**T.A.G.  
Target Advanced Game**

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

2 Analisi 4

2.1 Analisi del dominio 4

2.2 Analisi e specifica dei requisiti 4

2.2.1 Spiegazione elementi tabella dei requisiti: 5

2.3 Use case 5

2.4 Pianificazione 5

2.5 Analisi dei mezzi 5

2.5.1 Software 6

2.5.2 Hardware 6

3 Progettazione 6

3.1 Design dell’architettura del sistema 6

3.2 Design dei dati e database 6

3.3 Design delle interfacce 6

3.4 Design procedurale 6

4 Implementazione 7

5 Test 7

5.1 Protocollo di test 7

5.2 Risultati test 8

5.3 Mancanze/limitazioni conosciute 8

6 Consuntivo 8

7 Conclusioni 8

7.1 Sviluppi futuri 8

7.2 Considerazioni personali 8

8 Glossario 8

9 Bibliografia 9

9.1 Bibliografia per articoli di riviste: 9

9.2 Bibliografia per libri 9

9.3 Sitografia 9

10 Allegati 9

[Figura 1 - Interfaccia iniziale 9](#_Toc126315305)

[Figura 2 - Interfaccia Impostazioni 10](#_Toc126315306)

[Figura 3 – Interfaccia base di gioco 11](#_Toc126315307)

[Figura 4 - Interfaccia puntatore attivo 11](#_Toc126315308)

[Figura 5 - Interfaccia scudo attivo 12](#_Toc126315309)

[Figura 6 - Interfaccia classifica 12](#_Toc126315310)

# Introduzione

## Informazioni sul progetto

* Allievi: Simone Riva, Michael Dobeson, Stefano Ceschi
* Classe: I3AC-BC
* Docente responsabile: Michel Palucci
* Data inizio: 27.01.2023
* Data fine: 5.05.2023

## Abstract

*T.A.G. is a virtual recreation of the Laser Tag game, that is set in the future with a futuristic environment and characters.*

*In Laser Tag everyone is against everyone and you have to point your laser towards the enemy to disable his laser and earn a point.*

*In our game there will be a map with randomly generated obstacles where the player has to hit other enemies/other players with their laser pointer. You can also defend yourself using a shield to stop the laser from touching you.*

*The player is also able to see his scores in real time thanks to a leaderboard in the center of the map, he can also see the top statistics through his registered account on a website with a Database.*

## Scopo

Lo scopo del progetto è di avere una variazione virtuale del gioco laser tag con una generazione casuale di oggetti all’interno del campo di gioco per stimolare l’utente ad inventarsi una strategia di gioco diversa ad ogni partita.Analisi

## Analisi del dominio

Questo capitolo dovrebbe descrivere il contesto in cui il prodotto verrà utilizzato, da questa analisi dovrebbero scaturire le risposte a quesiti quali ad esempio:

* Background/Situazione iniziale
* Quale è e come è organizzato il contesto in cui il prodotto dovrà funzionare?
* Come viene risolto attualmente il problema? Esiste già un prodotto simile?
* Chi sono gli utenti? Che bisogni hanno? Come e dove lavorano?
* Che competenze/conoscenze/cultura posseggono gli utenti in relazione con il problema?
* Esistono convenzioni/standard applicati nel dominio?
* Che conoscenze teoriche bisogna avere/acquisire per poter operare efficacemente nel dominio?
* …

## Analisi e specifica dei requisiti

Priorità 2/3 = opzionale

|  |  |
| --- | --- |
| **ID: REQ-01** | |
| **Nome** | Movimento Giocatore |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | L'utente puo' usare dei comandi di tastiera o joystick per muovere il personaggio. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-02** | |
| **Nome** | Interfaccia Grafica |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | L'utente puo' navigare attraverso l'interfaccia di gioco usando tastiera o joystick. |
| **Sotto requisiti** | |
| **001** | Interfaccia UI per gestire tutte le impostazioni (es: audio, grafica, account, ...). |

|  |  |
| --- | --- |
| **ID: REQ-03** | |
| **Nome** | Ambiente Partita |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Deve essere presente una mappa di gioco. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-04** | |
| **Nome** | Sito Web |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Deve essere presente un sito web. |
| **Sotto requisiti** | |
| **001** | **L’utente crea un account** |

|  |  |
| --- | --- |
| **ID: REQ-05** | |
| **Nome** | Interazioni Giocatore |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | L'utente puo' mirare e puntare un laser ed eventualmente parare i laser degli altri usando i comandi di tastiera o joystick. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-06** | |
| **Nome** | Ostacoli Casuali |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | I muri si generano in maniera casuale all'inizio di ogni partita usando l'algoritmo di Manhattan Mapper. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-07** | |
| **Nome** | HUD |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | L'Heads-Up Display (HUD) mostrerà varie informazioni durante la partita. |
| **Sotto requisiti** | |
| **001** | **Tempo rimasto di partita.** |
| **002** | **Punteggio** |

|  |  |
| --- | --- |
| **ID: REQ-08** | |
| **Nome** | AI Nemico |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Un'AI che si comporta come un giocatore. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-09** | |
| **Nome** | Database |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Il database conterrà tutte le tabelle utili per lo storage di dati e per il multiplayer. |
| **Sotto requisiti** | |
| **001** | Tabella Leaderboard Globale: sarà una tabella consultabile dal sito web dove verranno mostrati i migliori giocatori di sempre con il punteggio. |
| **002** | Tabella Leaderboard della Partita: conterrà i punteggi e i giocatori della partita corrente. |

|  |  |
| --- | --- |
| **ID: REQ-10** | |
| **Nome** | Multiplayer |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Il giocatore puo' collegarsi ad una sessione con altri giocatori. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-11** | |
| **Nome** | Audio |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Musica e Effetti Sonori del gioco. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-12** | |
| **Nome** | Modelli ed Animazioni |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | I modelli ed animazioni dei personaggi. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-13** | |
| **Nome** | Algoritmo di Cell-Shading |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | La grafica del gioco e' da cartone animata però in 3D. |
| **Sotto requisiti** | |

|  |  |
| --- | --- |
| **ID: REQ-14** | |
| **Nome** | Video Clip |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** | Le parite vengono registrate e mandati al sito web per essere visualizzati. |
| **Sotto requisiti** | |

### Spiegazione elementi tabella dei requisiti:

**ID**: identificativo univoco del requisito

**Nome**: breve descrizione del requisito

**Priorità**: indica l’importanza di un requisito nell’insieme del progetto, definita assieme al committente. Ad esempio, poter disporre di report con colonne di colori diversi ha priorità minore rispetto al fatto di avere un database con gli elementi al suo interno. Solitamente si definiscono al massimo di 2-3 livelli di priorità.

**Versione**: indica la versione del requisito. Ogni modifica del requisito avrà una versione aggiornata.

Sulla documentazione apparirà solamente l’ultima versione, mentre le vecchie dovranno essere inserite nei diari.

**Note**: eventuali osservazioni importanti o riferimenti ad altri requisiti.

**Sotto requisiti**: elementi che compongono il requisito.

## Use case

I casi d’uso rappresentano l’interazione tra i vari attori e le funzionalità del prodotto.

## Pianificazione

Prima di stabilire una pianificazione bisogna avere almeno una vaga idea del modello di sviluppo che si intende adottare. In questa sezione bisognerà inserire il modello concettuale di sviluppo che si seguirà durante il progetto. Gli elementi di riferimento per una buona pianificazione derivano da una scomposizione top-down della problematica del progetto.

La pianificazione può essere rappresentata mediante un diagramma di Gantt:

Se si usano altri metodi di pianificazione (p.es. scrum), dovranno apparire in questo capitolo.

## Analisi dei mezzi

Elencare e descrivere i mezzi disponibili per la realizzazione del progetto. Ricordarsi di sempre descrivere nel dettaglio le versioni e il modello di riferimento.

### Software

SDK, librerie, tools utilizzati per la realizzazione del progetto e eventuali dipendenze.

### Hardware

Su quale piattaforma dovrà essere eseguito il prodotto? Che hardware particolare è coinvolto nel progetto? Che particolarità e limitazioni presenta? Che HW sarà disponibile durante lo sviluppo?

# Progettazione

Questo capitolo descrive esaustivamente come deve essere realizzato il prodotto fin nei suoi dettagli. Una buona progettazione permette all’esecutore di evitare fraintendimenti e imprecisioni nell’implementazione del prodotto.

## Design dell’architettura del sistema

Descrive:

* La struttura del programma/sistema lo schema di rete...
* Gli oggetti/moduli/componenti che lo compongono.
* I flussi di informazione in ingresso ed in uscita e le relative elaborazioni. Può utilizzare *diagrammi di flusso dei dati* (DFD).
* Eventuale sitemap

## Design dei dati e database

Descrizione delle strutture di dati utilizzate dal programma in base agli attributi e le relazioni degli oggetti in uso.

Schema E-R, schema logico e descrizione.

Se il diagramma E-R viene modificato, sulla doc dovrà apparire l’ultima versione, mentre le vecchie saranno sui diari.

## Design delle interfacce



Figura - Interfaccia iniziale

La prima interfaccia visualizzata dall’utente comprende:

* Un bottone “Play” che fa iniziare la partita
* Un bottone “Settings” che permette all’utente di modificare delle impostazioni
* Un bottone “How to play” che spiega agli utenti i tasti per poter giocare
* Un bottone “Exit” che permette all’utente di uscire dal gioco

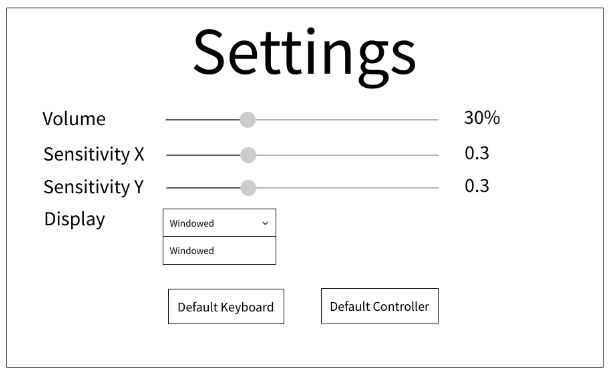


Figura - Interfaccia Impostazioni

La schermata di impostazioni viene visualizzata quando l’utente preme sul bottone “Settings”, comprende:

* Uno slider pel la regolazione del volume
* Uno slider pel la regolazione della sensibilità orizzontale
* Uno slider pel la regolazione della sensibilità verticale
* Un dropdown per la regolazione del display (a finestra o schermo intero)
* Un bottone che regola cambia gli slider ad un valore predefinito per l’uso della tastiera
* Un bottone che regola cambia gli slider ad un valore predefinito per l’uso della tastiera

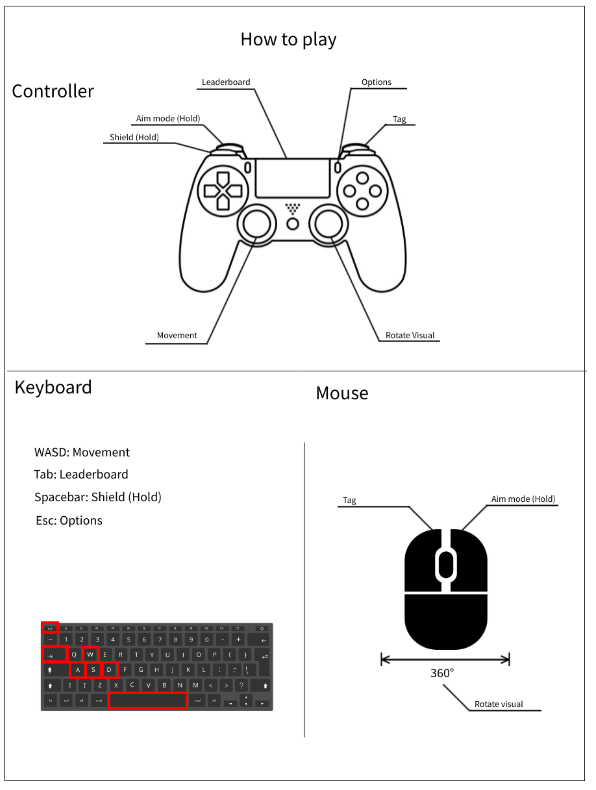


Figura - Mappatura tasti (controller e tastiera)

La schermata How to play viene visualizzata quando l’utente preme sul bottone “How to play”, questa schermata permette all’utente di poter visualizzare i tasti per giocare (sia controller che tastiera).

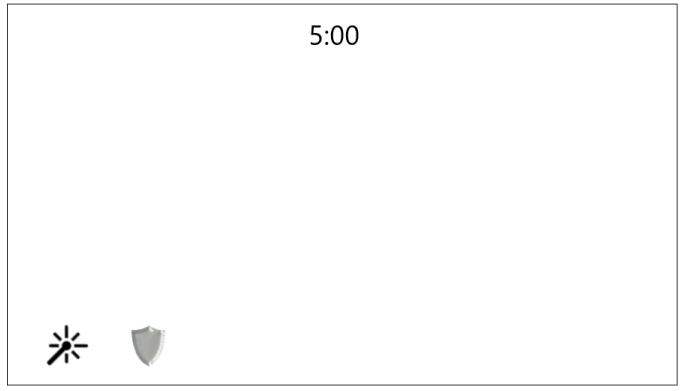


Figura – Interfaccia base di gioco

Questa interfaccia appare quando il giocatore preme il bottone “Play”. Vengono visualizzati la possibilità di utilizzare il puntatore laser e lo scudo tramite due tasti del controller/tastiera, inoltre il player vede l’ambiente di gioco. Il tempo in alto è la durata della partita



Figura - Interfaccia puntatore attivo

Questa interfaccia appare quando il giocatore sta utilizzando il puntatore laser impedendo di utilizzare lo scudo, inoltre il player vede l’ambiente di gioco. Il tempo in alto è la durata della partita

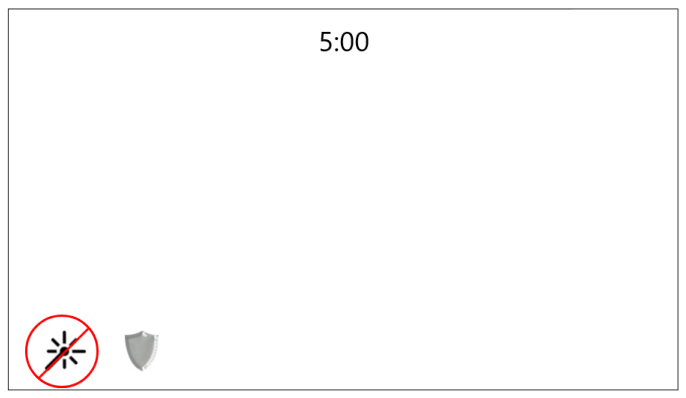


Figura - Interfaccia scudo attivo

Questa interfaccia appare quando il giocatore sta utilizzando lo scudo impedendo di utilizzare il puntatore laser, inoltre il player vede l’ambiente di gioco. Il tempo in alto è la durata della partita

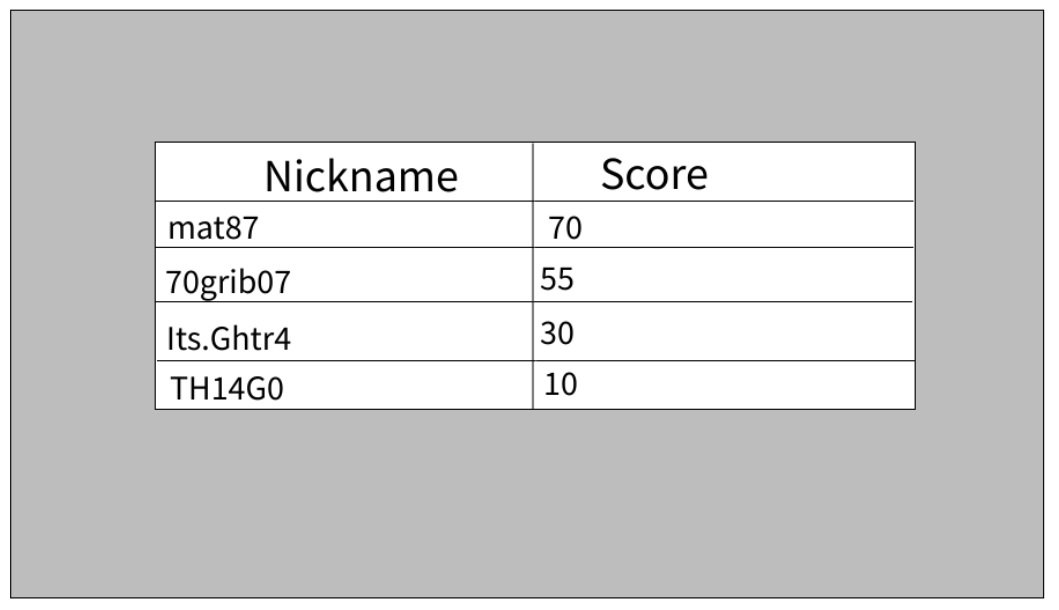


Figura - Interfaccia classifica

Questa schermata appare durante la partita quando il player schiaccia su “Tab”, viene visualizzata la classifica della partita. Inoltre il background diventa grigio.

## Design procedurale

Descrive i concetti dettagliati dell’architettura/sviluppo utilizzando ad esempio:

* Diagrammi di flusso e Nassi.
* Tabelle.
* Classi e metodi.
* Tabelle di routing
* Diritti di accesso a condivisioni …

Questi documenti permetteranno di rappresentare i dettagli procedurali per la realizzazione del prodotto.

# Implementazione

In questo capitolo dovrà essere mostrato come è stato realizzato il lavoro. Questa parte può differenziarsi dalla progettazione in quanto il risultato ottenuto non per forza può essere come era stato progettato.

Sulla base di queste informazioni il lavoro svolto dovrà essere riproducibile.

In questa parte è richiesto l’inserimento di codice sorgente - Print Screen - di maschere solamente per quei passaggi particolarmente significativi e/o critici.

Inoltre, dovranno essere descritte eventuali varianti di soluzione o scelte di prodotti con motivazione delle scelte.

Non deve apparire nessuna forma di guida d’uso di librerie o di componenti utilizzati. Eventualmente questa va allegata.

Per eventuali dettagli si possono inserire riferimenti ai diari.

# Test

## Protocollo di test

Definire in modo accurato tutti i test che devono essere realizzati per garantire l’adempimento delle richieste formulate nei requisiti. I test fungono da garanzia di qualità del prodotto. Ogni test deve essere ripetibile alle stesse condizioni.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-01 | **Nome:** | Movimento Giocatore |
| **Descrizione:** | L'utente puo' usare dei comandi di tastiera o joystick per muovere il personaggio. | | |
| **Procedura:** | 1. L’utente apre il gioco 2. Inizia una partita 3. Si sposta con controller o tastiera | | |
| **Risultati attesi:** | L’utente è in grado di spostarsi in giro per la mappa sia con controller che con tastiera | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-002  REQ-02 | **Nome:** | Interfaccia grafica |
| **Descrizione:** | L'utente puo' navigare attraverso l'interfaccia di gioco usando tastiera o joystick. | | |
| **Procedura:** | 1. L’utente apre il gioco 2. Seleziona I vari oggetti della schermata | | |
| **Risultati attesi:** | L’utente è in grado di interagire con gi oggetti delle schermate sia con controller che con tastiera | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-003  REQ-03 | **Nome:** | Ambiente Partita |
| **Descrizione:** | Deve essere presente una mappa di gioco. | | |
| **Procedura:** | 1. L’utente apre il gioco 2. Preme sul bottone “Play” 3. Entra in partita | | |
| **Risultati attesi:** | L’utente visualizza la mappa di gioco con il personaggio | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-004  REQ-04 | **Nome:** | Sito web |
| **Descrizione:** | Deve essere presente un sito web | | |
| **Procedura:** | 1. L’utente apre un browser 2. Entra nel sito web | | |
| **Risultati attesi:** | L’utente visualizza il sito web | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-005  REQ-04 | **Nome:** | Creazione account |
| **Descrizione:** | L’utente crea un account sul sito web | | |
| **Procedura:** | 1. L’utente entra sul sito 2. Crea un account | | |
| **Risultati attesi:** | L’utente ha un account personale sul sito web | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-006  REQ-05 | **Nome:** | Interazioni giocatore |
| **Descrizione:** | L'utente puo' mirare e puntare un laser ed eventualmente parare i laser degli altri usando i comandi di tastiera o joystick. | | |
| **Procedura:** | 1. L’utente avvia una partita 2. Mira e punta un avversario 3. Si para con lo scudp | | |
| **Risultati attesi:** | L’utente è in grado di utilizzare tutte le mosse del personaggio | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-007  REQ-06 | **Nome:** | Ostacoli casuali |
| **Descrizione:** | I muri si generano in maniera casuale all'inizio di ogni partita usando l'algoritmo di Manhattan Mapper. | | |
| **Procedura:** | 1. L’utente avvia una partita 2. Visualizza la posizione casual dei muri | | |
| **Risultati attesi:** | L’utente può giocare su delle mappe tutte diverse | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-008  REQ-07 | **Nome:** | HUD |
| **Descrizione:** | L'Heads-Up Display (HUD) mostrerà varie informazioni durante la partita. | | |
| **Procedura:** | 1. L’utente avvia una partita 2. L’utente guard ail tempo scorrere | | |
| **Risultati attesi:** | L’utente può giocare per un tot di tempo una determinate partita | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-009  REQ-07 | **Nome:** | Incremento punteggio leaderboard |
| **Descrizione:** | Ad ogni colpo del puntatore laser su un avversario il punteggio aumenta | | |
| **Procedura:** | 1. L’utente avvia una partita 2. Punta un avversario 3. Guarda nella leaderboard il punteggio | | |
| **Risultati attesi:** | L’utente può visualizzare il suo punteggio incrementato ad ogni colpo corretto | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-010  REQ-08 | **Nome:** | AI Nemico |
| **Descrizione:** | Un'AI che si comporta come un giocatore. | | |
| **Procedura:** | 1. L’utente avvia una partita 2. L’utente gioca senza altri 3 giocatori 3. Vengono generate le AI che giocano come una persona | | |
| **Risultati attesi:** | L’utente può giocare anche con solamente AI | | |

## Risultati test

Tabella riassuntiva in cui si inseriscono i test riusciti e non del prodotto finale. Se un test non riesce e viene corretto l’errore, questo dovrà risultare nel documento finale come riuscito (la procedura della correzione apparirà nel diario), altrimenti dovrà essere descritto l’errore con eventuali ipotesi di correzione.

## Mancanze/limitazioni conosciute

Descrizione con motivazione di eventuali elementi mancanti o non completamente implementati, al di fuori dei test case. Non devono essere riportati gli errori e i problemi riscontrati e poi risolti durante il progetto.

# Consuntivo

Consuntivo del tempo di lavoro effettivo e considerazioni riguardo le differenze rispetto alla pianificazione (cap. 1.7) (ad esempio Gantt consuntivo).

# Conclusioni

Quali sono le implicazioni della mia soluzione? Che impatto avrà? Cambierà il mondo? È un successo importante? È solo un’aggiunta marginale o è semplicemente servita per scoprire che questo percorso è stato una perdita di tempo? I risultati ottenuti sono generali, facilmente generalizzabili o sono specifici di un caso particolare? ecc.

## Sviluppi futuri

Migliorie o estensioni che possono essere sviluppate sul prodotto.

## Considerazioni personali

Cosa ho imparato in questo progetto? ecc.

# Glossario

Inserite una semplice tabella con due colonne che spieghi i termini specifici del progetto (lista dei termini in ordine alfabetico A-Z)

Esempio:

|  |  |
| --- | --- |
| **Termine** | **Descrizione** |
| AJAX | **Asynchronous JavaScript And XML**: una tecnica che permette di eseguire richieste ed ottenere dati da una pagina web in modo asincrono. |
| CSS | **Cascading Style Sheets**: linguaggio che permette di definire il layout e la grafica di una pagina web. |

# Bibliografia

## Bibliografia per articoli di riviste:

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo dell’articolo (tra virgolette),
3. Titolo della rivista (in italico),
4. Anno e numero
5. Pagina iniziale dell’articolo.

## Bibliografia per libri

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo del libro (in italico),
3. ev. Numero di edizione,
4. Nome dell’editore,
5. Anno di pubblicazione,
6. ISBN.

## Sitografia

<https://wireframepro.mockflow.com/> 03.02.2023

# Allegati

Elenco degli allegati, esempio:

* Diari di lavoro
* Codici sorgente/documentazione macchine virtuali
* Istruzioni di installazione del prodotto (con credenziali di accesso) e/o di eventuali prodotti terzi
* Documentazione di prodotti di terzi
* Eventuali guide utente / Manuali di utilizzo
* Mandato e/o QdC
* Prodotto
* …