

# Biomes Run

## Sommario

1	Introduzione.....	4
1.1	Informazioni sul progetto.....	4
1.2	Abstract.....	4
1.3	Scopo.....	4
2	Analisi.....	5
2.1	Analisi del dominio.....	5
2.2	Analisi e specifica dei requisiti.....	5
2.3	Use case.....	9
2.4	Pianificazione.....	10
2.5	Analisi dei mezzi.....	11
2.5.1	Software.....	11
2.5.2	Hardware.....	11
3	Progettazione.....	12
3.1	Design dell'architettura del sistema.....	12
3.2	Design dei dati e database.....	12
3.3	Design delle interfacce.....	13
3.4	Design procedurale.....	19
4	Implementazione.....	20
4.1	Costruzione labirinto.....	20
4.2	Costruzione 3 livelli in Unity.....	22
4.3	Decorazione Labirinto.....	23
4.3.1	Modelli 3D.....	23
4.3.2	Decorazione labirinto.....	24
4.3.3	Albero cartelle.....	26
4.3.4	Script Rotatore.....	26
4.4	Prima persona.....	27
4.5	Vite personaggio.....	29
4.6	Programmazione mostro.....	31
4.7	Programmazione schermate.....	32
4.7.1	Schermata 0 (Benvenuto).....	32
4.7.2	Schermata 1 (Vite) Schermata di gioco.....	33
4.7.3	Schermata 2 (Livello1).....	33
4.7.4	Schermata 3 (Secondo livello).....	34
4.7.5	Schermata 4 (Livello3).....	34
4.7.6	Schermata 5 (Pausa).....	35
4.7.7	Schermata 6 (Chiave falsa).....	35
4.7.8	Schermata 7 (Vittoria).....	36
4.7.9	Schermata 8 (Vite finite).....	36
5	Test.....	37
5.1	Protocollo di test.....	37
5.2	Risultati test.....	40
5.3	Mancanze/limitazioni conosciute.....	40
6	Consuntivo.....	40
7	Conclusioni.....	40
7.1	Sviluppi futuri.....	40
7.2	Considerazioni personali.....	40
8	Glossario.....	40
9	Bibliografia.....	40
9.1	Bibliografia per articoli di riviste.....	40
9.2	Bibliografia per libri.....	40
9.3	Sitografia.....	40
10	Allegati.....	40

## Indice delle figure

Figura 1 - Use Case.....	9
Figura 2 - Gantt Iniziale.....	10
Figura 3 - Tabella classifica.....	12
Figura 4 - Schermata iniziale.....	13
Figura 5 - Schermata di pausa .....	14
Figura 6 - Schermata di vittoria .....	15
Figura 7 - Livello uno .....	16
Figura 8 - Livello due .....	16
Figura 9 - Livello tre .....	17
Figura 10 - Schermata perdita chiave falsa .....	18
Figura 11 - Schermata perdita vite .....	19
Figura 12 - Empty Object .....	20
Figura 13 - Costruzione muri.....	20
Figura 14 – Inizio costruzione labirinto .....	21
Figura 15 - Layout livello 3 .....	21
Figura 16 - Layout livello 2 .....	21
Figura 17 – Layout livello 1 .....	21
Figura 18 - Gerarchia modelli .....	23
Figura 19 - Modelli 3D.....	23
Figura 20 - Decorazione livello1 .....	24
Figura 21 - Decorazione livello 2 .....	24
Figura 22 - Decorazione livello3.....	25
Figura 23 - Ordine oggetti .....	26
Figura 24 - Script rotatore oggetti .....	26
Figura 25 - Chiave con script rotazione .....	27
Figura 26 - Pavimento Grezzo.....	27
Figura 27 - Personaggio Grezzo .....	28
Figura 28 - Inserimento immagine vite.....	29
Figura 29 - Script vite.....	30
Figura 30 - Inserimento immagini nell'array .....	30
Figura 31 - Bake del mostro .....	31
Figura 32 - tag del mostro .....	31
Figura 33 - script mostro .....	31
Figura 34 - Ordine schene.....	32
Figura 35 - Schermata 0 (Benvenuto).....	32
Figura 36 - Schermata 1 (Vite) Schermata di gioco .....	33
Figura 37 - Schermata 2 (Livello 1) .....	33
Figura 38 - Schermata 3 (Livello 2) .....	34
Figura 39 - Schermata 4 (Livello 3) .....	34
Figura 40 - Schermata 5 (Pausa) .....	35
Figura 41 - Schermata 6 (Chiave falsa).....	35
Figura 42 - Schermata 7 (Vittoria) .....	36
Figura 43 - Schermata 8 (Vite finite).....	36

## **1 Introduzione**

---

### **1.1 Informazioni sul progetto**

For this project, we decided to make a game programmed with Unity.

This project was born because we like programming video games and we wanted to program one with Unity.

At the moment there are already games similar to the one we would like to create, but we would like to create a completely different version from those already present online.

in short: this work is mainly focused on entertaining users thanks to our passion for programming video games.

### **1.2 Abstract**

For this project, basic knowledge of Unity will be needed.

Following the requirements set by us, for this project it was essential to organize ourselves to divide the various parts of the project in order to be able to restrict the time and stay with the deliveries.

Various objects will be implemented in the game, each with a specific task, which will lead the player to think while still having fun.

The game offers various levels, each with different characteristics that will not make the player bored during his gaming experience.

### **Results**

This game is made from scratch with a graphics engine (Unity). Not having followed a real module related to it but a simple two-day course, it will be a new thing for us too.

### **1.3 Scopo**

Lo scopo del progetto è principalmente riuscire a far divertire il maggior numero di persone con il nostro videogioco perché ci piace far divertire la gente, inoltre abbiamo deciso questo progetto per poter aumentare le nostre conoscenze con il linguaggio C# usando il motore grafico Unity.

## 2 Analisi

### 2.1 Analisi del dominio

Il gioco verrà utilizzato principalmente dai ragazzi, essendo un gioco progettato appositamente per loro, perché il nostro gioco ha come scopo di uscire da un labirinto nel minor tempo possibile. Inoltre consente ai ragazzi di liberare la mente e svagarsi restando comunque attivi mentalmente attraverso un gioco tranquillo e di ragionamento.

### 2.2 Analisi e specifica dei requisiti

<b>ID: REQ-01</b>	
<b>Nome</b>	Costruzione labirinto
<b>Priorità</b>	1
<b>Versione</b>	1.0
<b>Note</b>	Costruire la base del labirinto
<b>Sotto requisiti</b>	
<b>001</b>	Analisi e progettazione completate

<b>ID: REQ-02</b>	
<b>Nome</b>	Prima persona
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	La visuale dell'utente sarà in prima persona
<b>Sotto requisiti</b>	
<b>001</b>	Il personaggio deve essere completato

<b>ID: REQ-03</b>	
<b>Nome</b>	Movimento personaggio
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	L'utente deve avere la possibilità di spostarsi all'interno del labirinto
<b>Sotto requisiti</b>	
<b>001</b>	Il layout del gioco deve essere completato in maniera grezza (Livelli)
<b>002</b>	Il personaggio deve essere completato

<b>ID: REQ-04</b>	
<b>Nome</b>	Decorazione labirinto
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	Decorare il labirinto grezzo, far sì che ci siano i 3 biomi dei 3 livelli ed aggiungere delle canzoncine spaventose e posizionare tutti gli oggetti 3d
<b>Sotto requisiti</b>	
<b>001</b>	Labirinto grezzo completo

<b>ID: REQ-05</b>	
<b>Nome</b>	Programmazione mostri
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	Nei livelli ci saranno dei mostri che cercheranno di fermare il giocatore
<b>Sotto requisiti</b>	
<b>001</b>	Il layout livelli deve essere completato
<b>002</b>	Il personaggio deve essere completato

<b>ID: REQ-06</b>	
<b>Nome</b>	Vite personaggio
<b>Priorità</b>	2
<b>Versione</b>	1.0
<b>Note</b>	L'utente dispone di tre vite durante il gioco
<b>Sotto requisiti</b>	
<b>001</b>	I mostri devono essere completati
<b>002</b>	Il personaggio deve essere completato

<b>ID: REQ-07</b>	
<b>Nome</b>	<b>Funzionamento schermata iniziale</b>
<b>Priorità</b>	3
<b>Versione</b>	1.0
<b>Note</b>	<b>Rendere funzionante la schermata iniziale</b>
<b>Sotto requisiti</b>	
<b>001</b>	<b>Design schermata iniziale completato</b>
<b>002</b>	<b>Funzionamento gioco completo</b>

<b>ID: REQ-08</b>	
<b>Nome</b>	Funzionamento schermata impostazioni
<b>Priorità</b>	3
<b>Versione</b>	1.0
<b>Note</b>	Rendere funzionante la schermata iniziale
<b>Sotto requisiti</b>	
<b>001</b>	Design schermata impostazioni completato
<b>002</b>	Funzionamento gioco completo

<b>ID: REQ-09</b>	
<b>Nome</b>	Database
<b>Priorità</b>	3
<b>Versione</b>	1.0
<b>Note</b>	Creare un database il quale contiene i migliori tempi dei giocatori ed i loro nickname
<b>Sotto requisiti</b>	
<b>001</b>	Design schermata vincita e perdita
<b>002</b>	Funzionamento gioco completo
<b>003</b>	Funzionamento schermata impostazioni e iniziale

<b>ID: REQ-10</b>	
<b>Nome</b>	Funzionamento schermata vincita (classifica)
<b>Priorità</b>	3
<b>Versione</b>	1.0
<b>Note</b>	Quando il giocatore vince la partita compare la schermata di vincita la quale contiene una classifica dei migliori tempi (DB)
<b>Sotto requisiti</b>	
<b>001</b>	Database funzionante
<b>002</b>	Gioco completo



## 2.3 Use case

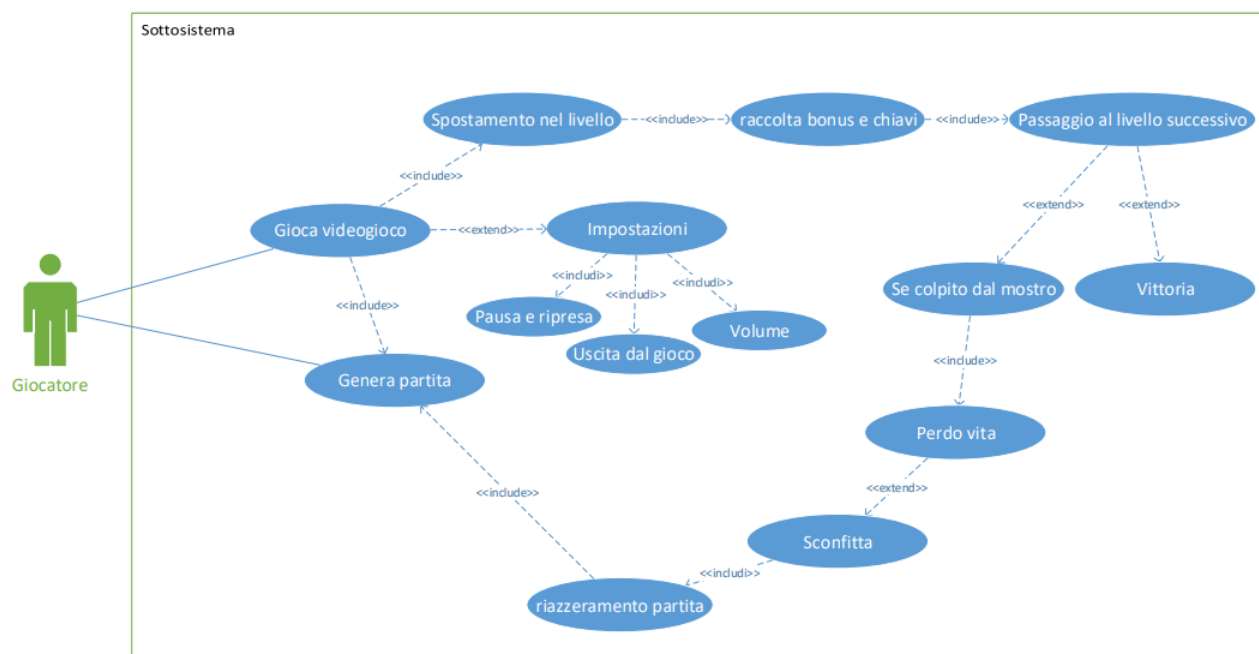


Figura 1 - Use Case

All'interno dell'applicativo c'è solo un utente che è il giocatore:

- Esso può giocare al videogioco, ha come scopo quello di scappare dal labirinto nel minor tempo possibile prendendo tutti gli oggetti e le chiavi.
- Giocando al videogioco esso genera una nuova partita.
- Esso può utilizzare la schermata di impostazioni contenenti:
  - Regolazione volume.
  - Pausa e ripresa durante il gioco.
  - Tasto per uscire dal gioco

## 2.4 Pianificazione

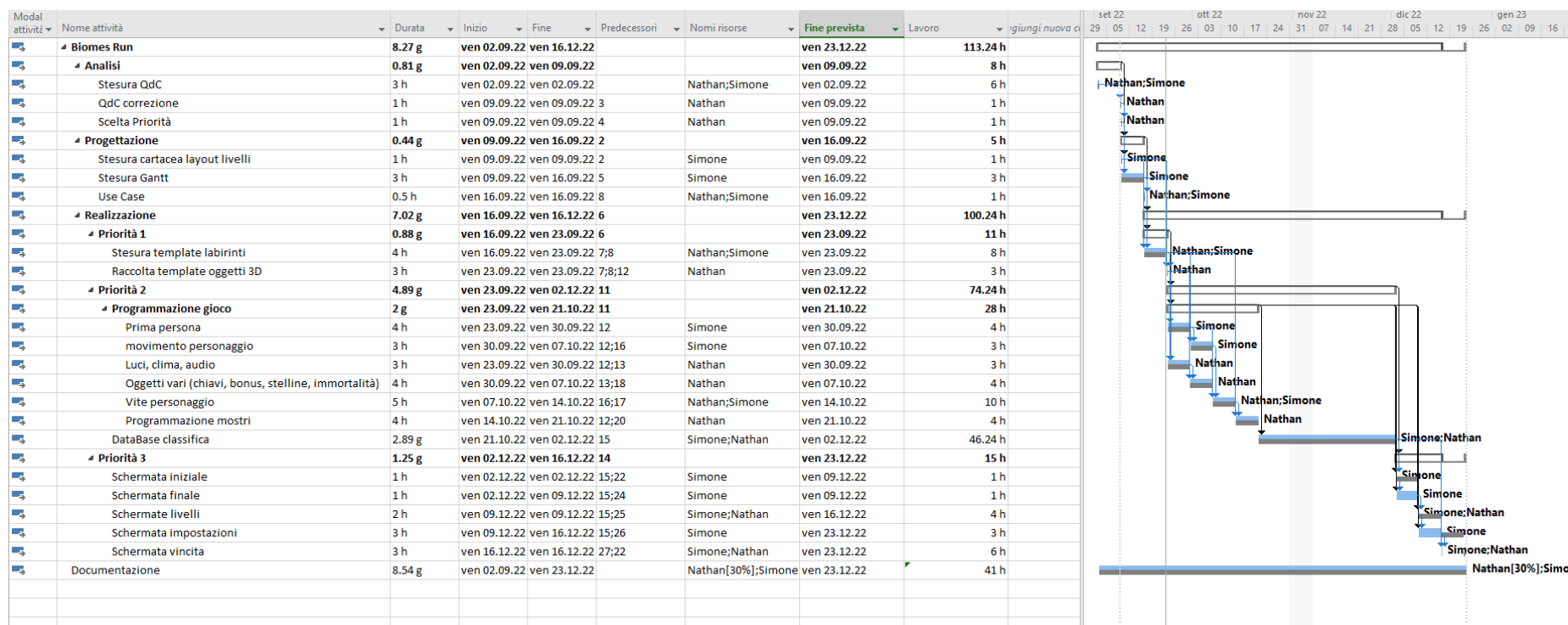


Figura 2 - Gantt Iniziale

La durata del nostro progetto, come si può vedere nell'immagine qui sopra, dura una settimana in meno rispetto alla data di consegna. Questo è perché ho deciso di inserire un margine di errore nel caso dovessi avere problemi oppure dovessi essere indietro rispetto al Gantt. Nel mio Gantt sono riuscito a ricavare varie attività con 29 righe da svolgere, inoltre ho ricavato 3 macrocategorie con 3 sottocategorie che sono:

1. **Analisi:** All'interno di questa macrocategoria ci sono le parti contenenti la pianificazione teorica del progetto con il Quaderno dei compiti e la sua relativa correzione e la scelta delle varie priorità delle parti del progetto.
2. **Progettazione:** All'interno di questa macrocategoria c'è l'inizio della vera e propria progettazione del progetto utilizzando un Diagramma di Gantt ed una Use Case, inoltre abbiamo iniziato a stendere i layout in maniera cartacea dei vari livelli.
3. **Realizzazione:** Questa parte riguarda la vera e propria realizzazione del progetto, l'ho suddiviso in tre categorie:
  - a. **Priorità 1:** In questa categoria c'è l'inizio della realizzazione dei livelli creandoli in Unity ed iniziando a cercare i vari oggetti per la loro decorazione.
  - b. **Priorità 2:** In questa categoria andremo a creare il DataBase per la classifica ed è presente una sottocategoria per la programmazione vera e propria del gioco.
    - i. **Programmazione gioco:** in questa micro-categoria sono presenti tutte le parti per la programmazione del videogioco, dalla prima persona, alla programmazione dei mostri rendendo il gioco quasi completo.
  - c. **Priorità 3:** In questa categoria sono presenti tutte le schermate relative al gioco che sono:
    - i. **Schermata benvenuto:** Questa schermata è la schermata iniziale che dà il benvenuto nel gioco all'utente
    - ii. **Schermata impostazioni:** Questa schermata è la schermata che permette all'utente di mettere il gioco in pausa, inoltre permette all'utente di regolare il volume, di uscire dal gioco e di riprenderlo.
    - iii. **Schermata livelli:** Queste schermate appaiono all'inizio di ogni livello, esse comprendono una breve descrizione del livello e di quello che si deve fare per passarlo, inoltre comprende anche una descrizione dei pericoli che aspettano l'utente nello svolgersi del livello.
    - iv. **Schermata vincita:** Questa schermata è la schermata finale, quando l'utente riesce a scappare dall'ultimo livello appare questa schermata contenente le congratulazioni per la vincita dell'utente.

## 2.5 Analisi dei mezzi

### 2.5.1 Software

- Visual studio 2022
- Unity 2022.1

### 2.5.2 Hardware

- 2 PC identici: Le componenti dei PC che verranno utilizzati per lo sviluppo dell'applicativo sono:
  - I7-9700 @ 3.00GHz
  - RAM 32GB
  - SSD 512GB
  - NVIDIA GeForce RTX 2060

### 3 Progettazione

---

#### 3.1 Design dell'architettura del sistema

SQL lite in Unity

#### 3.2 Design dei dati e database

LeaderBoard	
ID	int
nickname	varchar(45)
time	int
place	int

ID -> È una chiave AUTO\_INCREMENT con la quale distinguiamo i player

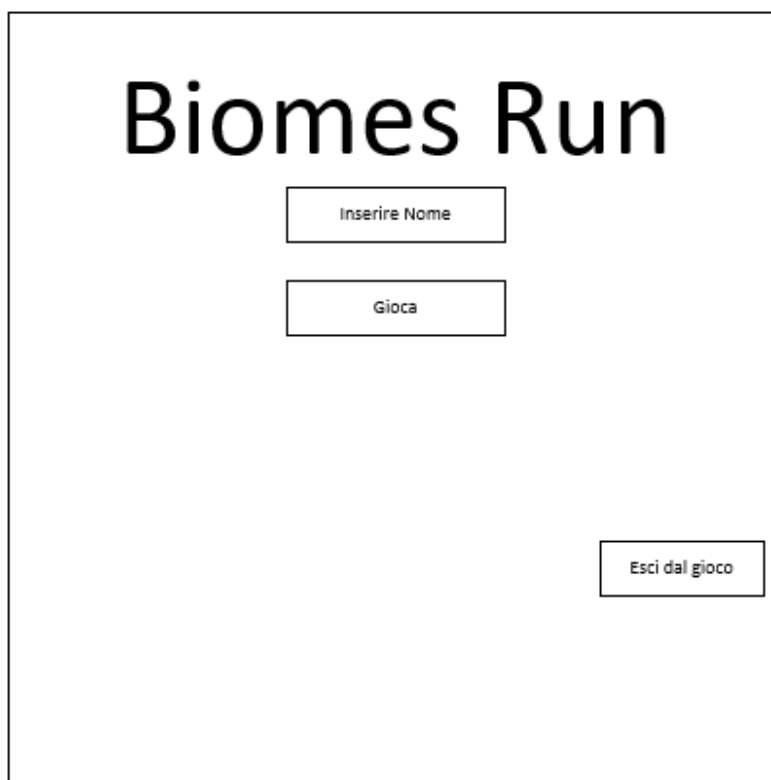
nickname -> È il nome utente del giocatore

time -> È il tempo che ha impiegato il giocatore per finire i 3 livelli

place -> È il posizionamento fatto ordinato in base al time (desc)

Figura 3 - Tabella classifica

### 3.3 Design delle interfacce



**Biomes Run**

Inserire Nome

Gioca

Esci dal gioco

Figura 4 - Schermata iniziale

La prima schermata che appare appena si apre il gioco è quella di benvenuto, essa è composta da:

- Il titolo del gioco.
- Un textbox dove l'utente dovrà inserire il suo nome che servirà per essere memorizzato nel DataBase per la classifica.
- Un bottone "Gioca" che permetterà all'utente di iniziare la partita
- Un bottone "Esci dal gioco" che permetterà all'utente di uscire dal gioco.

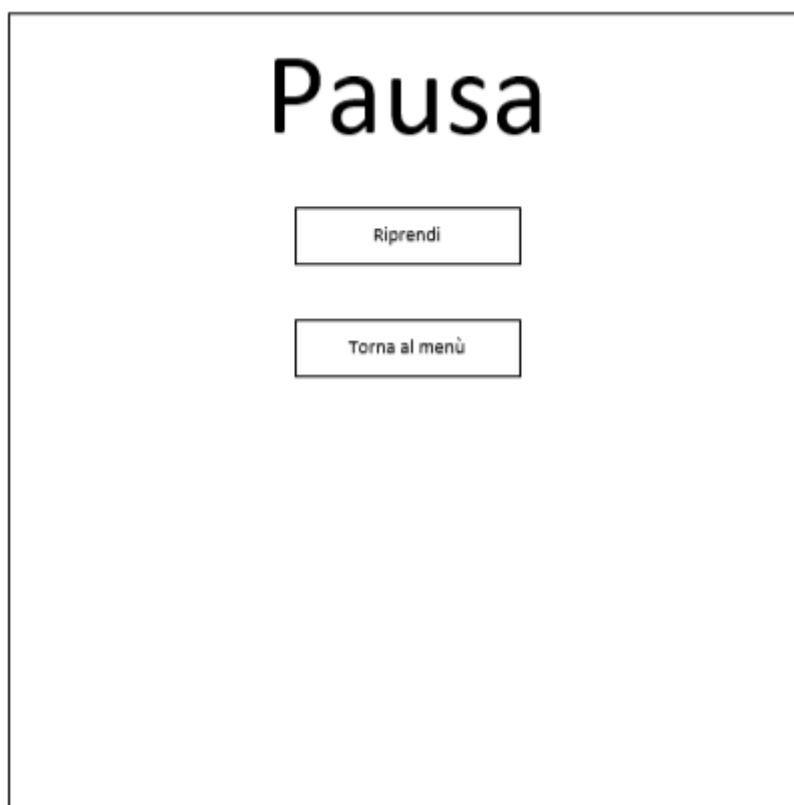


Figura 5 - Schermata di pausa

La schermata di pausa potrà essere accessibile durante il gameplay, l'utente potrà aprirla utilizzando il tasto "esc" e si aprirà questa schermata con:

- Un bottone "Riprendi" che riprenderà il gioco
- Un bottone "Esci dal gioco" che permetterà all'utente di uscire dal gioco.



Figura 6 - Schermata di vittoria

La schermata di vittoria apparirà all'utente quando riuscirà a completare anche l'ultimo livello, essa comprende:

- Il tempo impiegato per terminare il gioco
- La relativa posizione in classifica
- Un bottone "Rigioca" per ricominciare il gioco
- Un bottone "Torna al menù" per tornare alla schermata iniziale

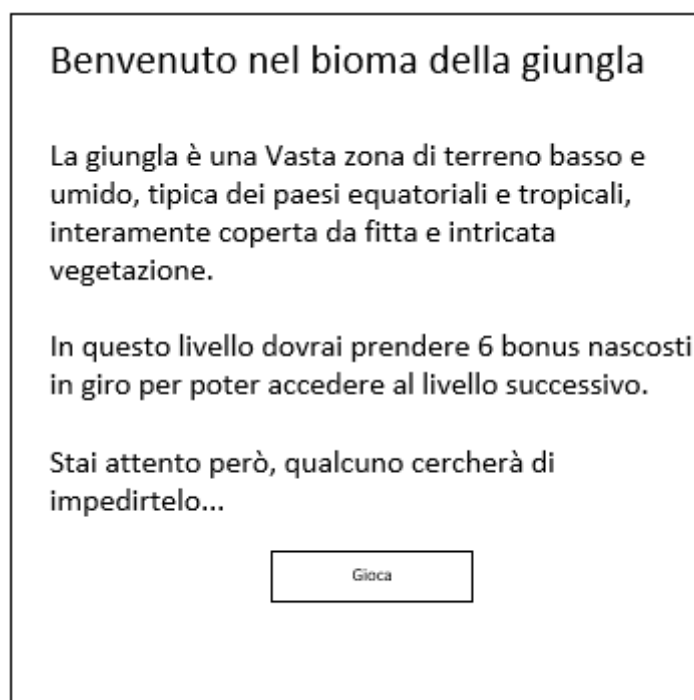


Figura 7 - Livello uno

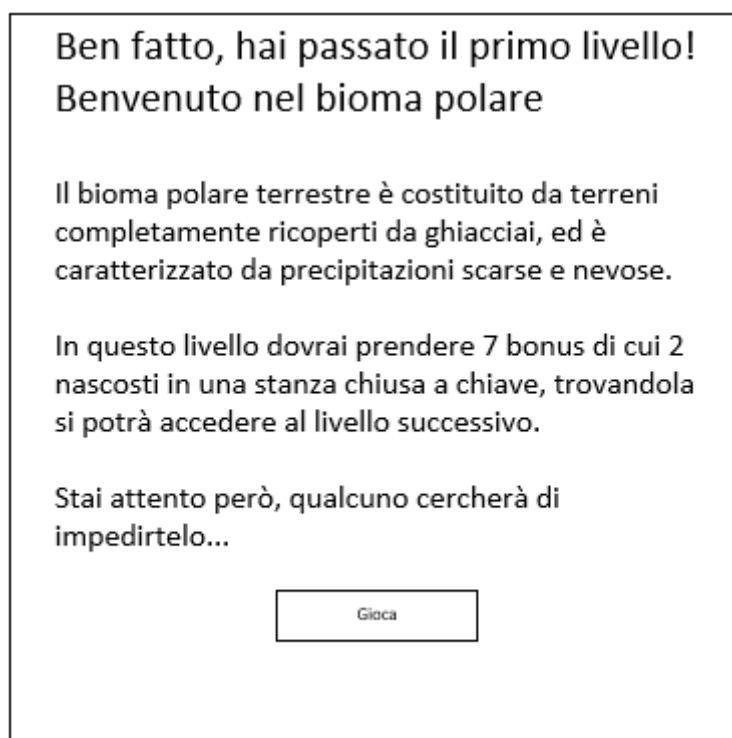


Figura 8 - Livello due



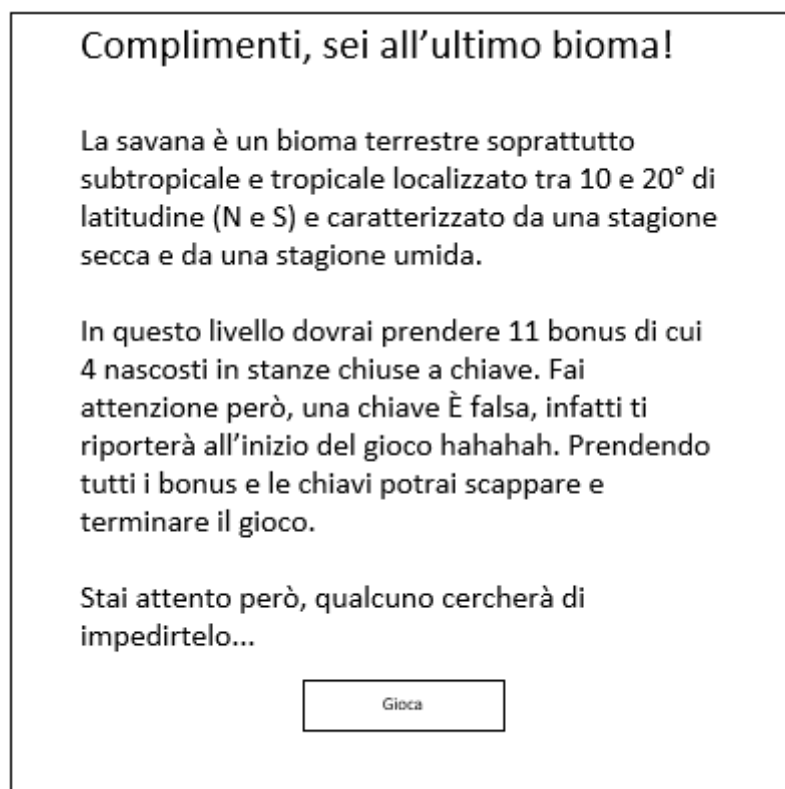


Figura 9 - Livello tre

Le schermate dei tre livelli apparranno sempre prima dell'inizio del relativo livello e contengono:

- Una descrizione del relativo bioma.
- La spiegazione di quello che l'utente dovrà fare per accedere al livello successivo
- Un bottone "Gioca" per iniziare il livello



Figura 10 - Schermata perdita chiave falsa

Questa schermata appare all'utente se nell'ultimo livello dovesse prendere la chiave sbagliata, essa contiene due bottoni:

- Il bottone "Continua" fa ricominciare il gioco all'utente
- Il bottone "Torna al menù" riporta l'utente alla schermata iniziale

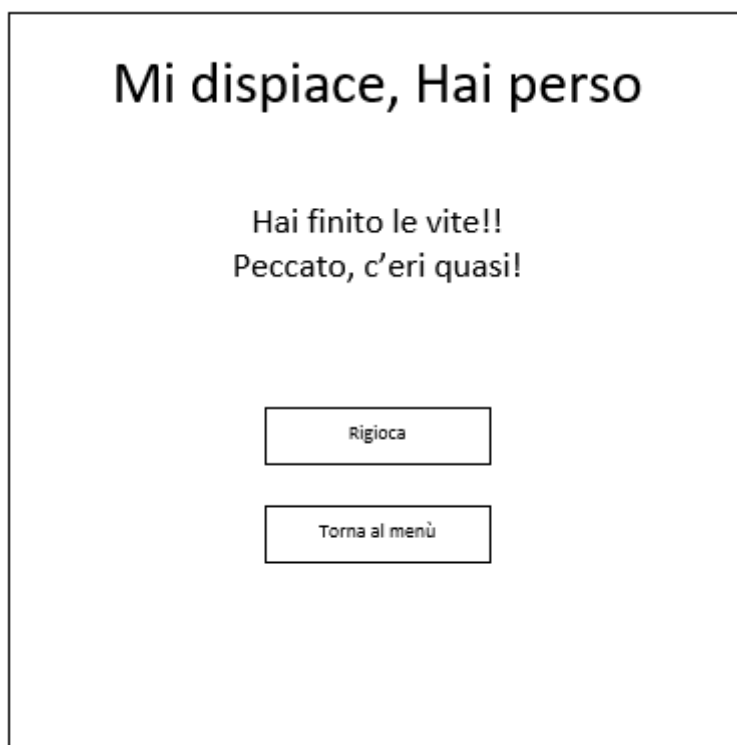


Figura 11 - Schermata perdita vite

Questa schermata appare all'utente se dovesse perdere tutte le vite per via de mostri. Essa contiene due bottoni:

- Il bottone "Rigioca" fa ricominciare il gioco all'utente
- Il bottone "Torna al menù" riporta l'utente alla schermata iniziale

### 3.4 Design procedurale

## 4 Implementazione

### 4.1 Costruzione labirinto

Per avere ordine nella barra della gerarchia creerò man mano durante la costruzione del labirinto degli empty object nei quali dividerò tutti i modelli 3d. Per esempio: ho creato un empty object chiamato Walls nel quale metterò tutti i muri

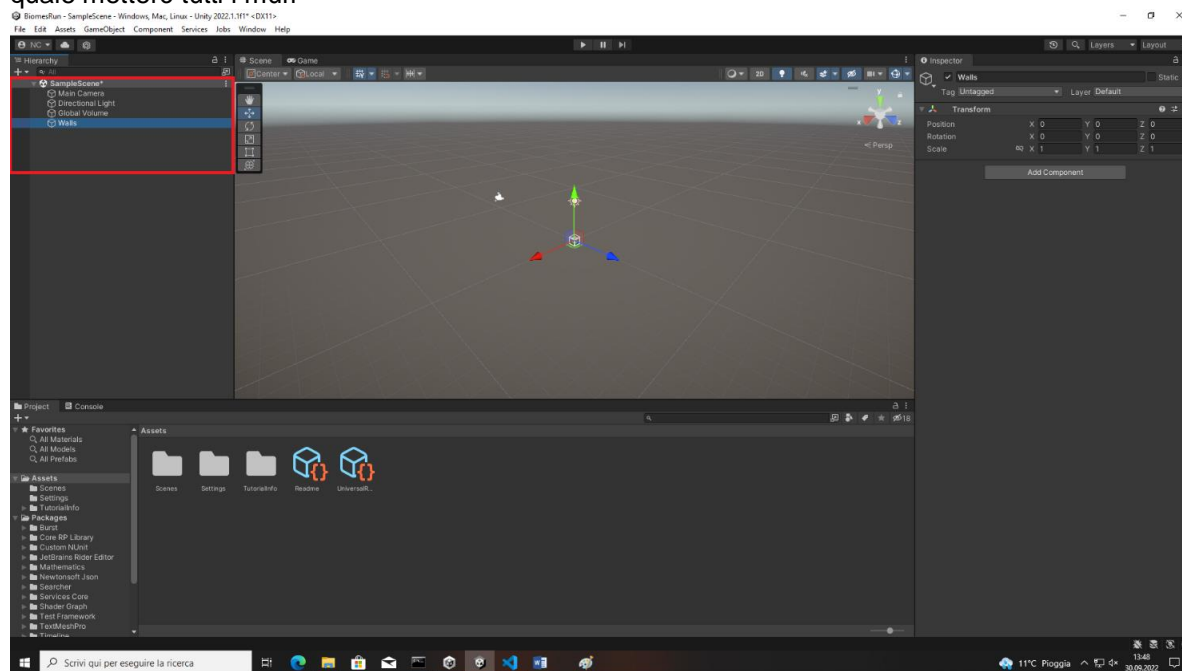


Figura 12 - Empty Object

Ho creato i muri e il piano per costruire la base del labirinto. Nella barra della gerarchia si vede che i muri sono nell' empty object Walls.

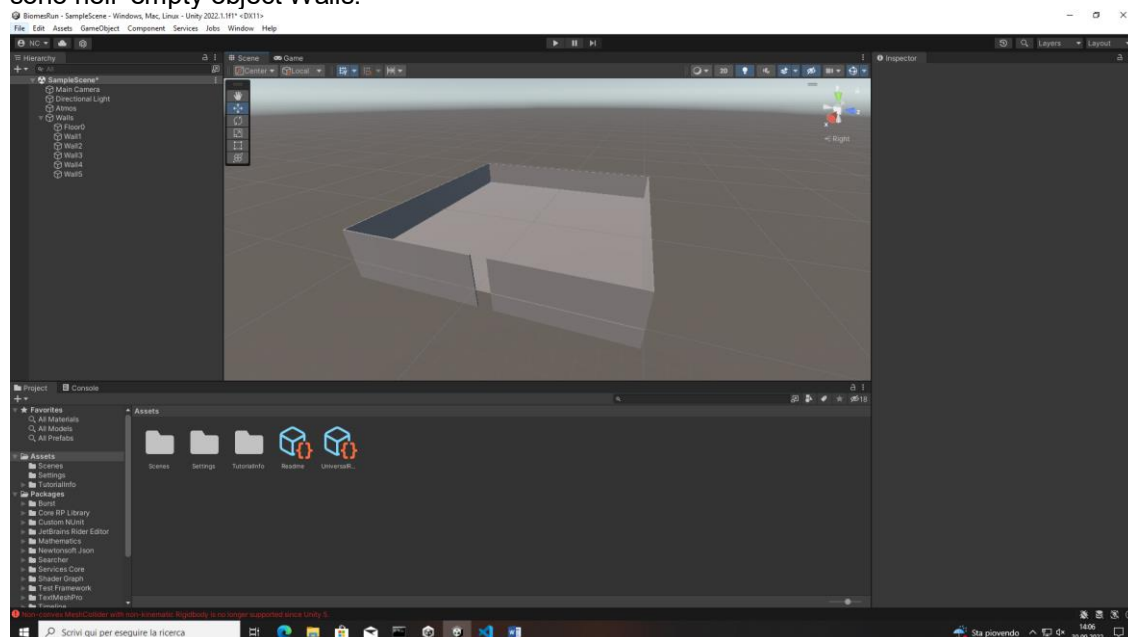


Figura 13 - Costruzione muri

#### 4.1.1.1 Layout labirinto dei 3 livelli

Prima di creare i labirinti direttamente su Unity, ho sviluppato casualmente il loro layout con Paint.

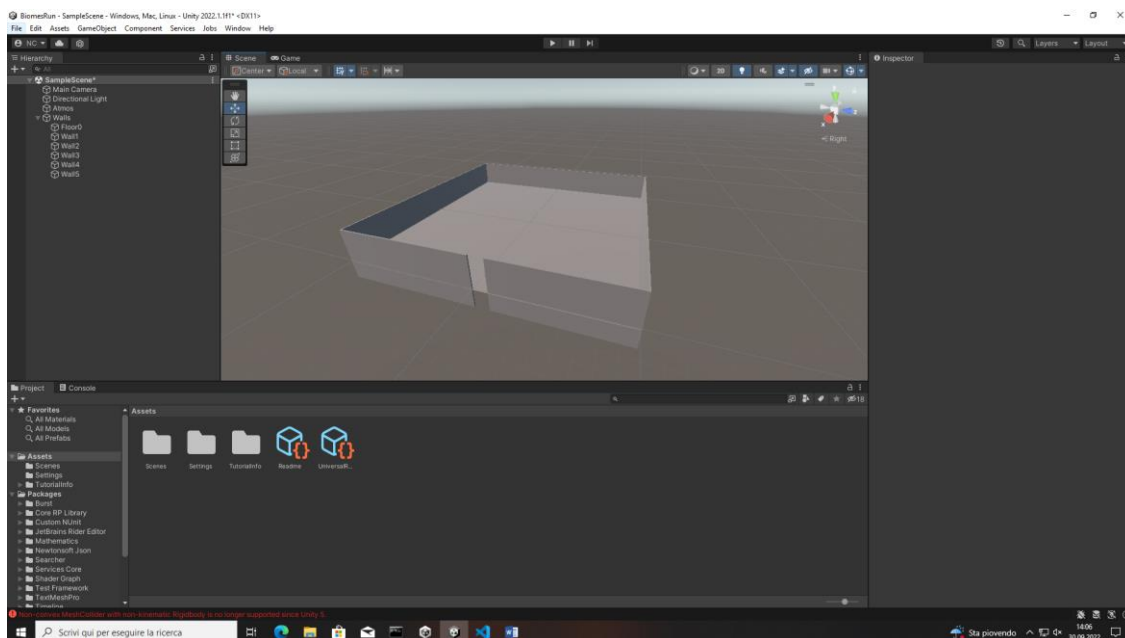


Figura 14 – Inizio costruzione labirinto

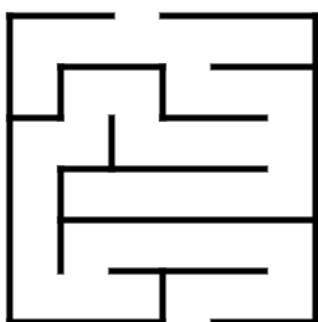


Figura 17 – Layout livello 1

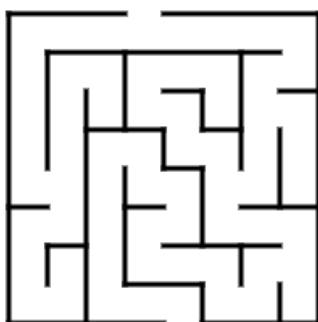


Figura 16 - Layout livello 2

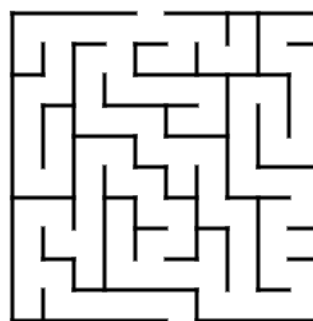


Figura 15 - Layout livello 3

## 4.2 Costruzione 3 livelli in Unity

Ho fatto il layout dei 3 labirinti seguendo più o meno i layout creati in Unity.

Legenda dei colori utilizzati:

- I buchi presenti in alto a sinistra dei labirinti segnano il punto di inizio di ogni livello
- La parete bucata in mezzo è la fine del labirinto, e quando si avrà varcato la porta c si ritroverà in alto a sinistra del livello successivo.
- I muri verdi rappresentano le porte quali si apriranno con le chiavi mentre il muro verde rappresenta una porta falsa, in quella stanza sarà presente la chiave falsa, la quale ti farà riiniziare il gioco

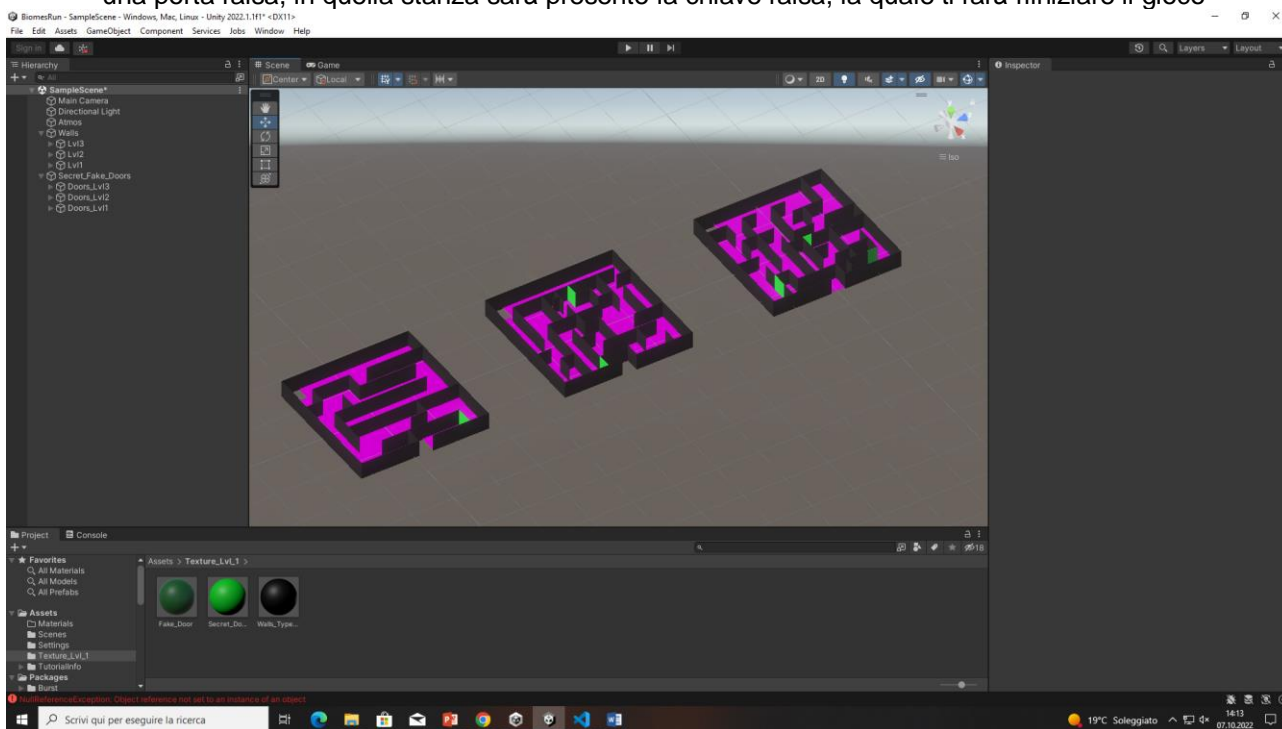


Figura 18 - Layout 3 labirinti

## 4.3 Decorazione Labirinto

### 4.3.1 Modelli 3D

Per decorare il labirinto ho scaricato dei modelli 3D (da <https://kenney.nl/assets/nature-pack> ) e gli ho messi in una cartella che mi contiene tutti gli oggetti. Per avere ordine ho creato una struttura di cartelle in questo modo:

Livello\_nr/3D Models/\*.fbx

Ogni livello ha la propria cartella dei modelli 3D.

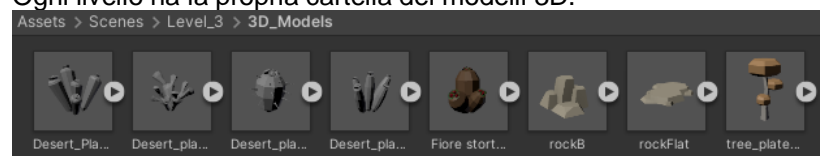


Figura 19 - Modelli 3D

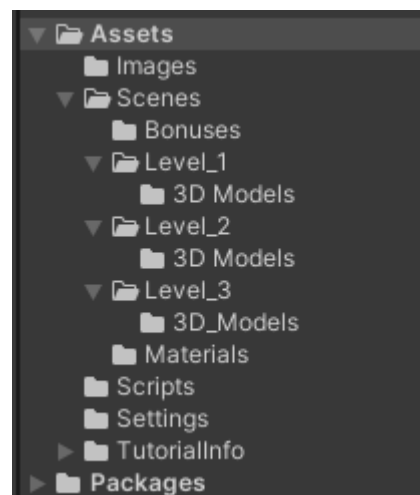


Figura 18 - Gerarchia modelli

### 4.3.2 Decorazione labirinto

Ho decorato i 3 labirinti utilizzando i modelli 3D scaricati dal [sito](#) di prima. I 3 labirinti gli ho decorati in base ad un bioma (vedi [qdc](#))

- Livello 1 è basato sul bioma della giungla



Figura 20 - Decorazione livello1

- Livello 2 è basato su un bioma polare



Figura 21 - Decorazione livello 2



- Livello 3 è basato sul bioma della savana

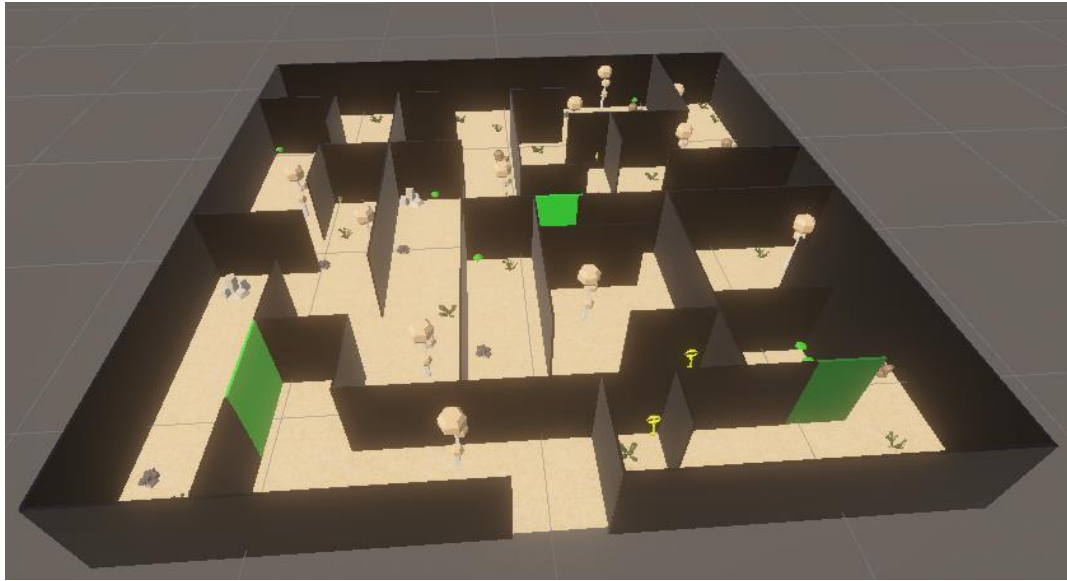


Figura 22 - Decorazione livello3

### 4.3.3 Albero cartelle

Alla fine della decorazione per avere ordine nei miei oggetti ho creato diversi empty object per smistare il tutto per livelli.

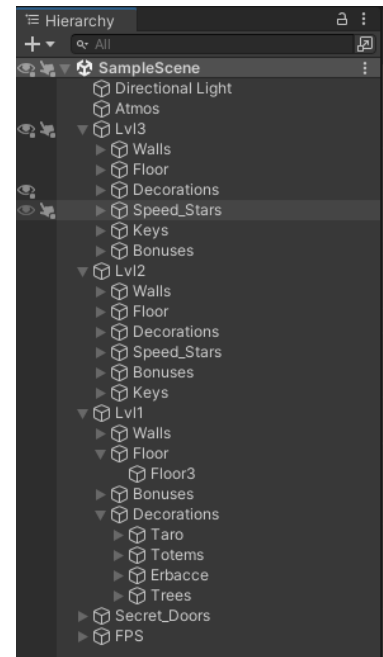


Figura 23 - Ordine oggetti

### 4.3.4 Script Rotatore

Chiavi, stelline e punti bonus gli ho fatti roteare su sé stessi utilizzando uno script. Ho creato la mia cartella degli Scripts in Assets e ho creato uno script chiamato KeyRotator, all'interno ho scritto questo codice:

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

public class KeyRotator : MonoBehaviour
{
    public Vector3 rotation;

    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        GetComponent<Transform>().Rotate(rotation);
    }
}
```

Figura 24 - Script rotatore oggetti

E su unity collego lo script agli oggetti, e nello script potrò scegliere la velocità e verso che verso farli girare

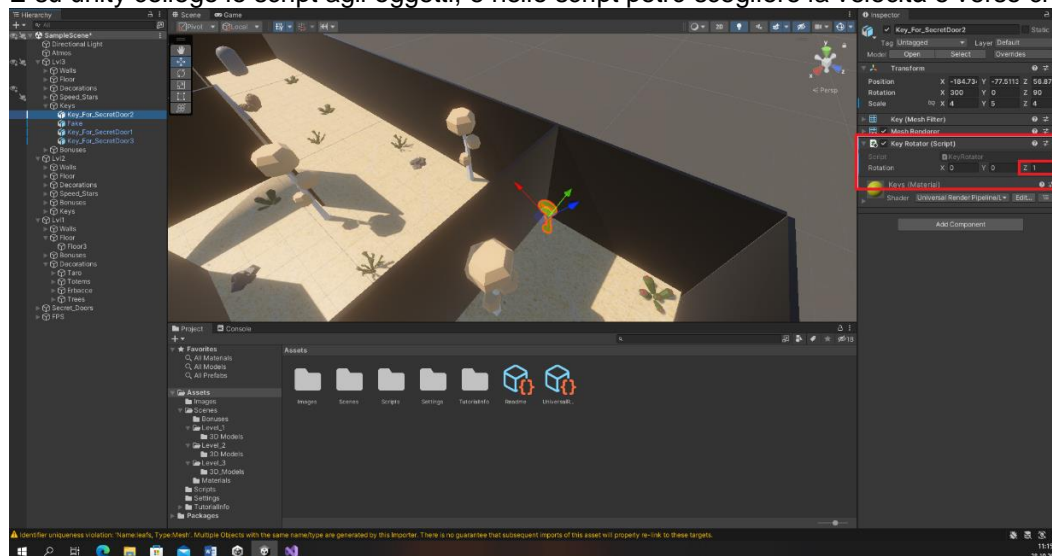


Figura 25 - Chiave con script rotazione

#### 4.4 Prima persona

Per il movimento in prima persona ho creato un piano al quale ho aggiunto il componente Rigidbody, ho tolto la gravità e ho reso statico per far sì che non si sposti durante la partita

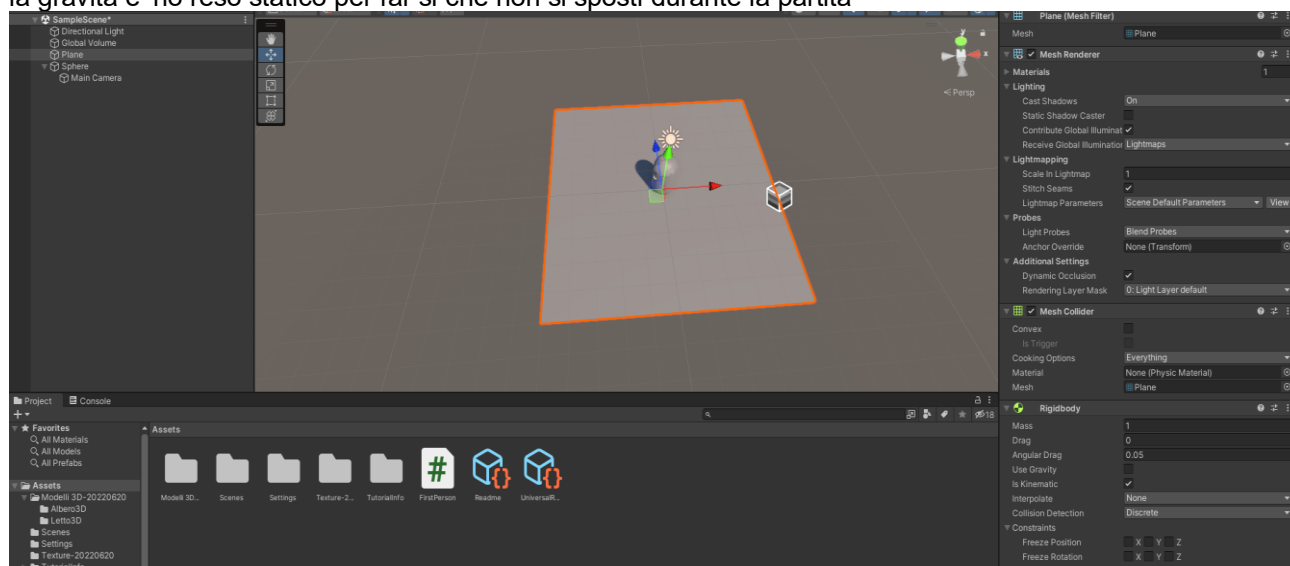


Figura 26 - Pavimento Grezzo

Per introdurre la camera in prima persona ho creato un personaggio grezzo (Capsule) al quale ho bloccato la rotazione X e Z perché altrimenti si sarebbe spostato.

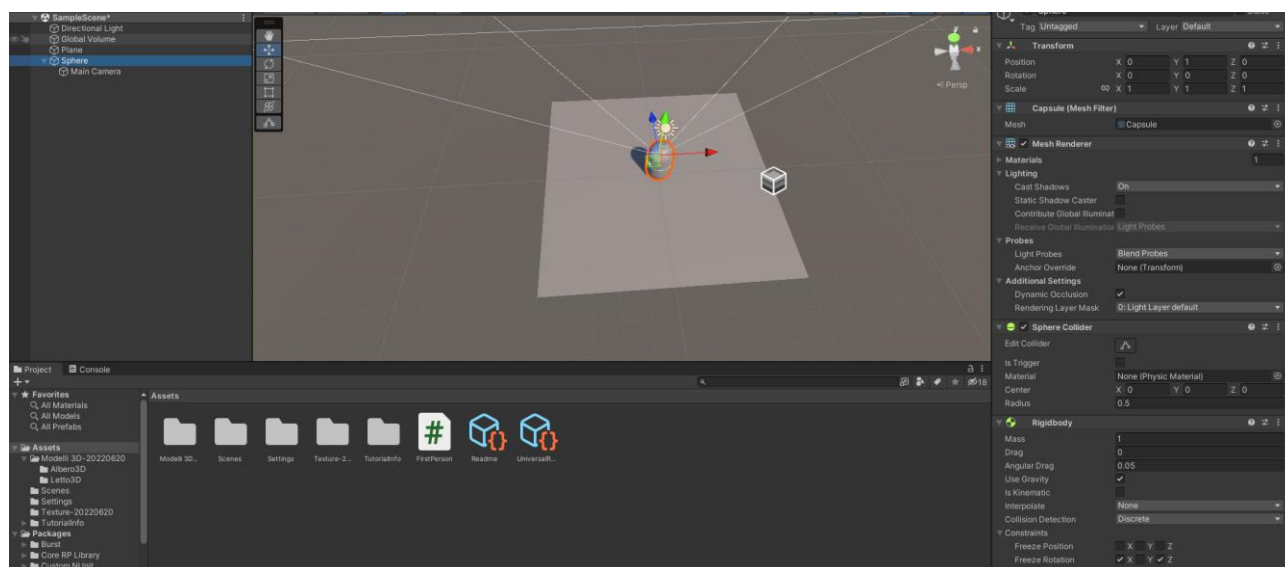
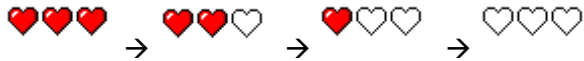


Figura 27 - Personaggio Grezzo

Inoltre per il movimento effettivo in prima persona ho implementato nella gerarchia della sfera la camera ed ho inserito il seguente script:

#### 4.5 Vite personaggio

Per creare le vite del personaggio ho utilizzato un sito che mi ha permesso di avere delle immagini con background trasparente, il sito si chiama Piskel. Il risultato è questo:



Per poter avere un player con delle vite ho dovuto creare all'interno del gameObject del player un canvas con all'interno un'immagine contenente l'immagine delle tre vite:

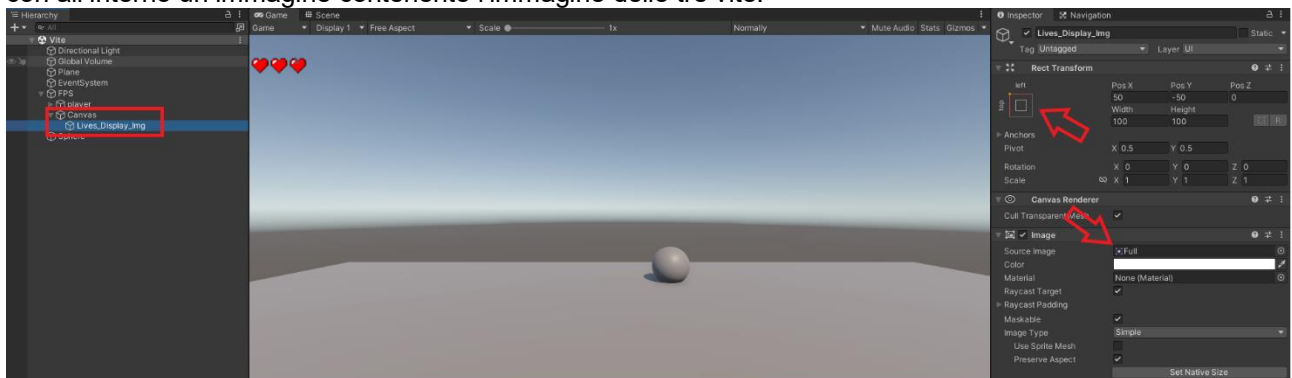


Figura 28 - Inserimento immagine vite

Dopodiché ho creato uno script contenente un array di immagini dove ogni volta che il player viene colpito, incrementa una variabile che aumenta la posizione dell'array e l'immagine delle vite cambia. Dopo averlo creato ho inserito nella lista dello script le vite.

```

13 public Image imgLives;
14
15 private int lives = 3;
16
17 [SerializeField]
18 private Sprite[] _liveSprites;
19
20 0 riferimenti
21 public IEnumerator WaitSec()
22 {
23     yield return new WaitForSeconds(1000);
24 }
25
26 1 riferimento
27 public int updateLives(int currentLives)
28 {
29     imgLives.sprite = _liveSprites[currentLives];
30     return currentLives;
31 }
32
33 * Messaggio Unity | 0 riferimenti
34 public void OnCollisionEnter(Collision collision)
35 {
36     if (collision.gameObject.tag == "mostro")
37     {
38         lives--;
39         imgLives.sprite = _liveSprites[updateLives(lives)];
40     }
41     if (lives == 0)
42     {
43         SceneManager.LoadScene(7);
44     }
45 }

```

Figura 29 - Script vite

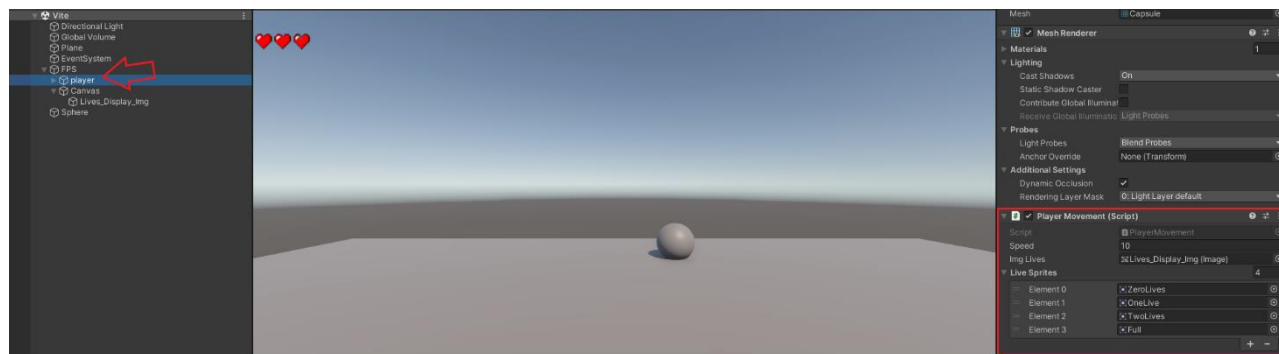


Figura 30 - Inserimento immagini nell'array

Facendo così basta dare al nemico il tag "mostro" e il gioco è fatto. Se il player dovesse finire le vite, verrebbe caricata la schermata di perdita.

## 4.6 Programmazione mostro

Per la programmazione dei mostri ho creato uno script che inseguisse il player. Inoltre bisogna “dire” al nemico dove può passare e non grazie al bake.

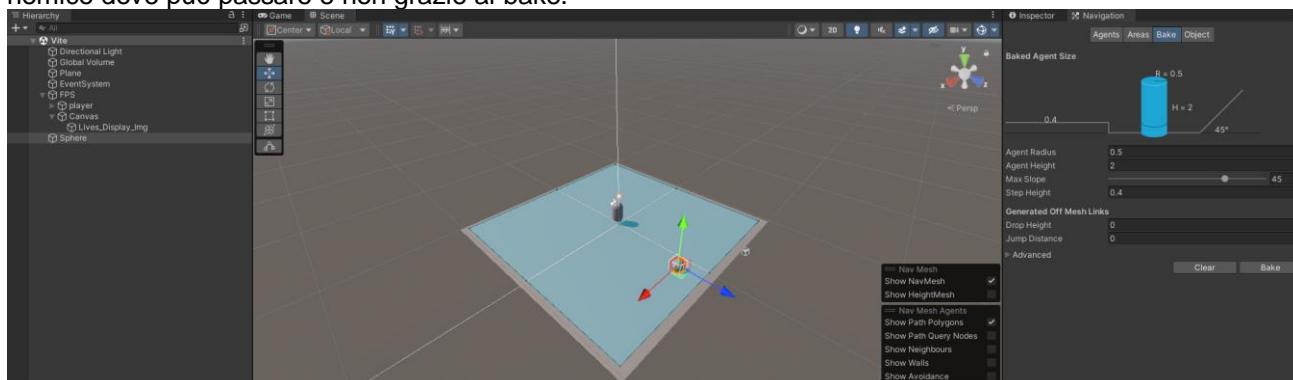


Figura 31 - Bake del mostro

Il mostro deve chiamarsi mostro per poter far funzionare il counter delle vite

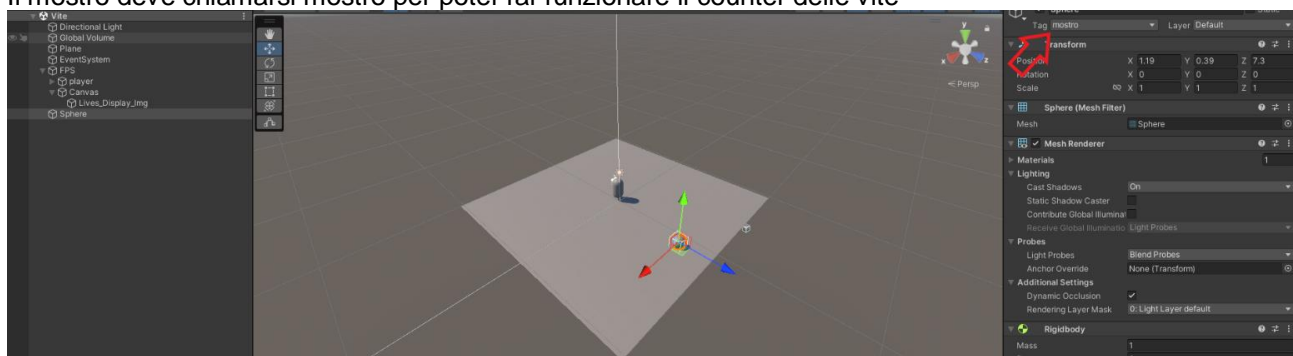


Figura 32 - tag del mostro

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.AI;
5  using static UnityEngine.GraphicsBuffer;
6
7  [Script Unity (1 riferimento ad asset) | 0 riferimenti]
8  public class Enemy : MonoBehaviour
9  {
10     public Transform Target;
11     // Start is called before the first frame update
12     [Messaggio Unity | 0 riferimenti]
13     void Start()
14     {
15     }
16
17     // Update is called once per frame
18     [Messaggio Unity | 0 riferimenti]
19     void Update()
20     {
21         NavMeshAgent agent = GetComponent<NavMeshAgent>();
22         agent.destination = Target.position;
23     }
24 }

```

Figura 33 - script mostro

## 4.7 Programmazione schermate

Oltre alla schermata delle vite, ho creato diverse schermate:

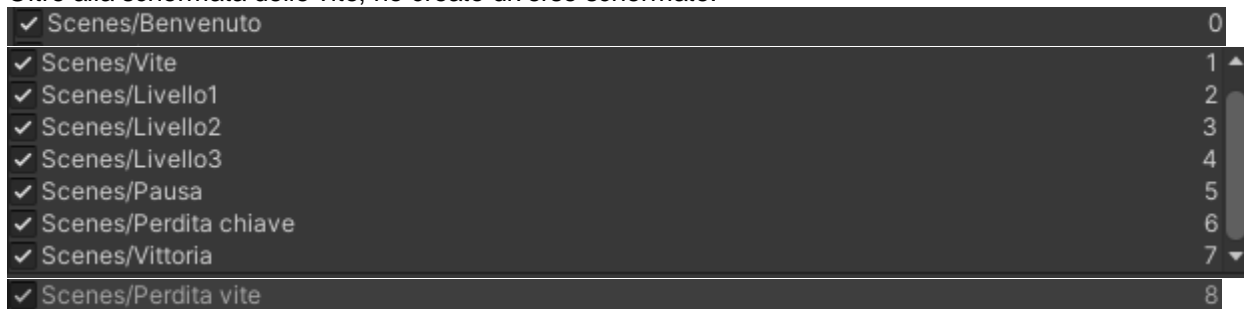


Figura 34 - Ordine schene

### 4.7.1 Schermata 0 (Benvenuto)

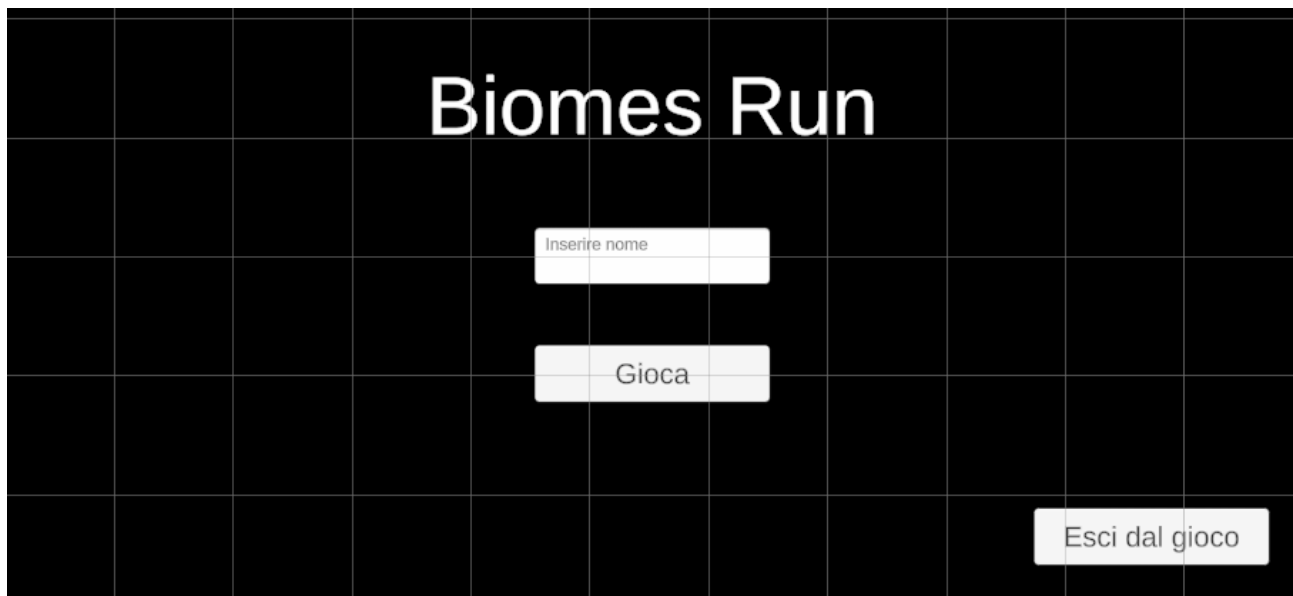


Figura 35 - Schermata 0 (Benvenuto)



#### 4.7.2 Schermata 1 (Vite) Schermata di gioco

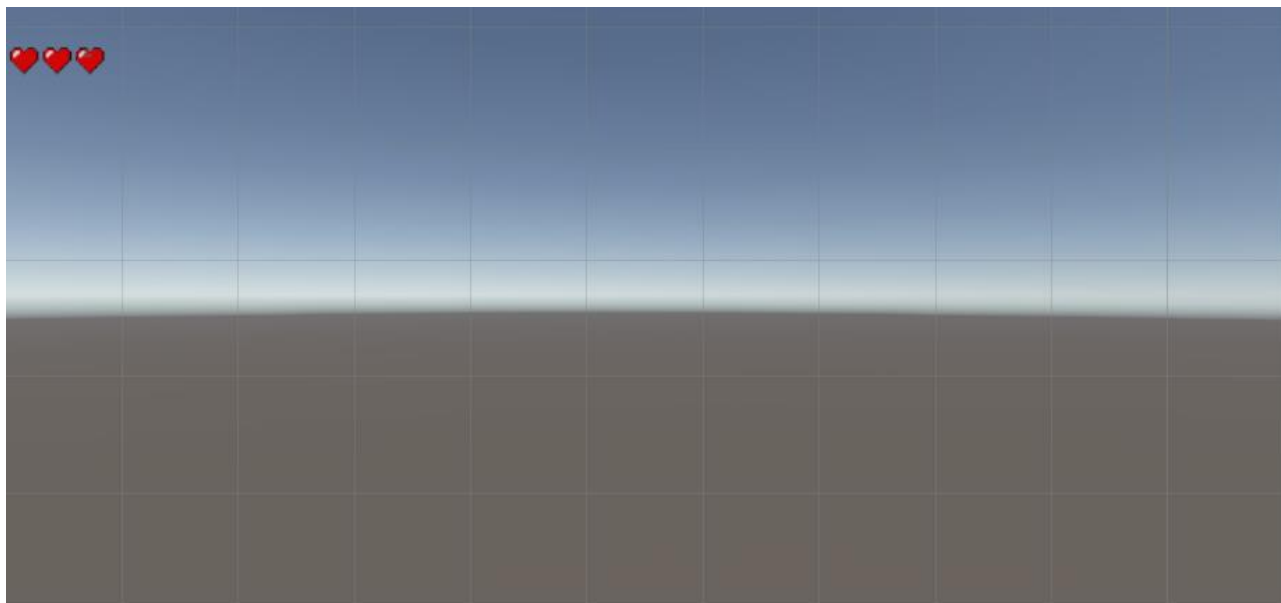


Figura 36 - Schermata 1 (Vite) Schermata di gioco

#### 4.7.3 Schermata 2 (Livello1)

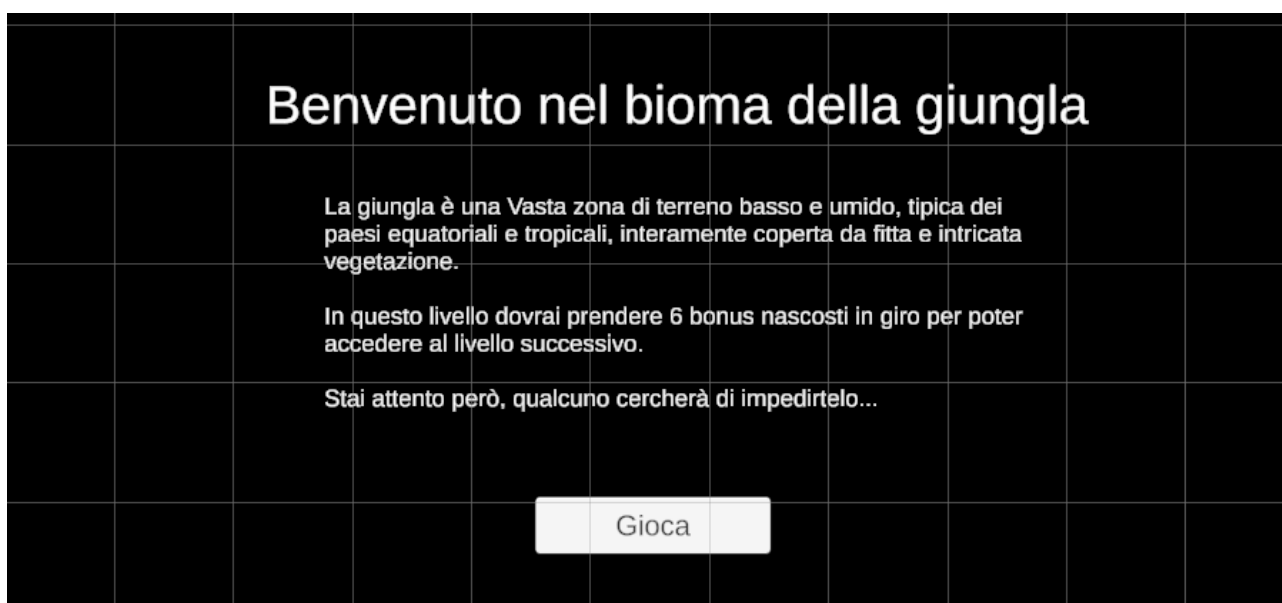


Figura 37 - Schermata 2 (Livello 1)

#### 4.7.4 Schermata 3 (Secondo livello)

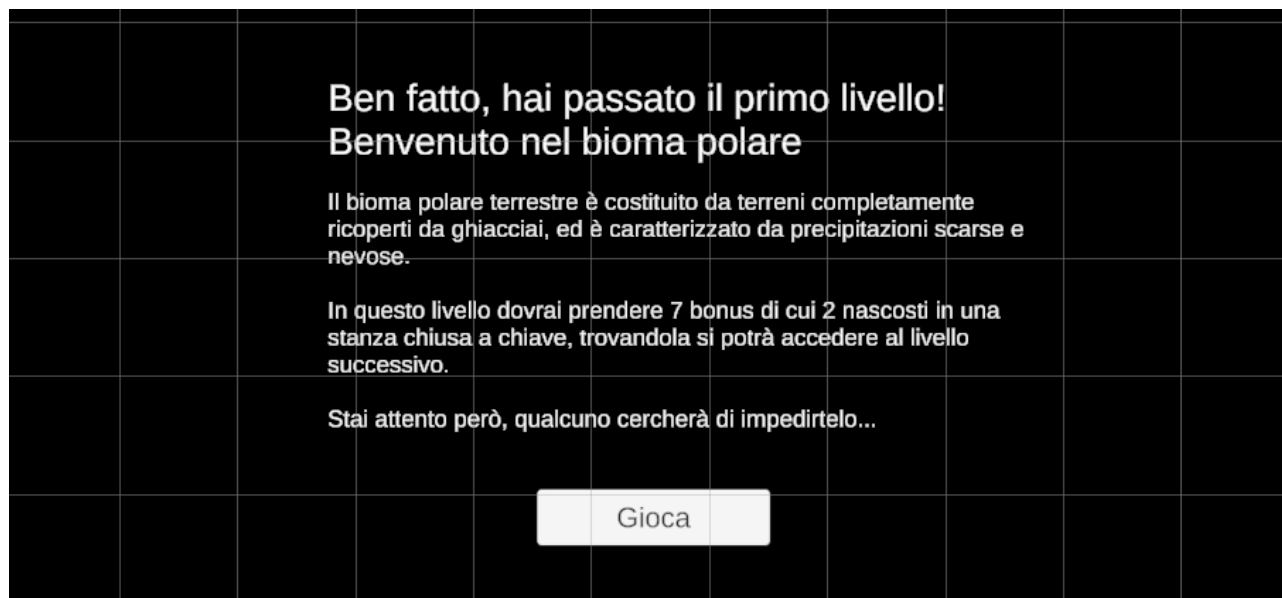


Figura 38 - Schermata 3 (Livello 2)

#### 4.7.5 Schermata 4 (Livello3)

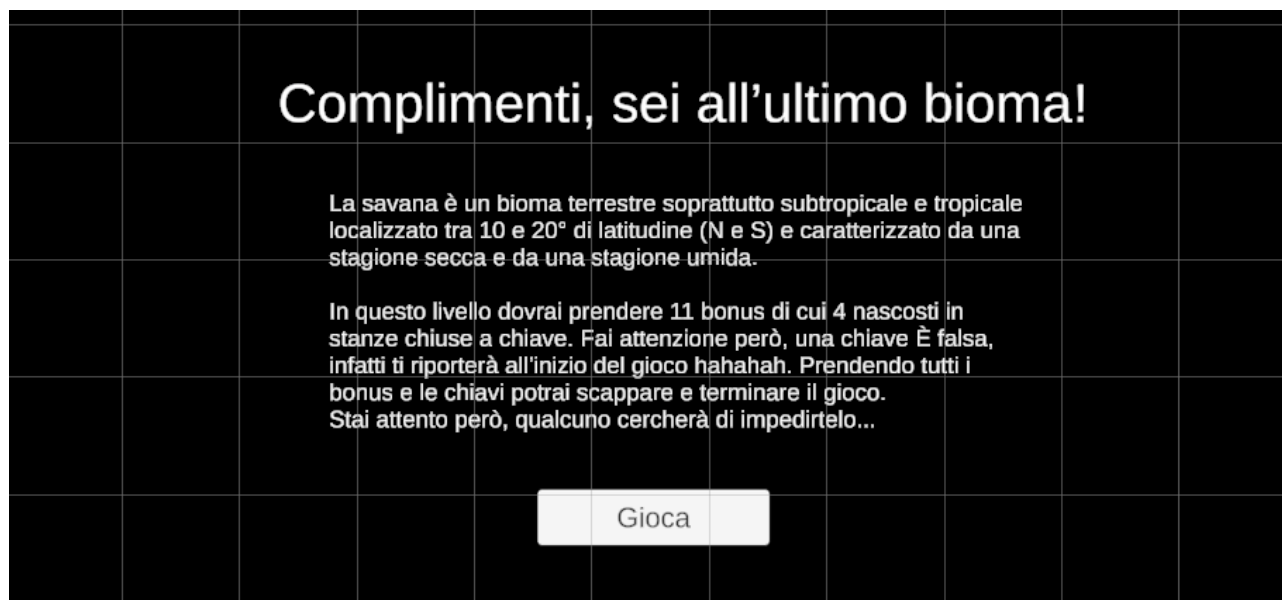


Figura 39 - Schermata 4 (Livello 3)

#### 4.7.6 Schermata 5 (Pausa)



Figura 40 - Schermata 5 (Pausa)

#### 4.7.7 Schermata 6 (Chiave falsa)

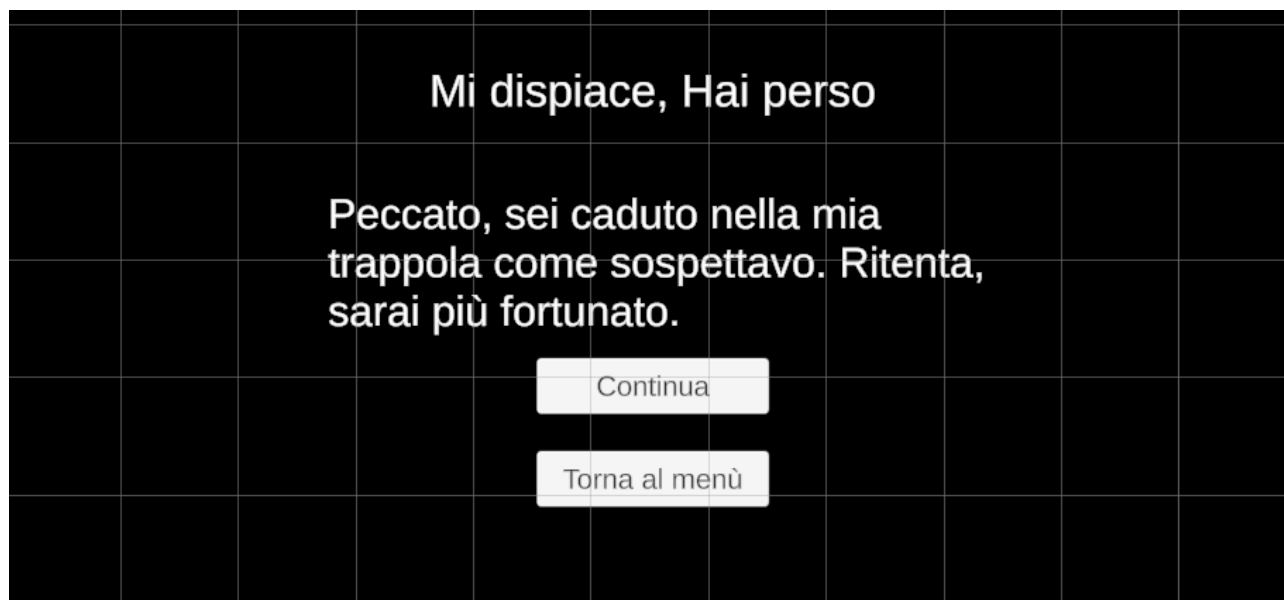


Figura 41 - Schermata 6 (Chiave falsa)

#### 4.7.8 Schermata 7 (Vittoria)

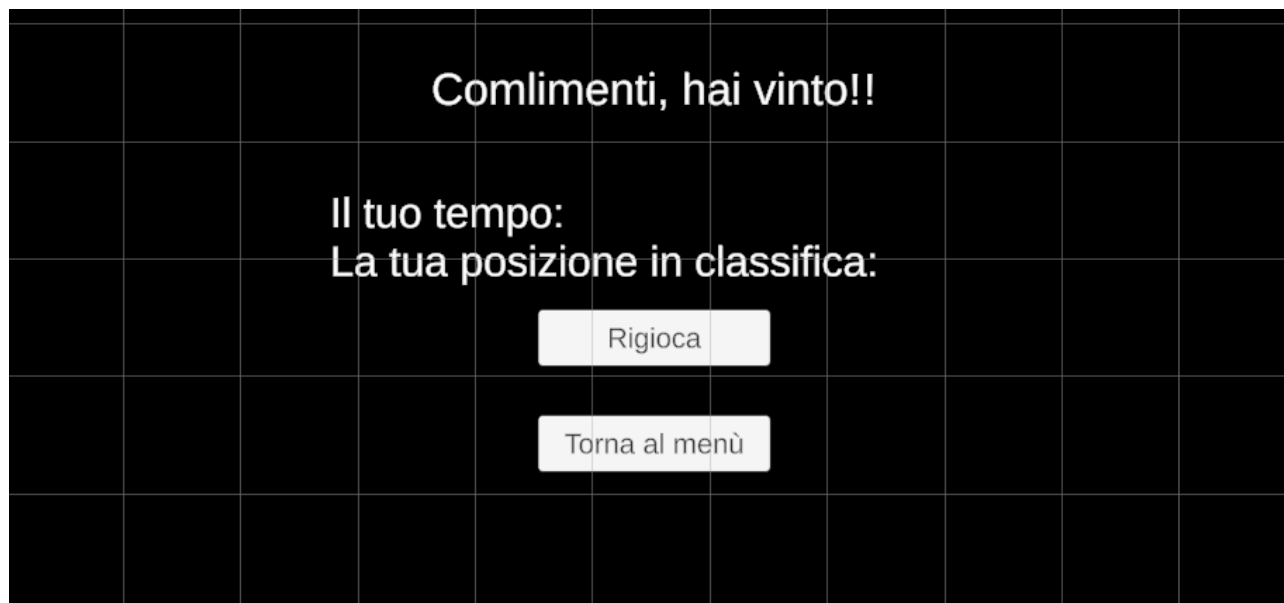


Figura 42 - Schermata 7 (Vittoria)

#### 4.7.9 Schermata 8 (Vite finite)



Figura 43 - Schermata 8 (Vite finite)

#### Aggiungere

- Script movimento e prima persona
- Script gestione schermate

## 5 Test

### 5.1 Protocollo di test

<b>Test Case:</b>	TC-002	<b>Nome:</b>	Prima persona
<b>Riferimento</b>	REQ-002		
<b>Descrizione</b>	Il movimento del personaggio è di 360° in orizzontale e 180° e l'utente sposterà la visuale con il mouse		
<b>Prerequisiti</b>	- Layout personaggio completato		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Iniziare la partita</li> <li>3. Muovere la visuale</li> </ol>		
<b>Risultati attesi</b>	Facendo partire il gioco, l'utente è in grado di cambiare la visuale del personaggio utilizzando il mouse		

<b>Test Case:</b>	TC-003	<b>Nome:</b>	Movimento personaggio
<b>Riferimento</b>	REQ-003		
<b>Descrizione</b>	Durante la partita, l'utente può muovere il personaggio tramite "WASD" all'interno dei labirinti		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Labirinti completati</li> <li>- Personaggio completato</li> <li>- Visuale personaggio completata</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire gioco</li> <li>2. Iniziare la partita</li> <li>3. Muovere il personaggio</li> </ol>		

<b>Test Case:</b>	TC-004	<b>Nome:</b>	Decorazione labirinto
<b>Riferimento</b>	REQ-004		
<b>Descrizione</b>	Raccolta di oggetti per la decorazione dei labirinti (bonus, chiavi, mostri, stelline, personaggio), inserimento luci e musiche		
<b>Prerequisiti</b>	- Livelli completati		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Iniziare la partita</li> <li>3. Muoversi nel labirinto</li> </ol>		
<b>Risultati attesi</b>	Alla fine si avranno i livelli completati definitivamente, con tutti gli interni completi.		

<b>Test Case:</b>	TC-005	<b>Nome:</b>	Programmazione mostri
<b>Riferimento</b>	REQ-005		
<b>Descrizione</b>	L'utente dovrà scappare dai guardiani dei biomi che avranno come obiettivo quello di far perdere tutte le vite all'utente		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Raccolta template mostri</li> <li>- Labirinti completati</li> <li>- Personaggio completato</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire gioco</li> <li>2. Iniziare la partita</li> <li>3. Aspettare l'arrivo dei mostri</li> </ol>		
<b>Risultati attesi</b>	I mostri dovranno inseguire l'utente		

<b>Test Case:</b>	TC-006	<b>Nome:</b>	Vite personaggio
<b>Riferimento</b>	REQ-006		
<b>Descrizione</b>	Quando l'utente viene toccato da un mostro perde una vita, quando perde tutte le vite il gioco finisce.		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Personaggio completato</li> <li>- Mostri completati</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Iniziare la partita</li> <li>3. Farsi colpito dai mostri</li> </ol>		
<b>Risultati attesi</b>	I mostri dovranno inseguire l'utente e se lo toccano, esso perde una vita		

<b>Test Case:</b>	TC-007	<b>Nome:</b>	Funzionamento schermata iniziale
<b>Riferimento</b>	REQ-007		
<b>Descrizione</b>	L'utente ha la possibilità di inserire il proprio nome e visualizzare la classifica dei tempi migliori.		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Design schermata iniziale completata</li> <li>- Gioco completato</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Inserire un nome</li> <li>3. Schiacciare il bottone classifica</li> <li>4. Schiacciare il bottone gioca</li> </ol>		
<b>Risultati attesi</b>	L'utente potrà visualizzare la classifica cliccando sul pulsante classifica, inoltre potrà inserire il suo nome ed impostare il volume, infine l'utente potrà iniziare una partita cliccando sul tasto "Gioca".		

<b>Test Case:</b>	TC-008	<b>Nome:</b>	Funzionamento
<b>Riferimento</b>	REQ-008		schermata impostazioni
<b>Descrizione</b>	L'utente ha la possibilità di aprire le impostazioni durante il corso della partita		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Design schermata impostazioni completata</li> <li>- Gioco completato</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Aprire la schermata impostazione</li> <li>3. Verificare i vari pulsanti</li> </ol>		
<b>Risultati attesi</b>	L'utente potrà visualizzare nella schermata impostazioni tramite che il tasto "ESC", potrà riprendere la partita cliccando sul bottone riprendi, uscire dal gioco schiacciando sul bottone "Torna al menù" ed impostare il volume		

<b>Test Case:</b>	TC-009	<b>Nome:</b>	DataBase
<b>Riferimento</b>	REQ-009		
<b>Descrizione</b>	Il DataBase contiene tutti i nomi dei giocatori e i loro relativi tempi che saranno inseriti nella classifica		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Schermata delle impostazioni e schermata iniziale finite</li> <li>- Gioco completato</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Schiacciare sul pulsante "Classifica"</li> <li>3. Vedere i posizionamenti</li> </ol>		
<b>Risultati attesi</b>	L'utente potrà visualizzare la classifica nella schermata iniziale con il tasto "Classifica" e vedere così il suo tempo effettivo.		

<b>Test Case:</b>	TC-010	<b>Nome:</b>	Funzionamento
<b>Riferimento</b>	REQ-010		schermata vincita
<b>Descrizione</b>	L'utente, quando finirà la partita, potrà vedere subito il suo tempo per completare il gioco.		
<b>Prerequisiti</b>	<ul style="list-style-type: none"> <li>- Database finito</li> <li>- Gioco completato</li> </ul>		
<b>Procedura</b>	<ol style="list-style-type: none"> <li>1. Aprire il gioco</li> <li>2. Finire il gioco</li> <li>3. Controllare il tempo effettivo e la posizione</li> </ol>		
<b>Risultati attesi</b>	L'utente potrà visualizzare alla fine della partita il suo tempo e la sua posizione in classifica.		

## **5.2 Risultati test**

## **5.3 Mancanze/limitazioni conosciute**

## **6 Consuntivo**

## **7 Conclusioni**

---

### **7.1 Sviluppi futuri**

### **7.2 Considerazioni personali**

## **8 Glossario**

---

## **9 Bibliografia**

---

### **9.1 Bibliografia per articoli di riviste:**

### **9.2 Bibliografia per libri**

### **9.3 Sitografia**

## **10 Allegati**

---