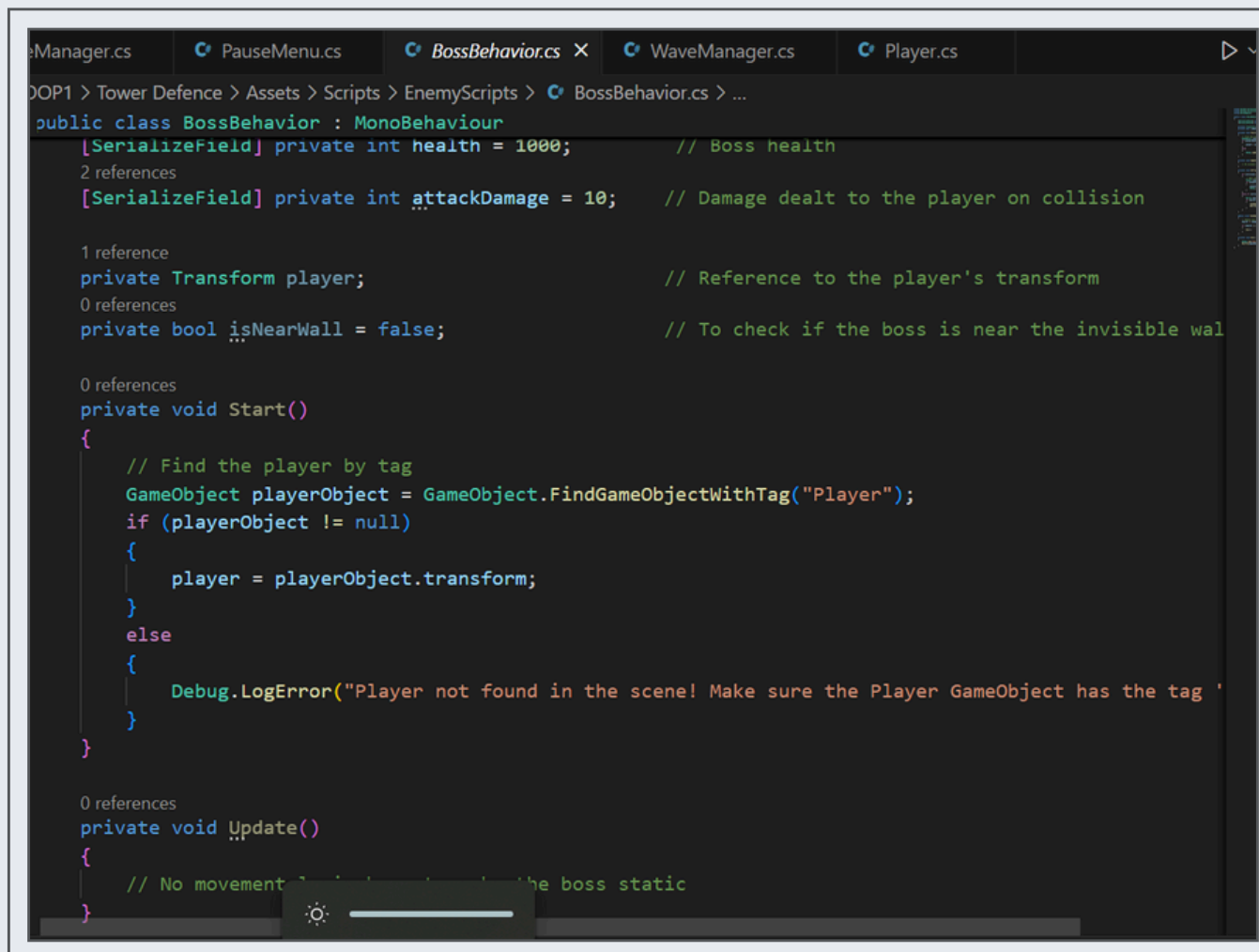


# Presentasi **PROYEK AKHIR OOP**

**KELOMPOK 1**



# Pendahuluan



```
Manager.cs | PauseMenu.cs | BossBehavior.cs | WaveManager.cs | Player.cs
DOP1 > Tower Defence > Assets > Scripts > EnemyScripts > BossBehavior.cs > ...
public class BossBehavior : MonoBehaviour
{
    [SerializeField] private int health = 1000; // Boss health
    [SerializeField] private int attackDamage = 10; // Damage dealt to the player on collision

    1 reference
    private Transform player; // Reference to the player's transform
    0 references
    private bool isNearWall = false; // To check if the boss is near the invisible wall

    0 references
    private void Start()
    {
        // Find the player by tag
        GameObject playerObject = GameObject.FindGameObjectWithTag("Player");
        if (playerObject != null)
        {
            player = playerObject.transform;
        }
        else
        {
            Debug.LogError("Player not found in the scene! Make sure the Player GameObject has the tag 'Player'");
        }
    }

    0 references
    private void Update()
    {
        // No movement for the boss static
    }
}
```

Tower Defence adalah game shooter di mana pemain mengendalikan tower bersenjata untuk mengalahkan musuh. Pemain dapat memilih berbagai jenis senjata untuk menambah variasi permainan. Ada 4 jenis musuh yang muncul secara acak dan mendekati tower, mengurangi HP pemain jika mencapai tower. Selain itu, ada bos yang memunculkan musuh tanpa henti hingga bos tersebut dikalahkan.

# Deskripsi Fitur

# Sistem Karakter Player

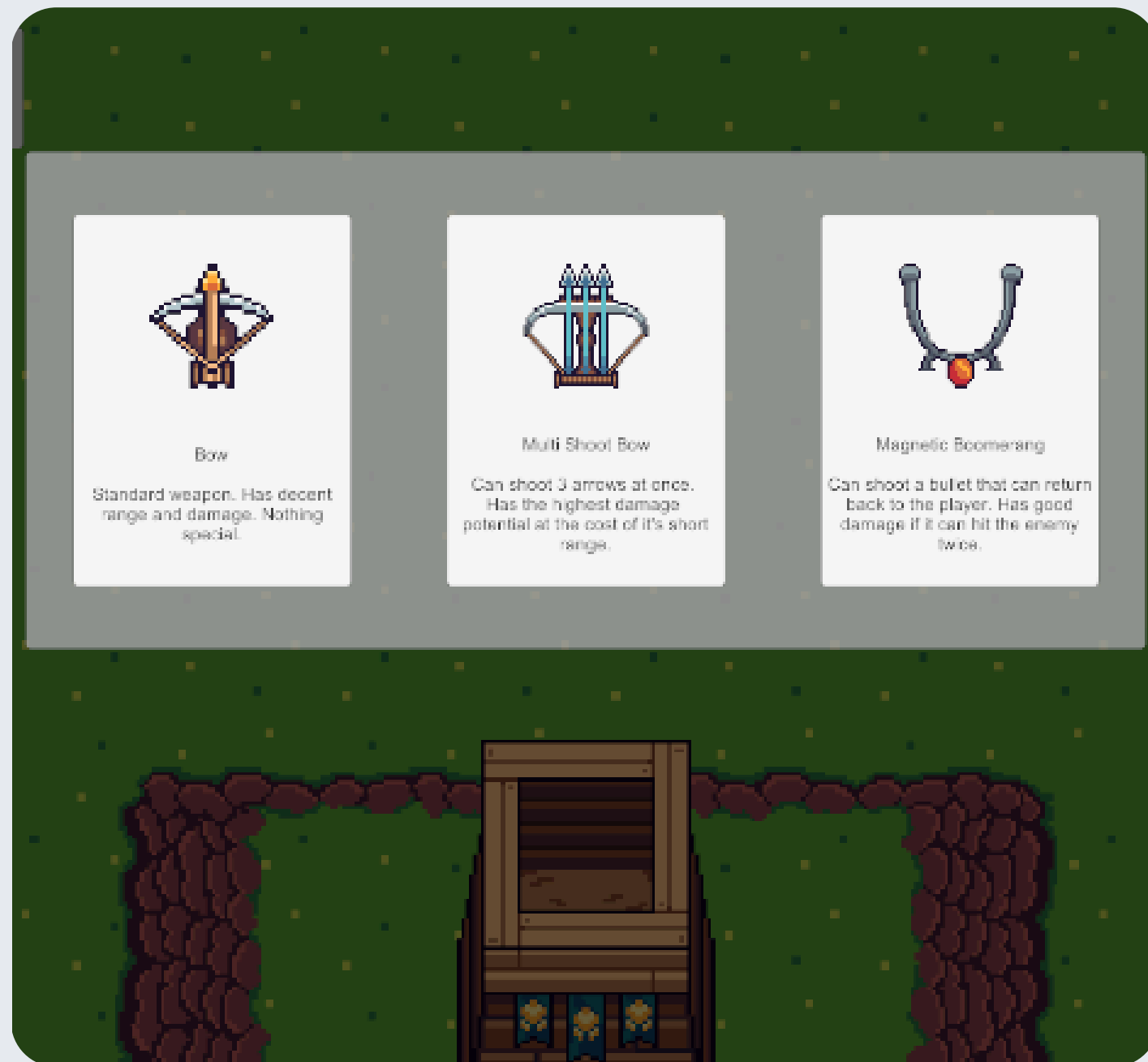


Player dalam game terdiri dari dua child object: Tower sebagai tampilan dan Weapon sebagai senjata yang bervariasi sesuai pilihan di awal. Player tidak dapat bergerak tetapi bisa menembak peluru ke arah musuh. Script Player menggunakan singleton pattern untuk memastikan hanya ada satu player dalam game.

Peluru menggunakan metode object pool dengan daur ulang 9 peluru untuk efisiensi. Objek yang membutuhkan Health dan Defense dapat menggunakan komponen terkait.

Saat player mati, scene akan diulang, dan wave counter kembali ke 0 untuk memulai permainan baru.

# Pemilihan Weapon



Pemain dalam game diberikan tiga pilihan senjata yang dapat dipilih di awal permainan, yaitu Bow, Multi Shoot Bow, dan Magnetic Boomerang. Bow merupakan senjata standar dengan jangkauan dan damage yang seimbang tanpa kemampuan khusus. Multi Shoot Bow memiliki kemampuan menembakkan tiga anak panah sekaligus, memberikan damage tinggi dengan jangkauan pendek. Sementara itu, Magnetic Boomerang adalah senjata unik yang dapat menembakkan peluru yang kembali ke pemain, memberikan damage tambahan jika peluru mengenai musuh dua kali.

# Sistem Kontrol



Kontrol pada game ini berfokus pada mengendalikan arah tembakan dari senjata milik player. Player akan menggunakan kursor mouse yang akan berfungsi sebagai target dari peluru milik player. Jika tombol kiri mouse ditekan, maka senjata player akan menembakkan peluru ke posisi kursor saat menembak.

Pertama-tama dalam script `MouseControl` digunakan function `SetCursor()` untuk menetapkan texture, mode, dan koordinat titik tengah kursor dan dijalankan ketika start sehingga hanya sekali saja dijalankannya.

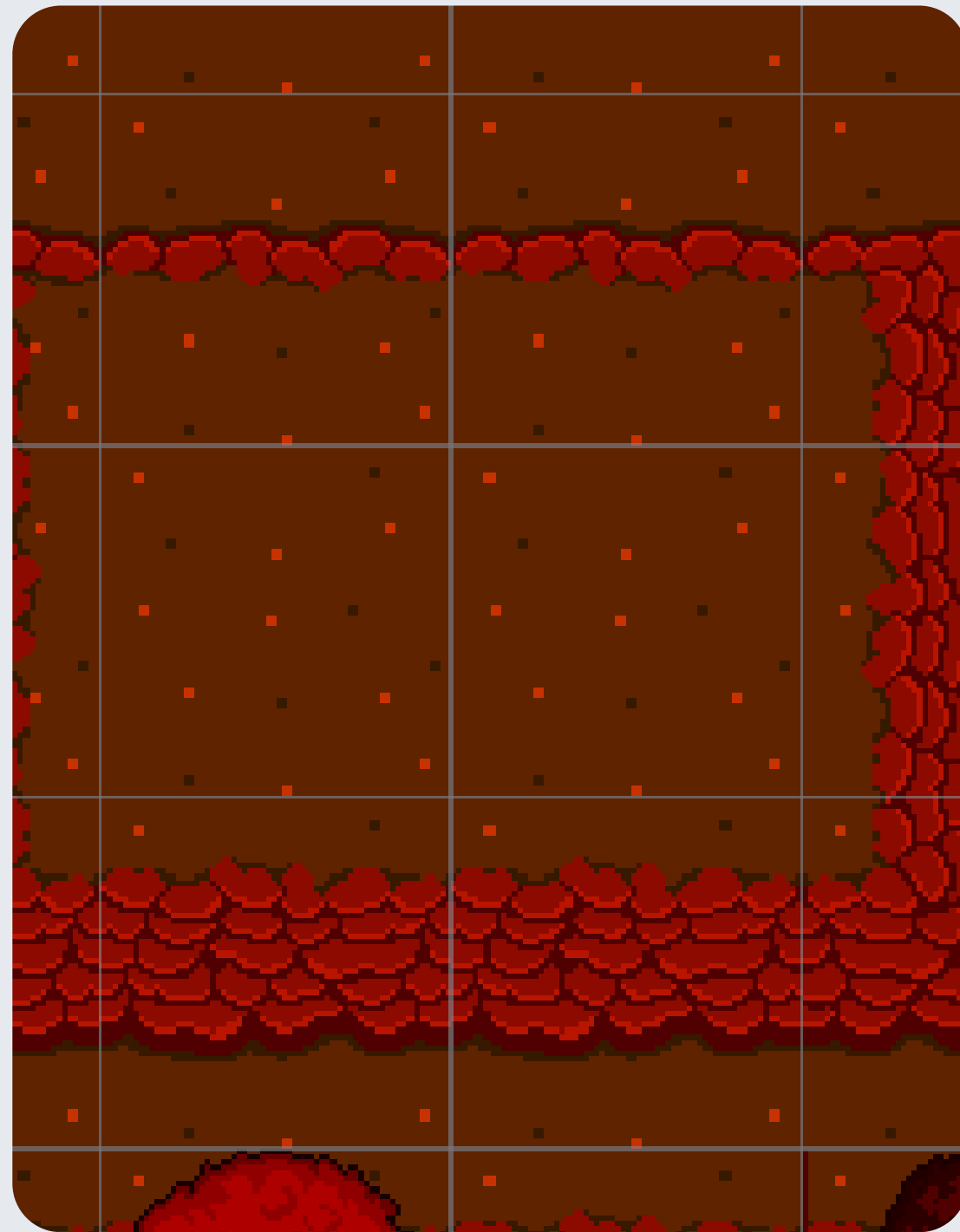
# Sistem Enemy



Enemy dalam game dikendalikan oleh Enemy Manager untuk mengatur spawn dan jumlah musuh di setiap wave, serta Enemy Behavior untuk mengelola collision dengan peluru atau objek lain. Saat terkena peluru, health musuh berkurang, dan jika mencapai 0, musuh mati dan objeknya dihapus tanpa reuse melalui object pool.

Musuh spawn secara acak dalam area kamera, dengan jumlah musuh dibatasi oleh array list di Enemy Manager untuk mencegah spawn tak terbatas. Scaling difficulty diterapkan dengan menambah dua musuh di setiap wave baru, yang dikelola oleh Wave Manager.

# Sistem Boss



Dalam Tower Defence, boss battle merupakan scene melawan boss akhir dari project ini, boss akan memunculkan musuh-musuh dengan interval tertentu tanpa henti hingga boss tersebut sendiri mati, boss tersebut sama seperti tower yaitu statis tidak bergerak sama sekali.



# Kesimpulan

Project Tower Defence merupakan implementasi dari prinsip OOP yang telah dipelajari selama semester ini. Dengan mengimplementasikan design pattern serta berbagai practice pemrograman yang baik, program yang dihasilkan diharapkan dapat scalable, secure, low coupling serta memiliki fokus yang jelas untuk setiap bagiannya. Untuk kedepannya, kekurangan seperti bug ataupun fitur yang belum lengkap akan menjadi bahan evaluasi kami.

# Terima Kasih

Murad Naser | Universitas Fauget | Ekonomi | 2025