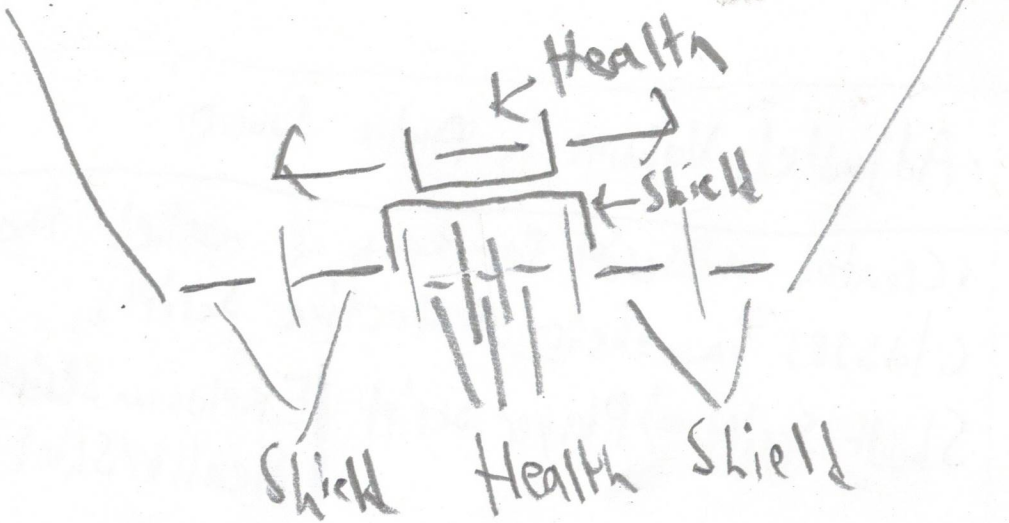
# Concepts

- Plinko Space Battle Game
- Modify Score Buckets into → Health
  → Shield




- Add moving bucket for alternate hit on Health
- Add music
- Add SFX
  - → Shooting ball of energy
  - → "Blowing" up Shield & Health
- Add Ship Sprite for Player
- Add Space BG
- Change object colors to light grey to fit tone of game

- Add Start + Win Screen → Win Screen will contain:
  - Win text
  - Victory SFX
  - → Start Screen will contain:
    - Onboarding text
    - Controls
    - Start Button

| Problems | Solutions |
| --- | --- |
| • Background image covered everything, despite it loading last in the scene | • Restarted Unity + Laptop |
| • Background music too loud | • Adjusted Volume in Audio Source |
| • Play Shoot & Explosion SFX at the correct input / time | • Created a script for each & called them in classes in their respective scripts.<br>Shoot Script => Player Script \| Explosion Script & Health/Shield Tracker |
| • Creating Separate Shield & Health Statuses | • Duplicated Health and adjusted code to fit what it's for<br>Shield => Destroys Shield Game Object<br>Health => Leads to Win Screen once 0. |
| • How to load into different scenes | • Referenced Module 4 content of Game Asset Development: 2D GFX |

ExplosionSFX.cs | PlayerShootSFX.cs | CameraFollow.cs | deductHealth.cs | HealthTracker.cs | Player.cs

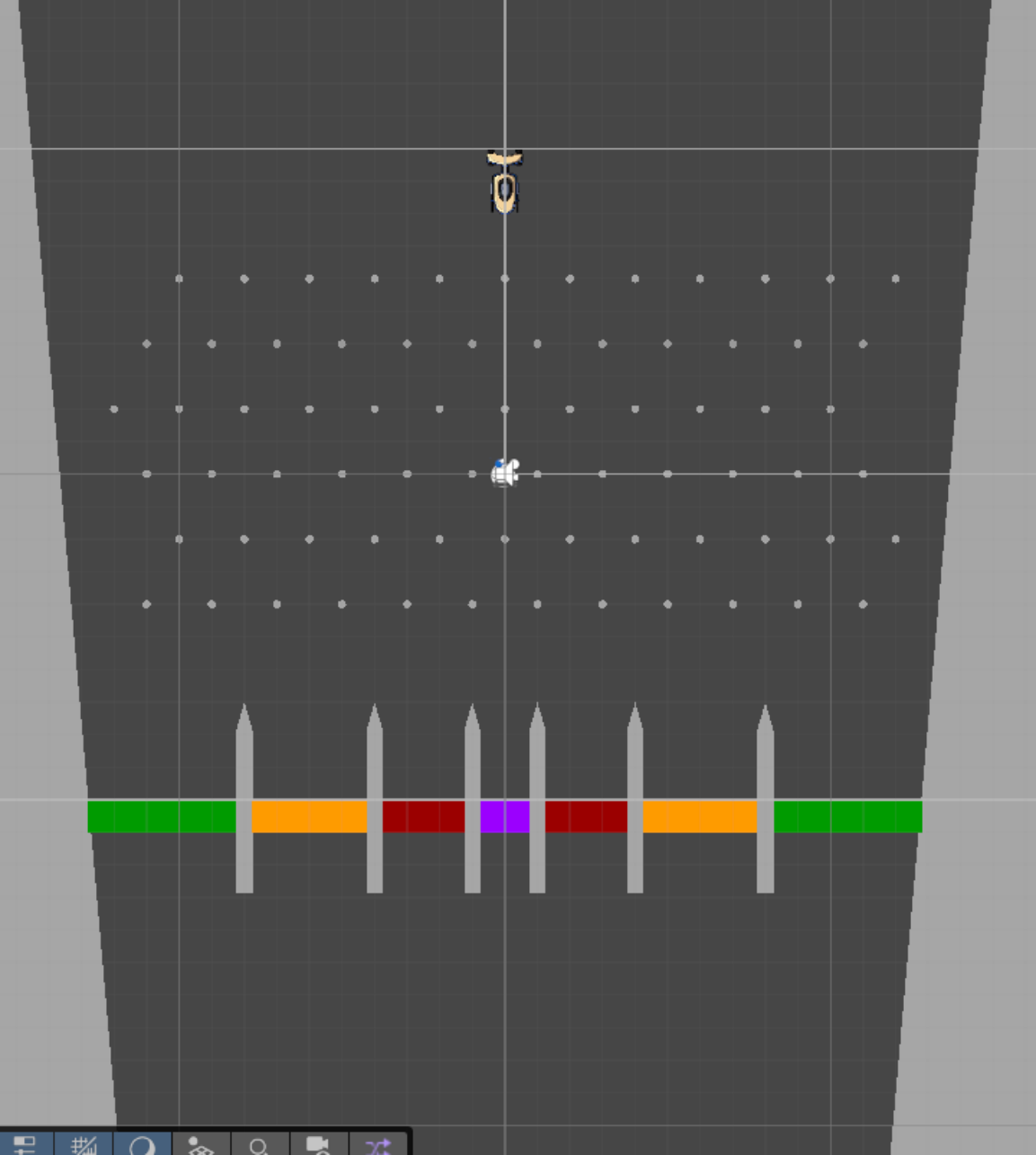
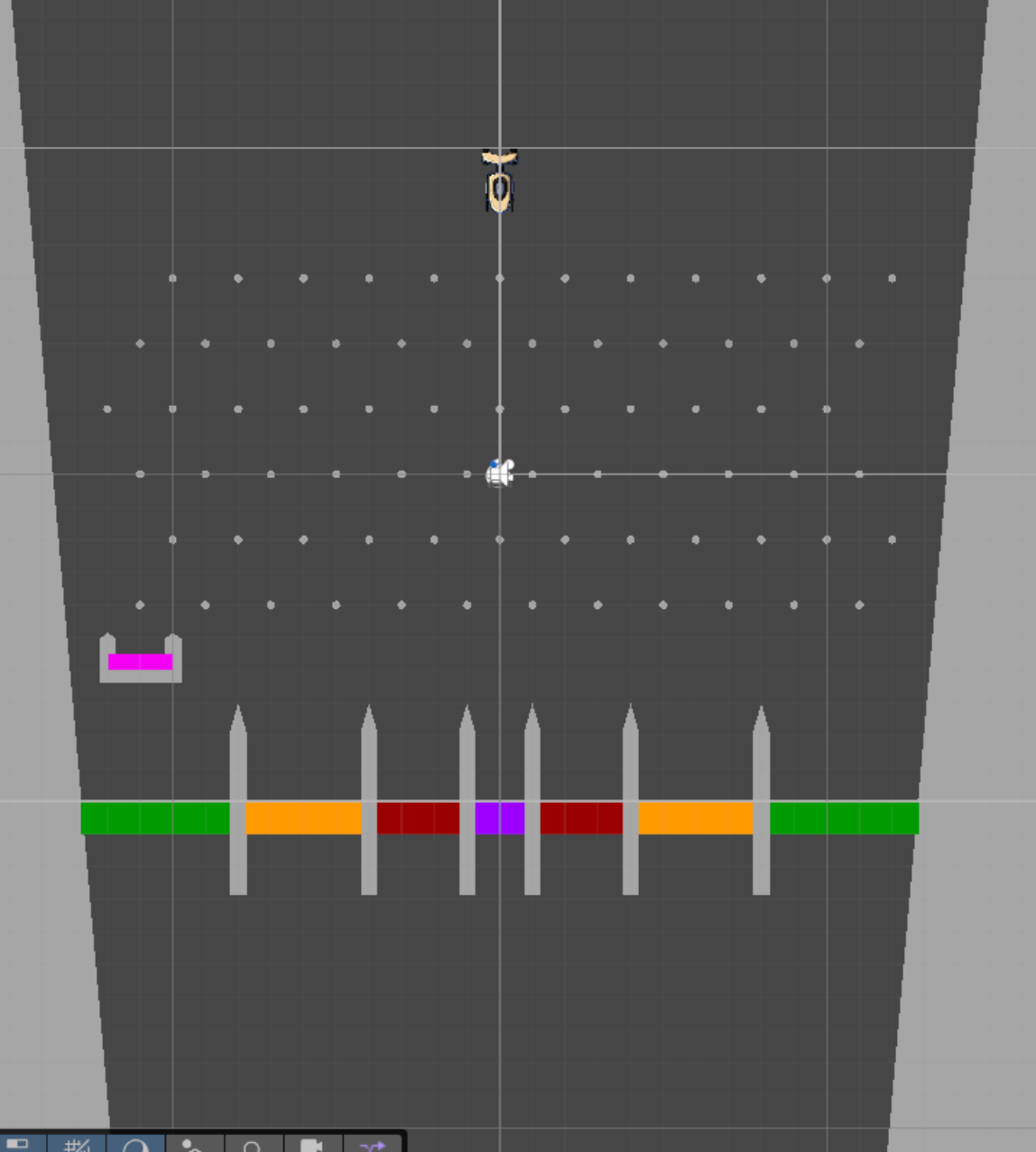
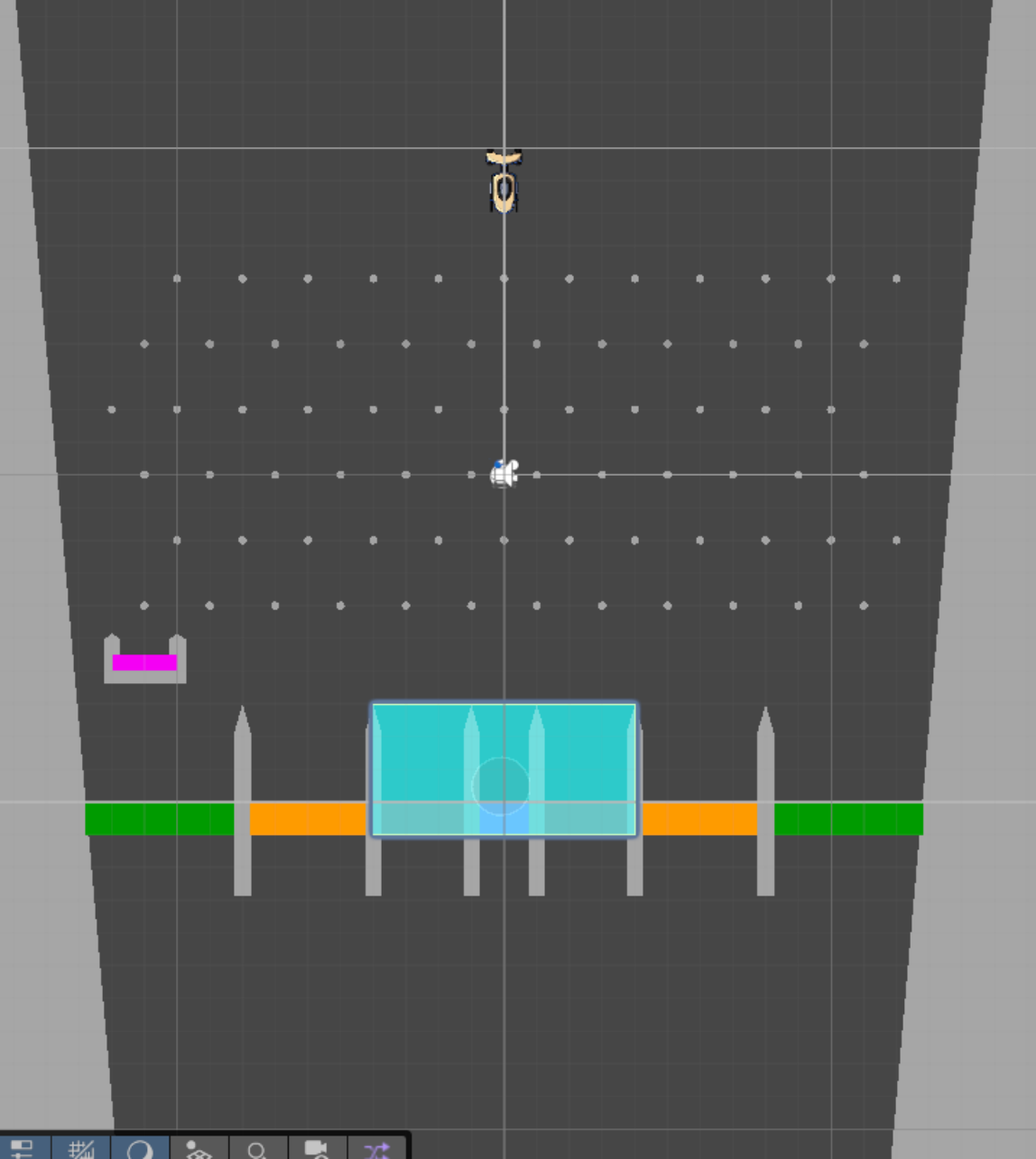
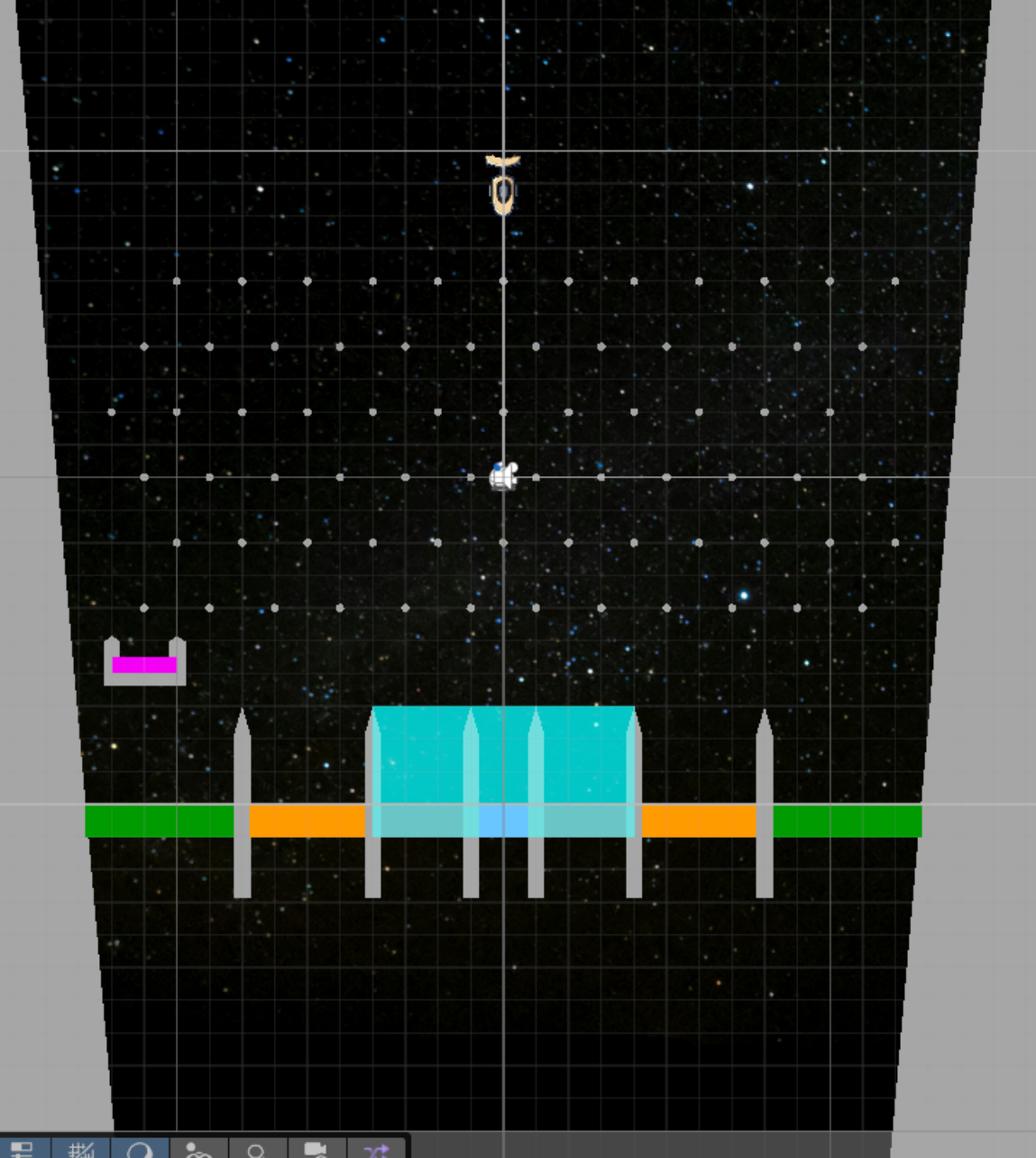
Plinko | ExplosionSFX | ExplosionSoundFX()

```
1   using UnityEngine;
2
    Unity Script (1 asset reference) | 2 references
3   public class ExplosionSFX : MonoBehaviour
4   {
5       // Call upon a source of Audio
6       public AudioSource audioSource;
        2 references
7       public void ExplosionSoundFX()
8       {
9           // Play the audio source
10          audioSource.Play();
11      }
12  }
13
```

```
{
    // Reference to the TMP_Text component for displaying shield + setting initial shield + ShieldDownSFX script
    public TMP_Text shieldDisplay;
    public GameObject shieldObject;
    public int shield = 500;

    Unity Message | 0 references
    private void Start()
    {
        UpdateShieldDisplay();
    }
    Unity Message | 0 references
    private void Update()
    {
        ZeroShield(shieldObject);
    }
    1 reference
    public void deductShield(int damage)
    {
        // Deduct shield by the damage amount
        shield -= damage;
        UpdateShieldDisplay();
    }
    2 references
    public void UpdateShieldDisplay()
    {
        // Update the shield display with leading zeros
        shieldDisplay.text = $"SHIELD: {shield,000000000}";
    }
    1 reference
    public void ZeroShield(GameObject shieldObject)
    {
        // If shield is 0 or less, destroy the Shield
        if (shield <= 0)
        {
            Destroy(shieldObject);
        }
    }
```

```
using UnityEngine;

Unity Script (1 asset reference) | 0 references
public class MovingBucket : MonoBehaviour
{
    // Define the game object to be moved
    public GameObject movingBucket;
    Unity Message | 0 references
    void Update()
    {
        // Move the bucket back and forth between -10 and 11 on the x-axis
        movingBucket.transform.position = new Vector3(Mathf.PingPong(Time.time, 21) + -10, movingBucket.transform.position.y, movingBucket.transform.position.z);
    }
}

```

StartGame.cs | MovingBucket.cs | ShieldTracker.cs | deductHealth.cs | DeductShield.cs | HealthTracker.cs | ExplosionSFX.cs

Plinko | StartGame | StartTheGame()

```
1   using UnityEngine;
2   using UnityEngine.SceneManagement;
3
    Unity Script (1 asset reference) | 0 references
4   public class StartGame : MonoBehaviour
5   {
        0 references
6       public void StartTheGame()
7       {
8           // Load the main game scene
9           SceneManager.LoadScene("GameScene");
10      }
11  }
12
```