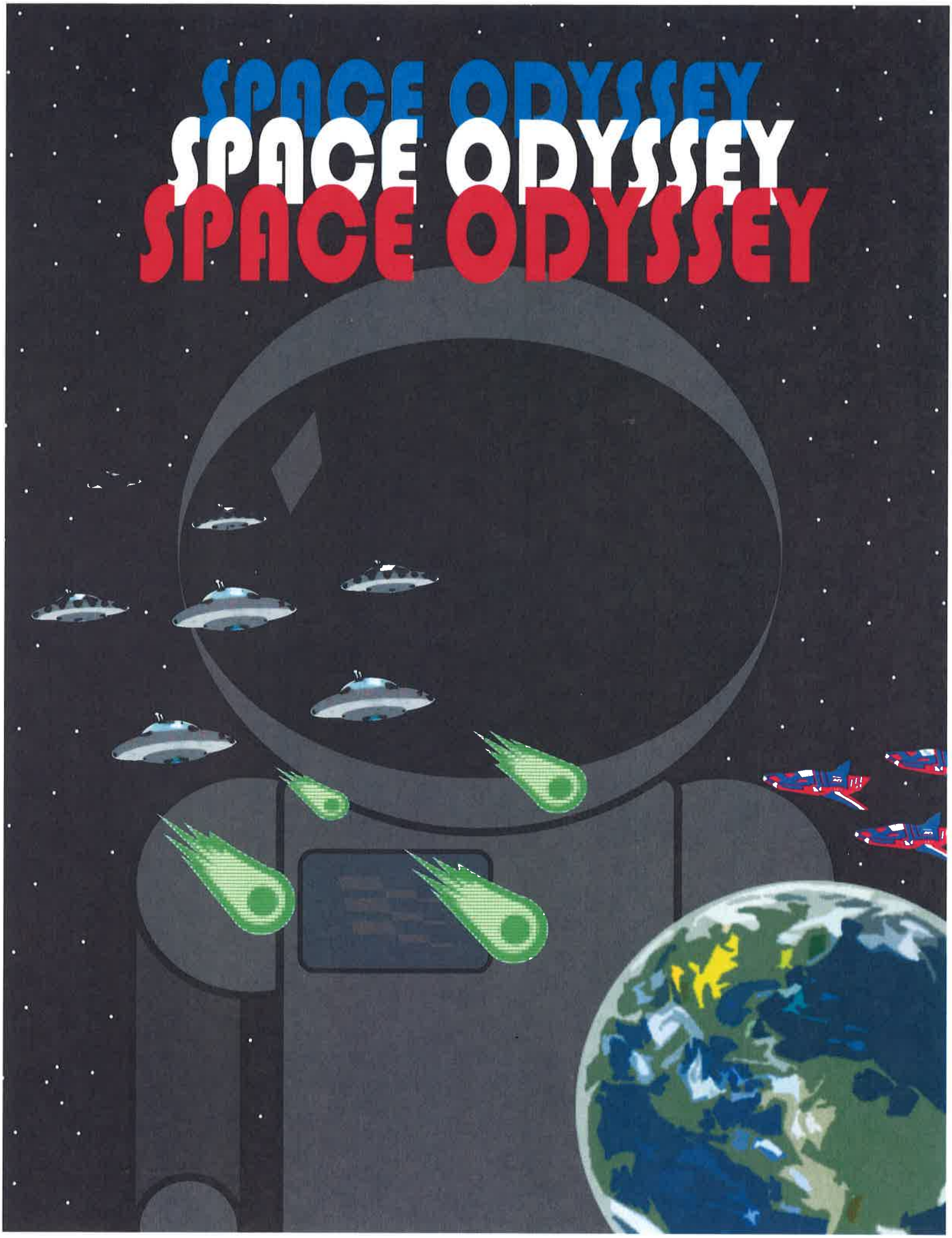

SPACE ODYSSEY
SPACE ODYSSEY
SPACE ODYSSEY

Space Odyssey

The year is 3000. Our beloved home has been taken hostage by an alien invasion. You are just a regular space cargo worker when you first learn about the invasion. However you become a hero as time unfolds. You will use a series of weapons to destroy whatever obstacle might barricade your way.

After countless enemies are defeated, several rooms cleared and defended you'll encounter the mothership and on that ship, the queen of this unknown invasive alien race. Upon defeating her, humanity is saved and fear and destruction quelled but only temporarily…

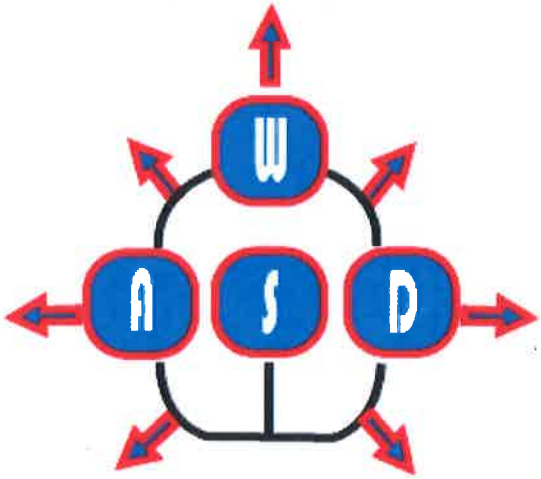
Our target audience depends with our genre for our game. This game is classified as a dungeon space RPG. Everyone from all ages are suited to play our game. All content has been thought out carefully and designed appropriately to make our game fun, compelling and intrusive. Our biggest audience could potentially be kids who are imaginative and like space fiction genre games. This game was intended for all ages.

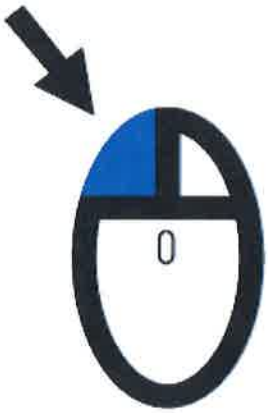
The overall purpose of the game is to clear each level and to defeat as many enemies possible while eventually reaching the final boss and putting an end to her tyrannical reign. You will use 2 weapons to defeat the enemies that you encounter. A flamethrower and a laser gun. The flamethrower will be specific to incinerate all alien lifeform. Whilst the laser gun is perfect to destabilize our robots that have been hacked by the aliens. Keeping in mind that both weapons will still do damage regardless to their weaknesses, It would be efficient to follow to these instructions for optimal performance. There are a total of 3 rooms and each room increases in difficulty slightly making the game more challenging and entertaining. you will need to clear in order to make our spaceship safe again, to return to earth and fight off the invasion.

Each enemy behaves differently especially their attacks and movement speed. There are health packs and ammo crates located all throughout the space ship. The player will learn to carefully use their resources efficiently so that they are not swarmed. The player will also learn how to dodge enemy fire and learn to hide behind obstacles for cover. Every artwork and design has a purpose whether it be for the player's benefit or in favor of the aliens.

The controls for the game are simple W,A,S,D to walk, and aim with the mouse

W being walking up
A being walking left
S being walking down
D being walking right



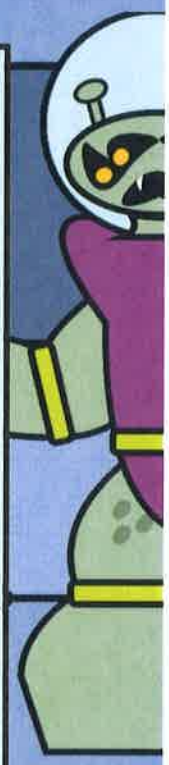


You can use the left click on the mouse to fire or spacebar to whatever best fits your preference. Finally you move the character around using the mouse.

Press E to swap weapons whether to use the laser gun or flamethrower or the laser gun.

Remember that aliens are weak to fire and the robots are weak to lazers
Defeat and conquer the aliens and take back what's ours!

SWOT ANALYSIS



STRENGTHS-

- Entertaining RPG game for all ages.
- Creative story and easy to play.
- Unique genre and character design.

WEAKNESSES-

- First game we've ever made.
- Communication and teamwork are skills we are still developing. but have improved.

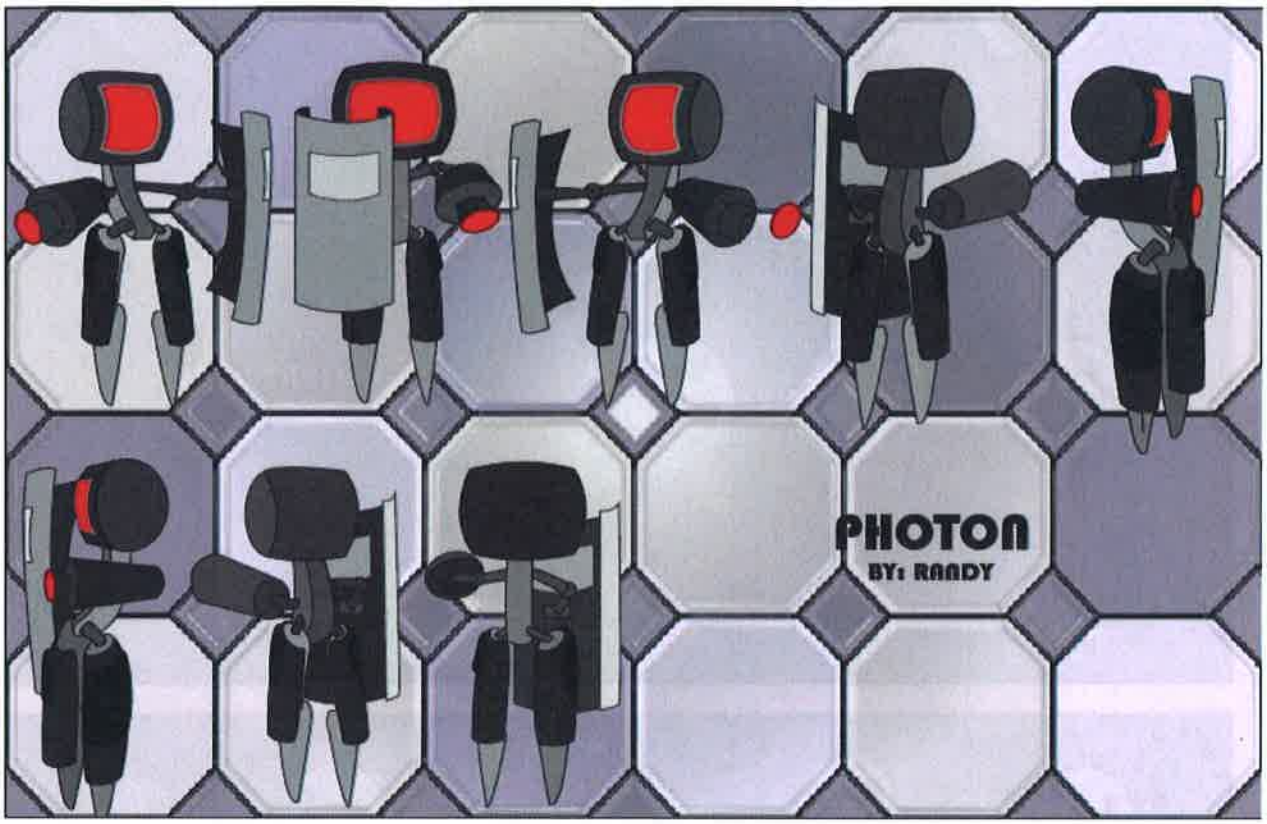
OPPORTUNITIES-

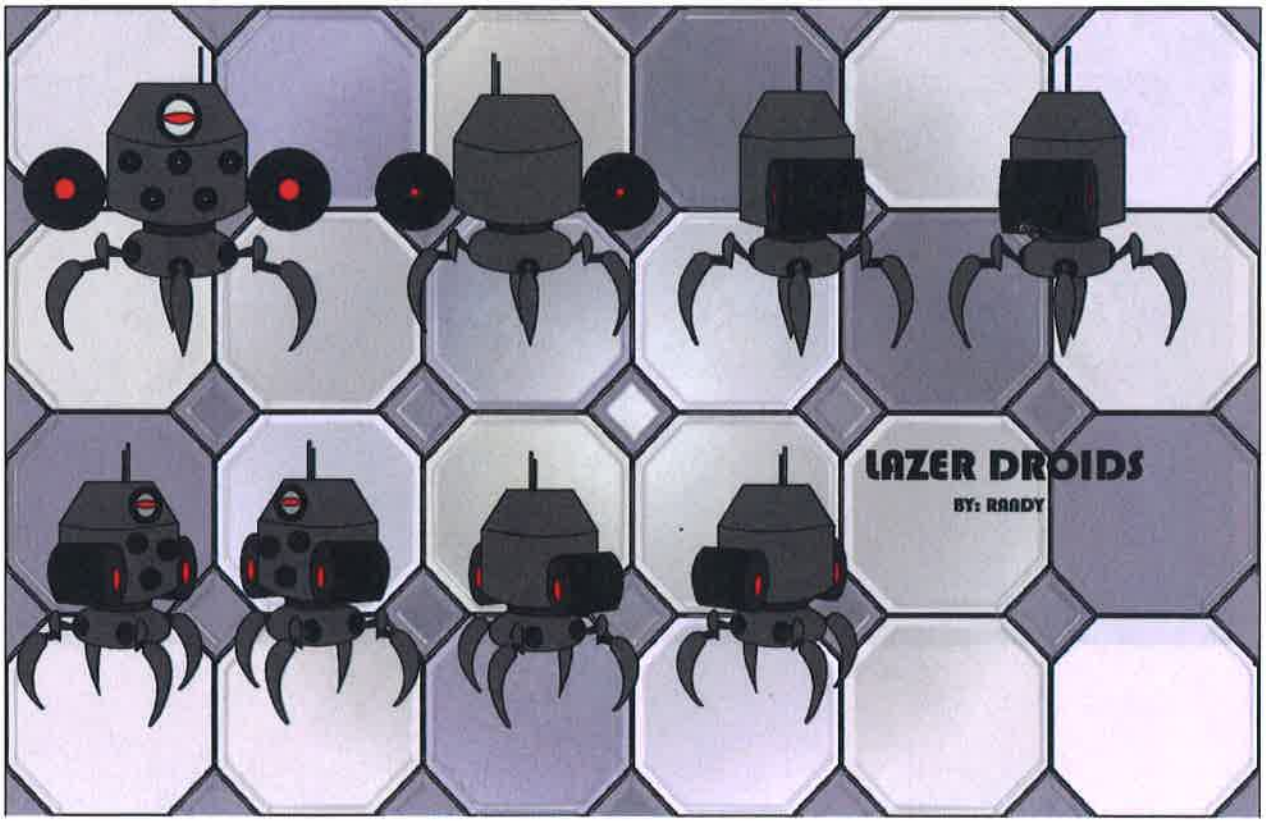
- To learn necessary skills to build a game from beginning to finish with a team and work efficiently.

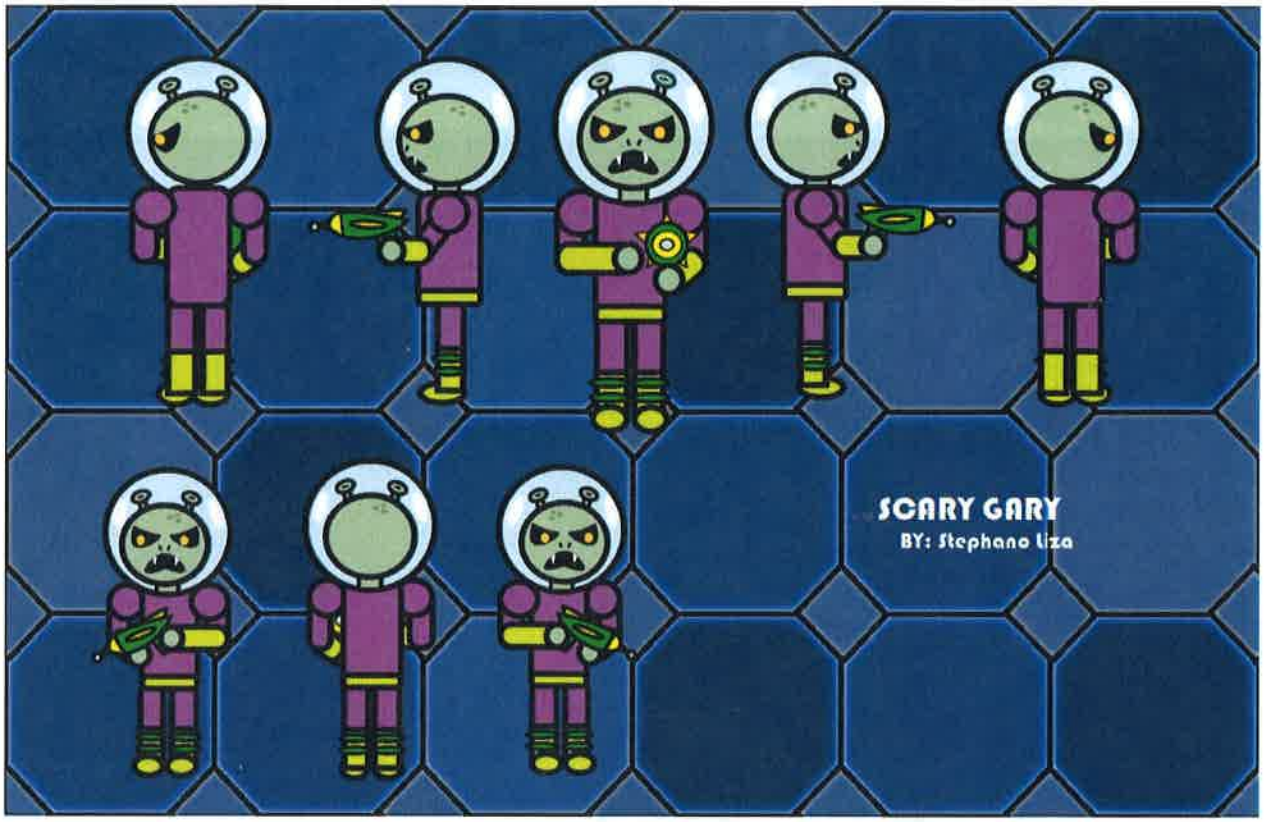
THREATS-

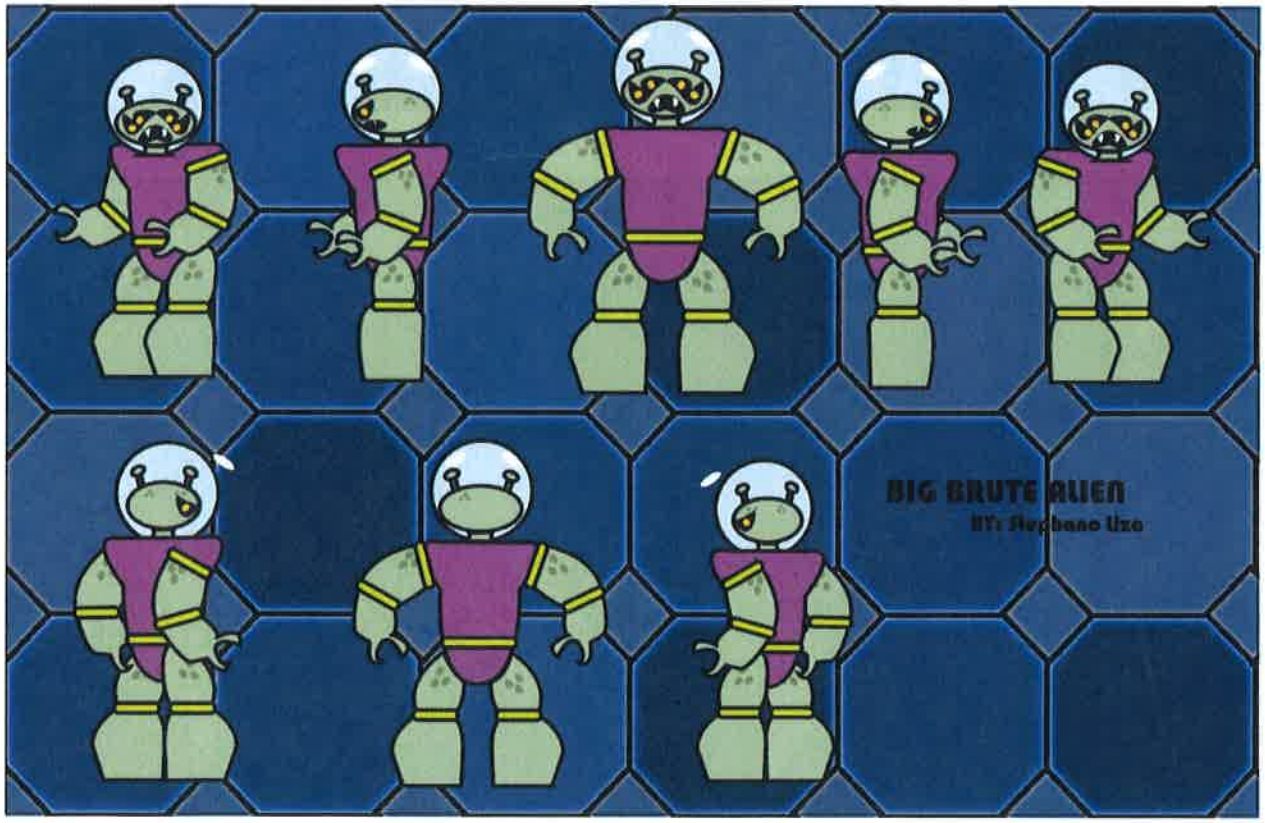
- We learn as we build. Might encounter tough obstacles that can slow down our progress. but only temporarily.

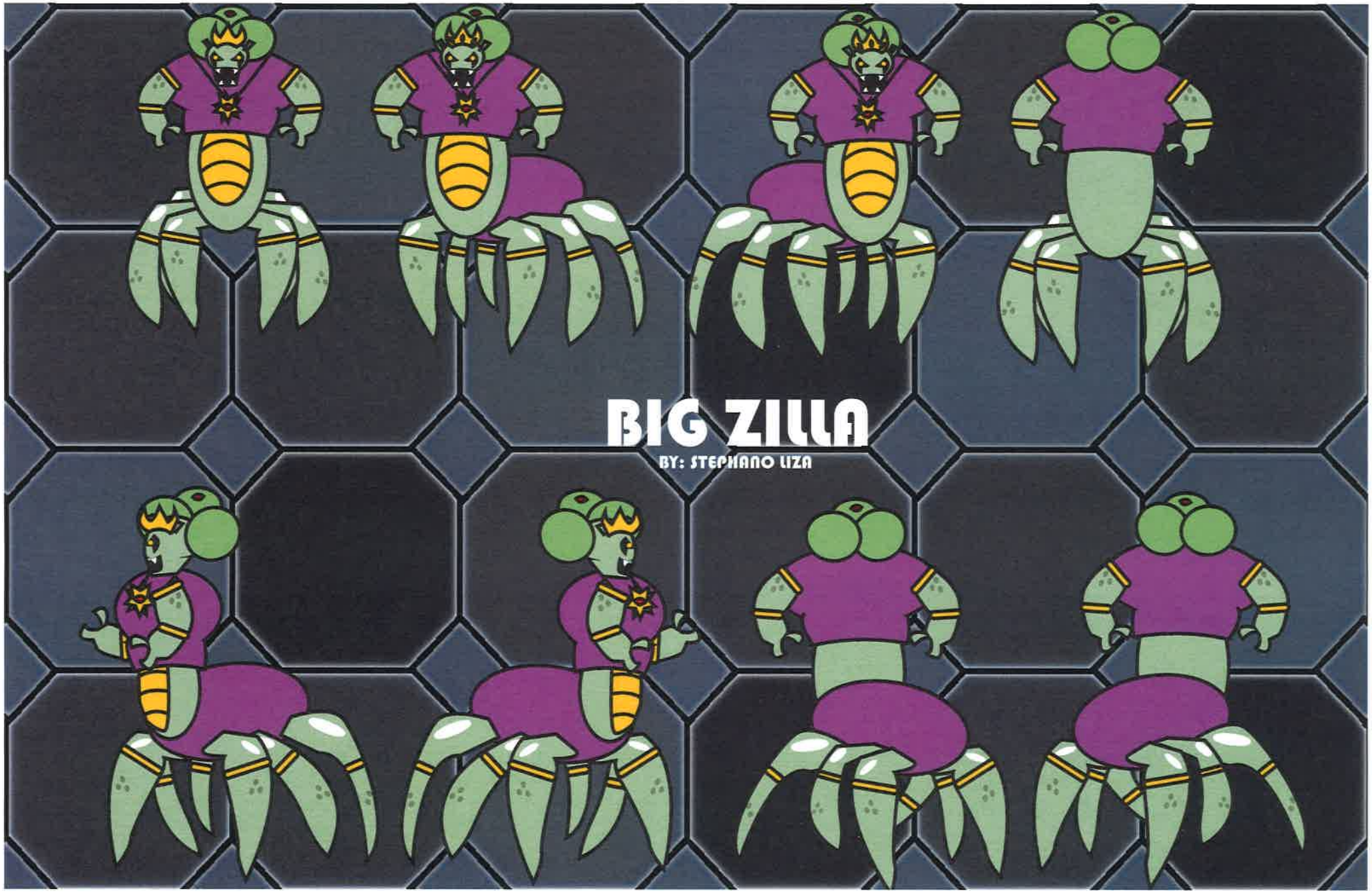
Concept Art ~ Enemies



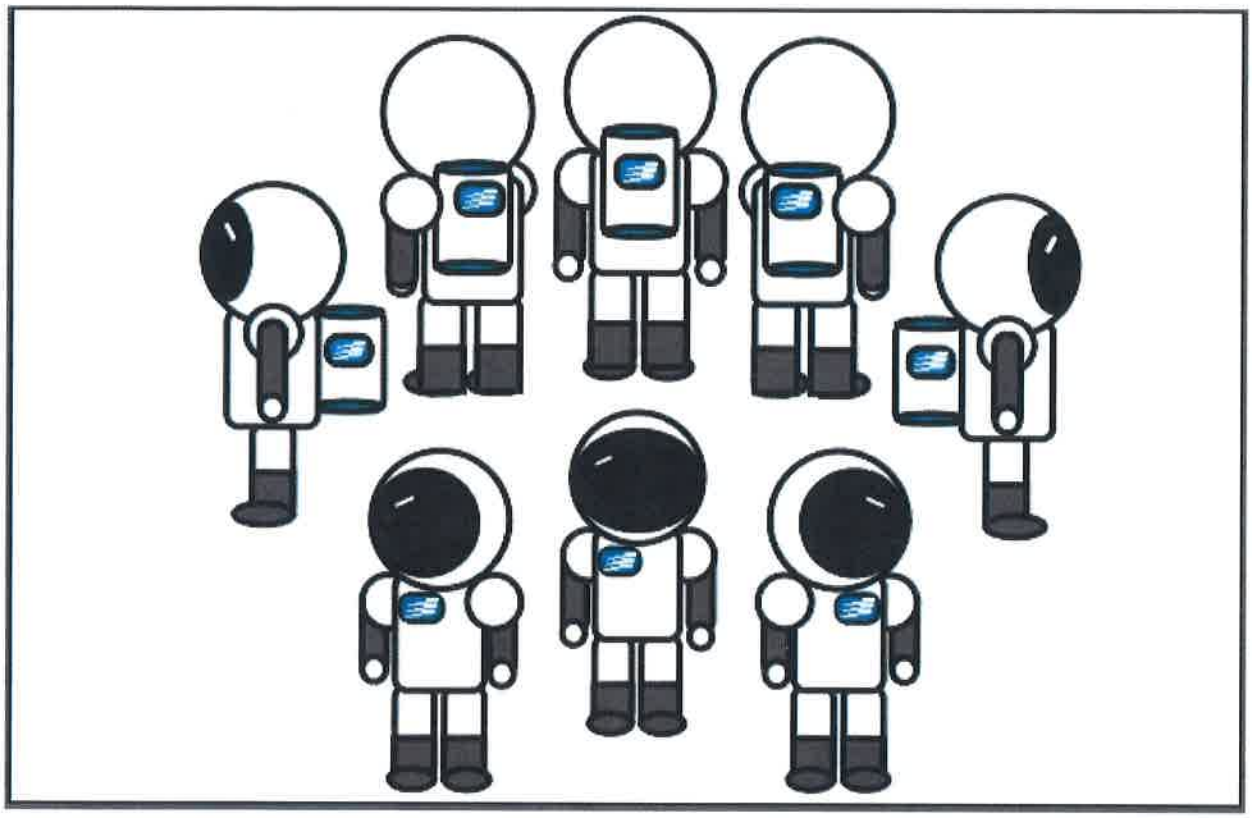



SCARY GARY
BY: Stephano Liza


BIG BRUTE ALIEN
BY: Stephano Liza


BIG ZILLA
BY: STEPHANO LIZA

Concept Art ~ Main character



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

public class stats : MonoBehaviour
{

    public int Health = 100;
    public int PlayerHealth = 100;
    public GameObject[] body;
    public GameObject fire;
    public int bodyspawn = 50;
    public int lazerammo = 10;
    public int flamefuel = 100;
    private int flamecap;
    private int lazercap;
    public bool scatterspawn = false;
    public int firetime = 50;
    public bool alien;
    public bool poisionres = false;
    public bool onfire = false;
    private bool onfirecontdown = false;
    public int x=0;
    private GameObject killthing;
    private bool fireswitch = true;
    //public bool isdead = false;
    // Start is called before the first frame update
    void Start()
    {
        PlayerHealth = Health;
        killthing = GameObject.Find("Killcontroler");
        flamecap = flamefuel;
        lazercap = lazerammo;
    }

    // Update is called once per frame
    void Update()
    {
        if(flamefuel > flamecap)
        {
            flamefuel = flamecap;
```

setting all the GameObjects and variables

setting max ammo and player health

```
        }
        if (lazerammo > lazercap)
        {
            lazerammo = lazercap;
        }
        if (onfire )//&& fire != null)
        {
            GetComponent<Renderer>().material.color = Color.red;
            X++;
         if (fireswitch)                          setting something on fire if it touches the flame on the
                                                        flamethrower, turning it red so we know it works
            {
                if (fire != null)
                {

                    GameObject fre = Instantiate(fire, this.transform.position + new Vector3(0f,0.5f,0f),
Quaternion.Euler(0f, 180f, 0f));
                    fre.GetComponent<bigflame>().pos = this.gameObject;
                }

                fireswitch = false;
            }
        }
        else
        {
            GetComponent<Renderer>().material.color = Color.white;
            fireswitch = true;
        }

        if(x > firetime)
        {
            onfire = false;
            x = 0;                                  setting the amount of time enemies are on fire
        }

        if(PlayerHealth <= 0)
        {
            if(this.tag == "Player") {
                if (SceneManager.GetActiveScene().name == "secondscene")
                {
                    SceneManager.LoadScene("respawnscene");
                }
                else if (SceneManager.GetActiveScene().name == "firstscene")
```

```
            {
        SceneManager.LoadScene("respawnforfirstscene");  setting the death animation/respawn
                                                                                    scene
            }
            else if (SceneManager.GetActiveScene().name == "thirdscene")
            {
                SceneManager.LoadScene("respawnthirdscene");
            }
        }
        Destroy(this.gameObject);
        if(this.tag == "Enemy")
        {

            int bla = killthing.GetComponent<Killcount>().kills += 1;
            PlayerPrefs.SetInt("kill", bla);
        }
        if(body.Length > 0 && body[0] != null)
        {
                    for(int i = 0; i < bodyspawn; i++)    a general spawn for anything we want on any
                                                                                    object we want
            {
                for (int ii = 0; ii < body.Length; ii++)
                {
                    if (scatterspawn)
                    {
                        Instantiate(body[ii], transform.position + new Vector3(i*.5f,0f,ii*.5f),
body[ii].transform.rotation);
                    }
                    else
                    {
                        Instantiate(body[ii], transform.position, body[ii].transform.rotation);


                    }


                }

            }
        }
    }

}
```

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

public class flamethrowerslider : MonoBehaviour
{
    public Slider FlamethrowerSlider;          setting the slider as a public draggable object
    private GameObject player;
    public int ammo = 30;
    public int currentammo = 1000;
    public int startingammo = 1000;
    stats stats;
    // Start is called before the first frame update
    void Start()
    {

        currentammo = ammo;                    setting what the slider is at when you start



    }

    // Update is called once per frame
    void Update()
    {
        FlamethrowerSlider.value = GameObject.Find("Player").GetComponent<stats>().flamefuel;
    }
    void outofammo()
    {
        if (currentammo < 1)                   setting it so if you are out of ammo you cannot shoot
        {
            stats.enabled = false;
        }
    }
}
```