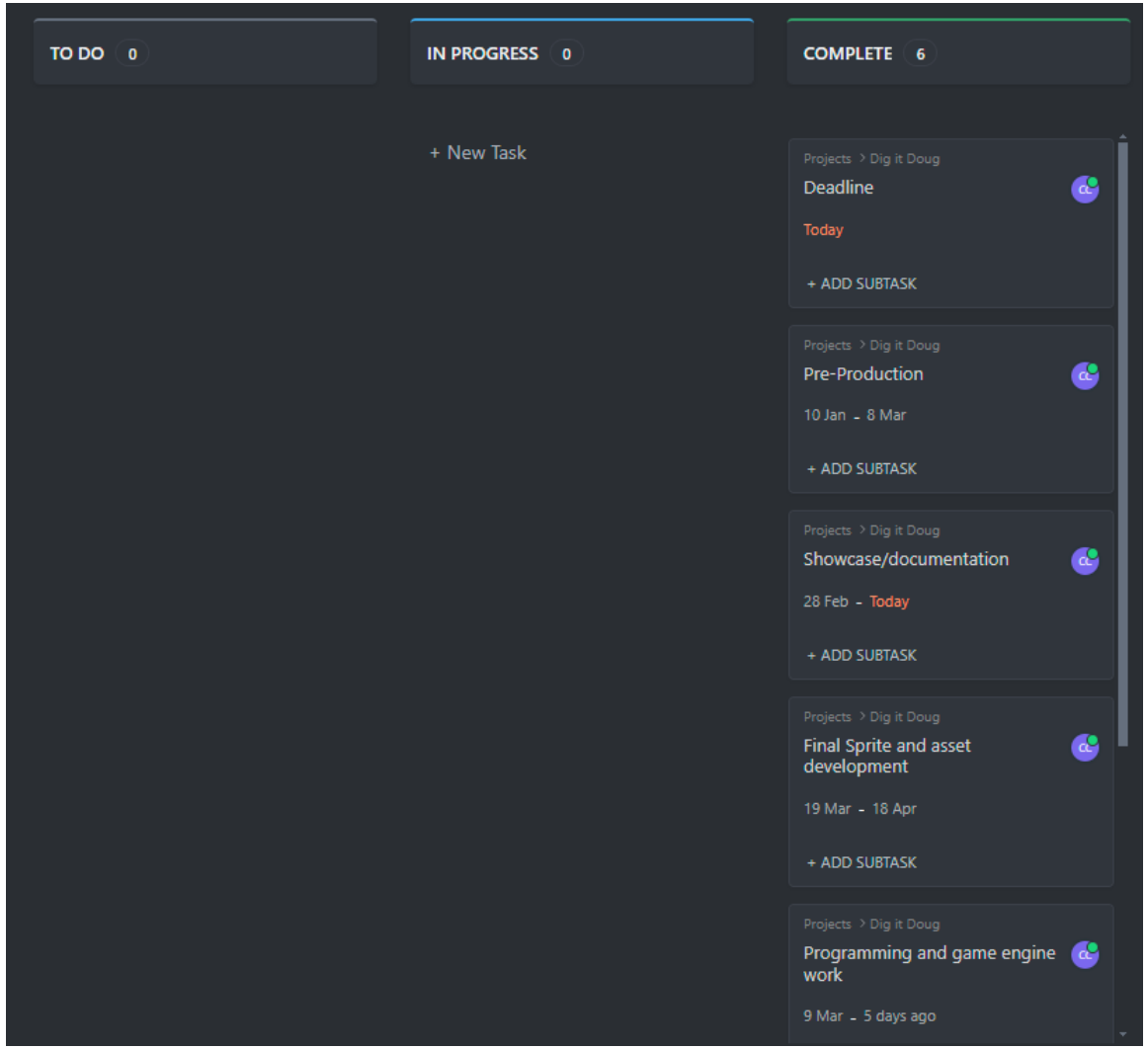
BOX ART AND FEEDBACK – DAY 72

- I added two gameplay screenshots onto the back of the box, then added my elevator pitch text next to the images and added a black border around them with the select and fill tools. I also added the title to the spine of the box, with inverted colors.
- My art lecturer responded to this iteration of the box art by saying that the title on the spine should be moved to the middle, the screenshots on the back should also be moved to the middle, and the X on the journal isn't necessary for the cover – as it's a UI element.



FINAL BOX ART – DAY 72

- I made the appropriate changes after receiving that feedback, bringing the box art to its final composition.



MILESTONE – DAY 72

- All the productions have been added to this document, meaning the project has officially come to an end!

THANKS FOR READING

BY CHARLIE COLL