

## **LAPORAN UAS**

### **PEMROGRAMAN BERORIENTASI OBJEK**

**Dosen Pengampu : Sayekti Harits Suryawan, S.kom., M.Kom.**



**INF-2143 / 2153**

### **KELOMPOK 13**

**Nama Anggota :**

- 1. Muhammad Aditya Rahman : 2211102441217**
- 2. Maliq Yazid Imani : 2211102441082**
- 3. Renaldi Yoga Rendy Menono : 2211102441147**

**S1 TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS MUHAMMADIYAH KALIMANTAN TIMUR**

**2023**

## **Berikut merupakan elemen-elemen yang mencakup pada Game Litte OctoLife :**

➤ **Berikut ini adalah tampilan utama dari game Little Octo Life:**

- Didalamnya terdapat objek logo dan beberapa button seperti play, about, dan tutorial.
- Masing-masing button memiliki halaman yang berbeda jika di klik.



➤ **Berikut adalah tampilan about jika button ABOUT diklik**

didalam halama ABOUT terdapat cerita singkat tentang game yang telah kami buat dan nama anggota kelompok 13. Pada halaman ini terdapat objek button BACK yang berfungsi untuk Kembali kehalaman utama.



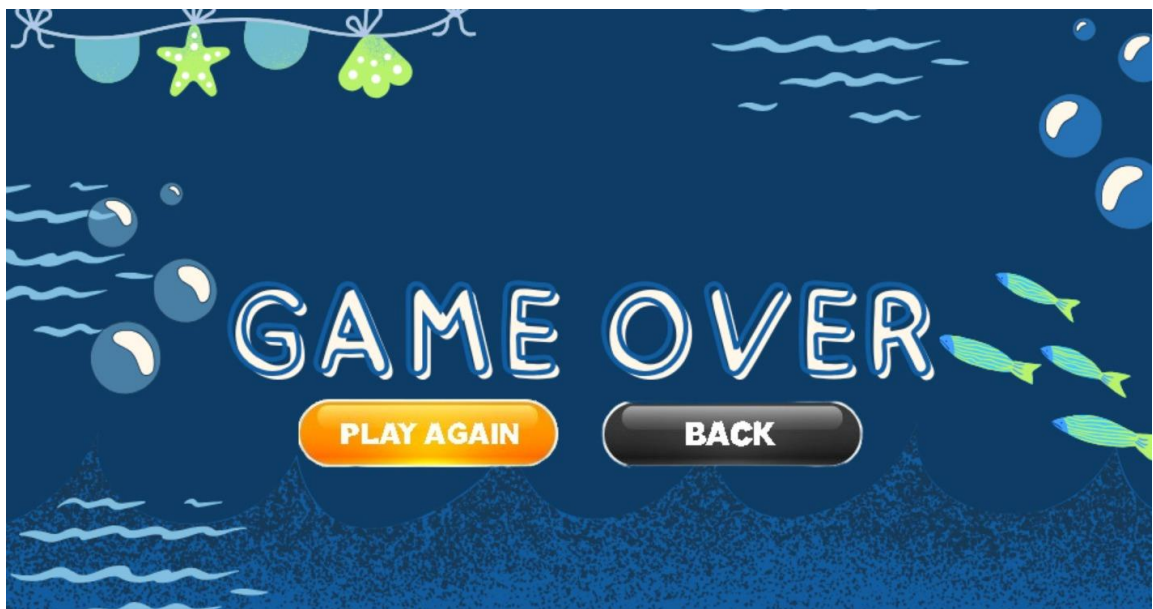
➤ **Berikut ini adalah halaman tutorial**

Didalamnya terdapat panduan untuk mengontrol atau menggerakkan player serta terdapat keterangan untuk menghindari musuh seperti yang di gambarkan dan keterangan untuk memakan/ mendapatkan score seperti yang digambarkan untuk melangkah ke level selanjutnya. Sama seperti halaman ABOUT, halaman ini terdapat objek button BACK untuk Kembali ke halaman utama.



➤ **Berikut adalah halaman Game Over**

Jika health player habis dan diarahkan ke halaman Game Over, didalam halaman ini terdapat button Play Again yang berfungsi untuk bermain lagi dari level awal dan button BACK untuk Kembali ke halaman utama.





➤ **Berikut adalah tampilan jika anda berhasil memenangkan level**

Akan mendapatkan score yang ditentukan, jika anda menekan tombol enter maka akan langsung diarahkan ke halaman level selanjutnya.



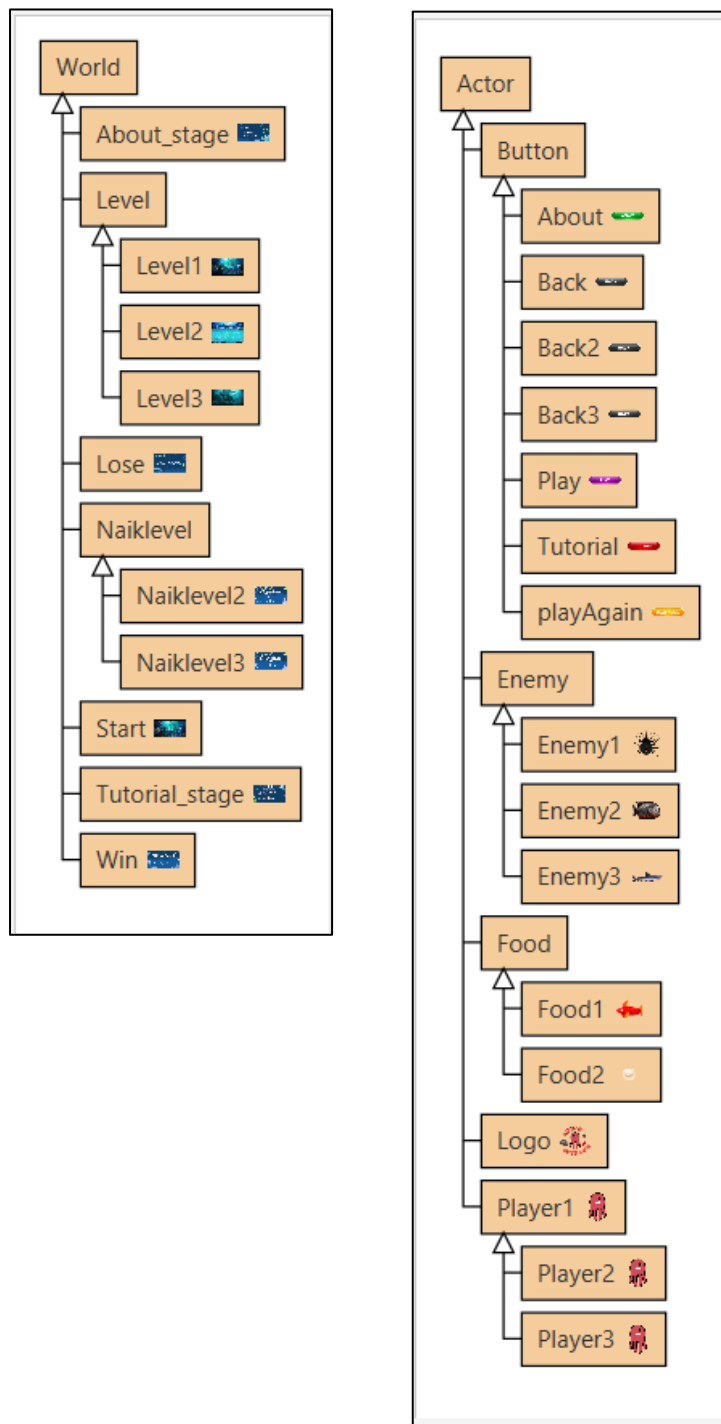
➤ **Berikut tampilan jika player telah memenangkan semua level**

Apabila player telah menyelesaikan semua tantangan, maka akan mengarah ke tampilan Game End seperti pada gambar dibawah



➤ **Kelas dan Objek :**

Berikut ini beberapa kelas dan objek pada game yang telah kami buat :



➤ **Pewarisan (Inheritance) :**

- Pada kelas Player1() dapat melakukan pewarisan untuk subkelasnya player2() dan player3()
- Setiap subkelas mewarisi sifat-sifat dari kelas player1() seperti kontrol dan animate

➤ **Polimorfisme :**

- Pada Kelas Food Terdapat dua subkelas yaitu Food1() dan Food2()
- Pada subkelas Food1() akan muncul dari arah kanan dan bergerak ke arah kiri, jika dia mencapai koordinat  $X = 0$  maka dia akan menghilang.
- Pada subkelas Food2() akan muncul dari arah Atas dan bergerak ke arah Bawah, jika dia mencapai koordinat  $Y = 0$  maka dia akan menghilang.
- Pada Kelas Enemy() Terdapat tiga subkelas yaitu Enemy1(), Enemy2() dan Enemy3().
- Pada subkelas Enemy1() akan muncul dari arah Atas dan bergerak ke arah Bawah, jika dia mencapai koordinat  $Y = 0$  maka dia akan menghilang.
- Pada subkelas Enemy2() akan muncul dari arah Kiri dan bergerak ke arah Kanan, jika dia mencapai koordinat  $X = 949$  maka dia akan menghilang.
- Pada subkelas Enemy2() akan muncul dari arah Kiri dan bergerak ke arah Kanan, jika dia mencapai koordinat  $X = 949$  maka dia akan menghilang.
- Pada subkelas Enemy3() akan muncul dari arah Kanan dan bergerak ke arah Kiri, jika dia mencapai koordinat  $X = 0$  maka dia akan menghilang.

➤ **Enkapsulasi :**

- Pada kelas Player1(), Player2(), dan Player3() memiliki Atribut yang menggunakan enkapsulasi dimana “jumlahFoodDimakan++” dan “enemyDimakan—” merupakan atribut yang tidak dapat diakses secara langsung dari luar kelas. Atribut ini diatur dan diakses melalui metode-metode dalam kelas tersebut.
- Pada Kelas Player1(), Player2(), dan Player3() terdapat Metode-metode yang menggunakan enkapsulasi dimana, Metode-metode seperti eatFood(), eatEnemy(), dan Checked() memberikan akses terkontrol ke atribut dan

fungsionalitas kelas. Penggunaan metode `getOneIntersectingObject()` untuk mendapatkan objek yang berinteraksi dengan `Player2` juga merupakan contoh enkapsulasi, karena detail implementasinya disembunyikan.

➤ **Interaksi Antar Objek :**

- Objek `Player1()`, `Player2()`, dan `Player3()` dapat berinteraksi dengan Object `Food1()` sampai `Food2()`, jika mereka saling bertabrakan maka akan mendapatkan 1 Score dan dapat mengeluarkan sound (Efek suara).
- Objek `Player1()`, `Player2()`, dan `Player3()` dapat berinteraksi dengan Object `Enemy1()`, `Enemy2()`, `Enemy3()`, jika mereka saling bertabrakan maka akan mengurangi 1 health dan dapat mengeluarkan sound (Efek suara).

➤ **Overriding dan Overloading :**

- **Overriding**

Metode `act()` pada subkelas `Player2()` dan `Player3()` meng-override metode `act()` dari kelas induk `Player1()`. Metode-metode lain seperti `eatFood()`, `eatEnemy()` dan `Checkec()` juga meng-override metode-metode dengan nama yang sama dari kelas `Player1()`.