Beat The Piano

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: 7

2.3 Use case 8

2.4 Pianificazione 9

2.5 Analisi dei mezzi 9

2.5.1 Software 9

2.5.2 Hardware 9

3 Progettazione 10

3.1 Design procedurale 10

3.1.1 Swimlane Main Menu 10

3.1.2 Swimlane Impostazioni 11

3.1.3 Swimlane Credits 11

3.1.4 Swimlane Gioco 12

3.2 Progettazione strutturale 13

3.2.1 Diagramma delle classi 13

3.3 Design delle interfacce 14

3.3.1 Interfaccia Main Menu 14

3.3.2 Interfaccia Impostazioni 14

3.3.3 Interfaccia Credits 15

3.3.4 Interfaccia Scelta Livello 15

3.3.5 Interfaccia Gioco 16

3.3.6 Interfaccia Risultati 16

4 Implementazione 17

5 Test 17

5.1 Protocollo di test 17

5.2 Risultati test 18

5.3 Mancanze/limitazioni conosciute 18

6 Consuntivo 18

7 Conclusioni 18

7.1 Sviluppi futuri 18

7.2 Considerazioni personali 18

8 Glossario 18

9 Bibliografia 19

9.1 Bibliografia per articoli di riviste: 19

9.2 Bibliografia per libri 19

9.3 Sitografia 19

10 Indice delle figure 19

11 Allegati 19

# Introduzione

## Informazioni sul progetto

**• Allievo:** Ivan Ostroumov

**• Docente:** Geo Petrini

**• Scuola e classe:** SAMT Sezione informatica, I3BD

**• Materia:** Modulo 306

**• Data inizio consegna:** 05.09.2025 – 29.12.2025

## Abstract

## Scopo

Il progetto ha l’obiettivo principale di sviluppare un videogioco musicale educativo, interattivo, divertente e coinvolgente. Attraverso l’uso di un’interfaccia digitale, il gioco permette agli utenti di esercitarsi nello sviluppare il senso del ritmo e migliorare la coordinazione tra mani e occhi, simulando l’esperienza reale di suonare uno strumento, o semplicemente divertirsi.

Dal punto di vista didattico, serve a me per imparare a usare il game engine Unity, espandere le mie conoscenze sul design grafico e esercitarsi col linguaggio di programmazione C#.

Tra l’altro mi aiuterà a imparare a gestire un progetto usando le diverse tecniche fornite come il waterfall, i diversi diagrammi, eccetera.

# Analisi

## Analisi del dominio

“Beat The Piano” è un software interattivo che unisce il concetto dei video MIDI al gameplay dei rhythm game. L’utente deve premere i tasti corretti sulla tastiera del computer nel momento in cui le note, che cadono dall’alto, raggiungono i tasti del pianoforte virtuale.

Il programma è pensato per tutti gli utenti che vogliono imparare il ritmo o semplicemente divertirsi, senza dover possedere una tastiera MIDI o conoscenze musicali. Ogni nota del pianoforte viene associata a un tasto della tastiera del PC (esempio: C# = 2, D = W), rendendo il gioco intuitivo e accessibile.

Attualmente esistono software simili come “Synthesia” o “Piano From Above”, ma richiedono l’uso di una tastiera MIDI e non prevedono una mappatura diretta con la tastiera del computer. “Beat The Piano” si distingue proprio per questo: offre un’esperienza completamente giocabile solo con la tastiera del PC, combinando apprendimento musicale e divertimento.

In più esistono giochi da telefono come per esempio “Piano Tales” che ha più o meno lo stesso funzionamento ma è limitato da 4 tasti che tra l’altro sono digitali, ovvero sullo schermo e non su una tastiera. Quindi anche in questo caso “Beat The Paino”, che copre 2 ottavi (24 tasti), offre una esperienza molto più avvicinata al suonare un vero pianoforte.

## Analisi e specifica dei requisiti

Qui sotto sono riportati i requisiti del progetto “Beat The Piano”.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-01 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica | | | | |
| **Note** | L’utente deve poter visualizzare tutte le funzionalità e pagine dell’applicativo. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve un bottone per andare nella pagina di gioco. | | | | |
| **002** | Serve un bottone per andare nella pagina di scelta di livello. | | | | |
| **003** | Serve un bottone per andare nella pagina delle impostazioni. | | | | |
| **004** | Serve un bottone per andare nella pagina dei credits. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-02 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica Impostazioni | | | | |
| **Note** | L’utente deve poter visualizzare tutte le impostazioni che può cambiare. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve un bottone per andare nella pagina principale. | | | | |
| **002** | Serve un volume slider per impostare il volume. | | | | |
| **003** | Serve un color picker per impostare il colore delle note. | | | | |
| **004** | Serve un bottone per cancellare i salvataggi. | | | | |
| **005** | Serve un’immagine di pianoforte del gioco con ogni tasto cliccabile, dopo il click si assegna il tasto rispettivo sulla tastiera. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-03 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica Credits | | | | |
| **Note** | L’utente deve poter visualizzare chi ha attribuito alla creazione del gioco. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve poter vedere chi ha creato le diverse parti del gioco. | | | | |
| **002** | Serve un bottone per andare nella pagina principale. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-04 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica Scelta livello | | | | |
| **Note** | L’utente deve poter visualizzare i diversi livelli che può giocare. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve un bottone per andare nella pagina principale. | | | | |
| **002** | Serve un bottone per ogni livello presente per giocarlo. | | | | |
| **003** | Serve uno sprite di lucchetto per ogni livello non ancora sbloccato. | | | | |
| **004** | Serve una scritta con il nome per ogni livello. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-05 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica Gioco | | | | |
| **Note** | L’utente deve poter visualizzare il processo di gioco. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve un background. | | | | |
| **002** | Serve uno sprite di pianoforte con su ogni tasto il tasto associato da schiacciare. | | | | |
| **003** | Servono blocchi cadenti per far capire quando e quale nota suonare. | | | | |
| **004** | Serve animare le due scritte in modo da rendere tutto più dinamico | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-06 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Interfaccia Grafica Risultati | | | | |
| **Note** | L’utente deve poter visualizzare il suo risultato di gioco | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve tre stelle che simboleggiano il risultato. | | | | |
| **002** | Serve un bottone per tornare nella pagina principale. | | | | |
| **003** | Serve un bottone per riprovare il livello. | | | | |
| **004** | Serve un bottone per andare al prossimo livello. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-07 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Classe Nota | | | | |
| **Note** | Ci deve essere la classe che comporrà le diverse melodie. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve una caratteristica spawnTime. | | | | |
| **002** | Serve una caratteristica duration. | | | | |
| **003** | Serve una caratteristica nota. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-08 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Classe Music | | | | |
| **Note** | Ci deve essere la classe su qui sono basati i livelli. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve una caratteristica name. | | | | |
| **002** | Serve una caratteristica difficulty. | | | | |
| **003** | Serve una caratteristica notes | | | | |
| **004** | Serve una caratteristica tempo | | | | |
| **005** | Serve un metodo calcolaPunteggio che confronta due Music. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-09 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Script Interpretator | | | | |
| **Note** | Ci deve essere lo script che ha la funzione Play creando i blocchi cadenti | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve un algoritmo che crea i blocchi leggendo le caratteristiche di Music. | | | | |
| **002** | Serve un algoritmo che cancella i blocchi leggendo le caratteristiche di Music. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-10 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Script playerController | | | | |
| **Note** | Ci deve essere uno script che gestisce i controlli dell’utente | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve una keymap che salva le relazioni tra tasti di pianoforte e tastiera. | | | | |
| **002** | Serve una funzione per cambiare i tasti associati. | | | | |
| **003** | Serve una funzione per suonare la nota schiacciata. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-11 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Livelli | | | | |
| **Note** | Ci devono essere i livelli (tutti per solo 2 ottavi) | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve trovare due midi facili. | | | | |
| **002** | Serve trovare due midi medi. | | | | |
| **003** | Serve trovare due midi difficili. | | | | |
| **004** | Serve trovare un midi difficilissimo. | | | | |
| **005** | Bisogna convertirli tutti nelle classi del programma. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-12 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Salvataggi | | | | |
| **Note** | Deve essere possibile salvare il progresso dell’utente. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve uno script per salvare tutti i dati. | | | | |
| **002** | Serve uno script per caricare tutti i dati salvati. | | | | |
| **003** | Serve uno script per aggiungere la cifratura in mezzo a salvare e caricare. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requisito** | Req-13 | **Priorità** | 1 | **Versione** | 1.0 |
| **Nome** | Suoni | | | | |
| **Note** | Ci devono essere i suoni nel gioco. | | | | |
| **Sotto requisiti** | | | | | |
| **001** | Serve uno suono per lo schiacciamento dei bottoni | | | | |
| **002** | Servono i suoni per le note. | | | | |
| **003** | Serve una musica di sottofondo nei menu. | | | | |
| **004** | Serve uno suono dei risultati. | | | | |

### 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

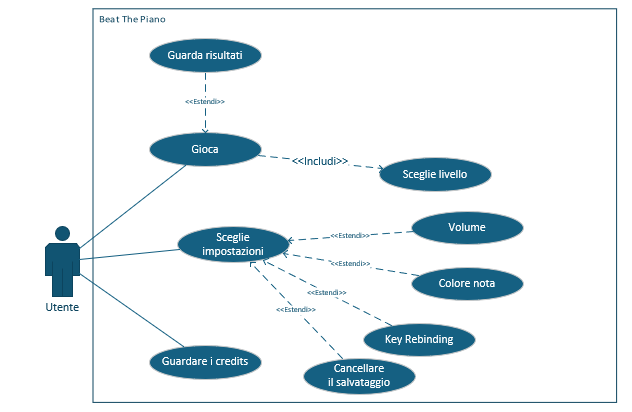


Figura 1 - Use Case

Questo è i use case dell’applicativo “Beat The Piano” che descrive il funzionamento dell’applicazione. C’è solo un attore presente:

**Utente:**

L’utente può svolgere principalmente 3 azioni:

La prima: può giocare ai livelli, scegliendolo prima, ed eventualmente guardare i risultanti ottenuti.

La seconda: può scegliere e cambiare le impostazioni di: “Volume”, “Colore nota”, “Key Rebinding” 🡪 cambiare quale tasto della tastiera è associato a quale nota del pianoforte, “Cancellare il salvataggio”.

La terza: In più l’utente può guardare i credits dove può vedere chi ha lavorato su che aspetto del gioco.

## 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:

|  |
| --- |
| Figura 2: Esempio di diagramma di Gantt. |

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

## Analisi dei mezzi

Per la realizzazione del progetto “Beat The Piano” ho usato i computer della scuola (Centro Professionale Tecnico Lugano-Trevano), il software già installato e diversi siti web.

### Software

Unity 6.1

C# 14

Photopea 5.6

Mockflow

Visual Studio 2022 17.14

Rider 2025.2.3

### Hardware

Computer con Windows 11, mouse, tastiera, monitor, SSD.

Caratteristiche pc:

Processore 13th Gen Intel(R) Core(TM) i7-13700 2.10 GHz

RAM installata 32.0 GB 4800MHz

Tipo sistema Sistema operativo Windows 11 a 64 bit, processore basato su x64

GPU NVIDIA T400 4GB

# Progettazione

## Design procedurale

Insomma ci sono 5 diagrammi di flusso tra qui: uno di impostazioni, uno del main menu, uno del gioco, uno dei credits e uno con tutto messo insieme.

Servono per chiarire come deve funzionare l’applicazione, in modo che sarà più semplice svilupparla.

### Swimlane Main Menu

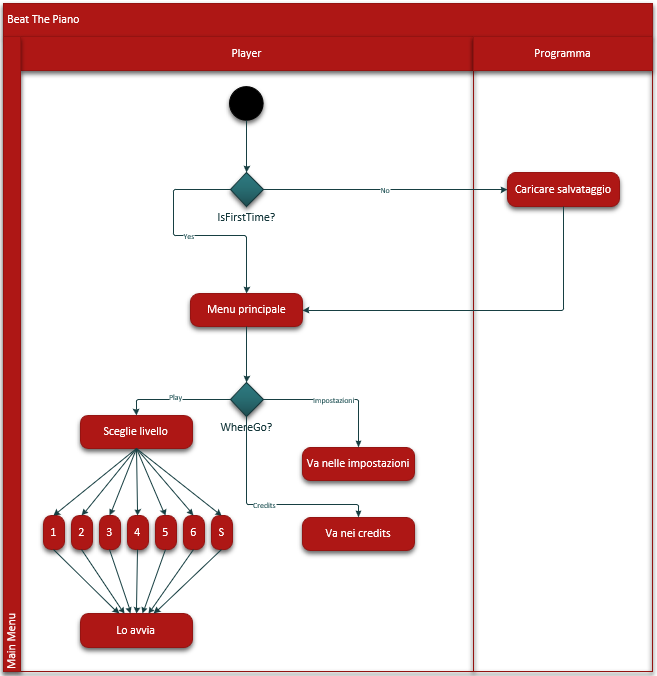


Figura 3 - Swimlane Main Menu

### Swimlane Impostazioni

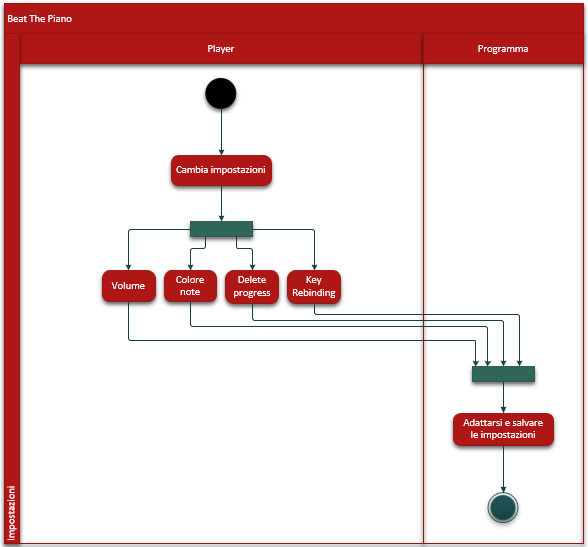


Figura 4 - Swimlanes Impostazioni

### Swimlane Credits

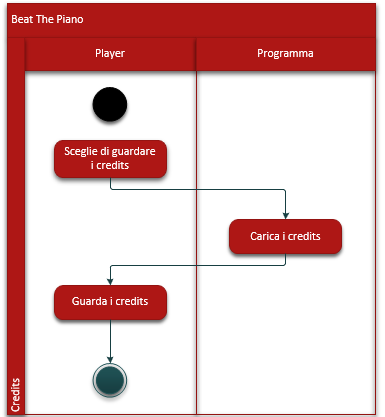


Figura 5 - Swimlanes Credits

### Swimlane Gioco

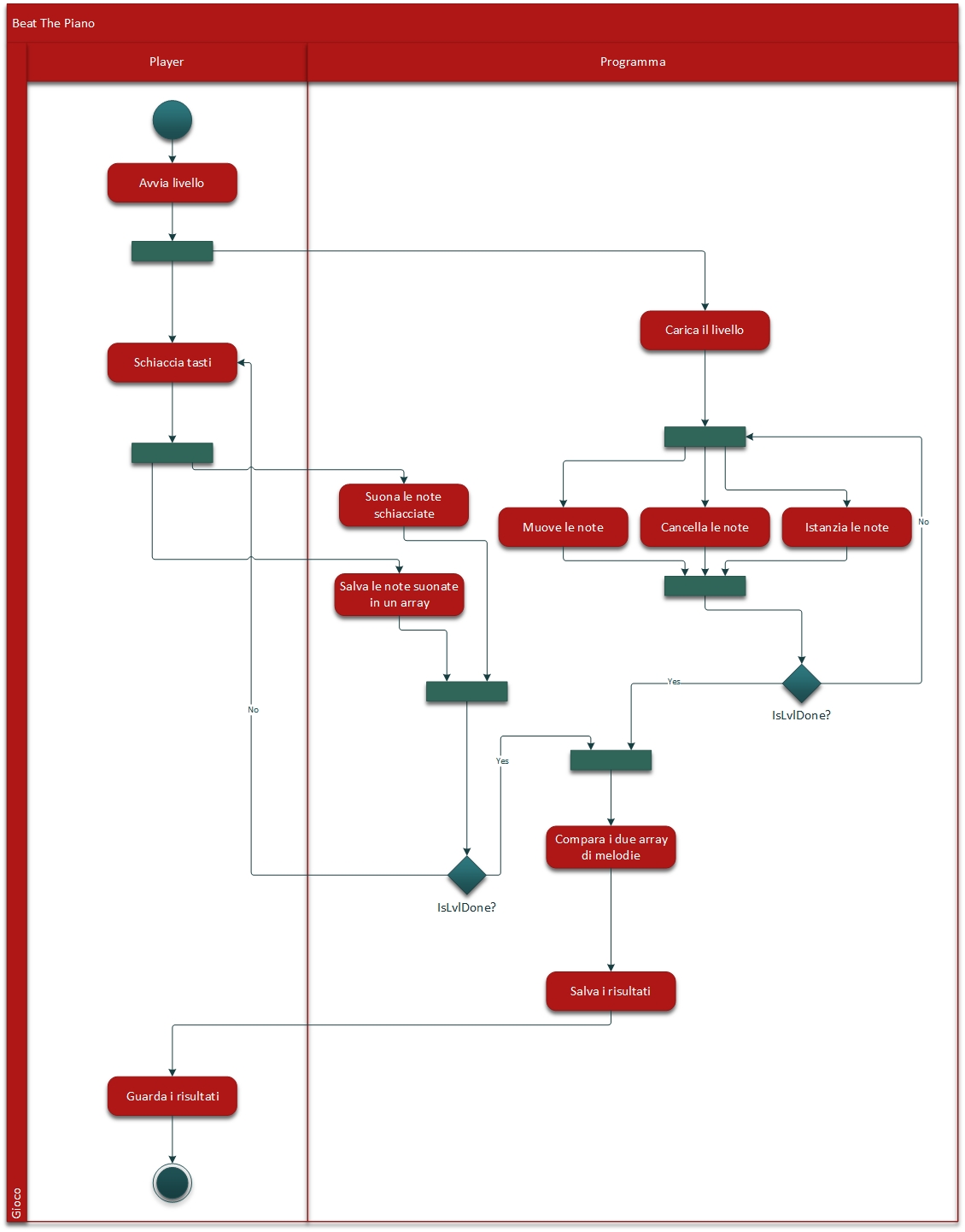


Figura 6 - Swimlane Gioco

## Progettazione strutturale

### Diagramma delle classi

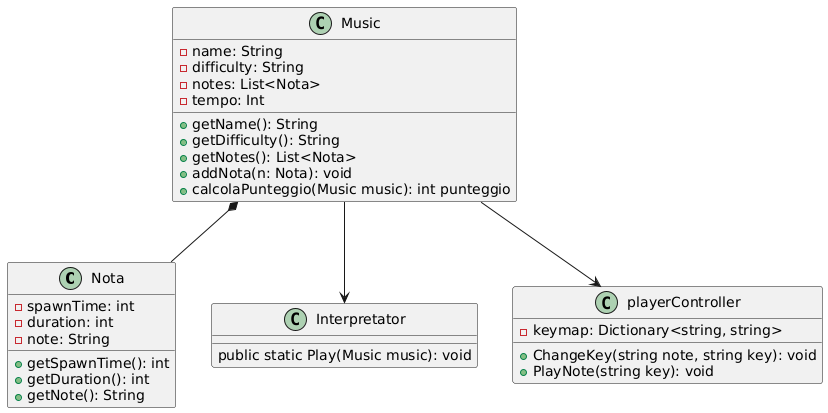


Figura 7 - Diagramma delle classi

Questo è il diagramma che descrive che classi e script ci saranno nel progetto, specificando che variabili e funzioni implementeranno. Ci sarà la classe nota che avrà spawnTime, duration e note che sono rispettivamente i: momento quando spawnare la nota, la lunghezza della nota (quanto l’utente la deve tenere) e note che specifica la nota stessa.

Poi c’è la classe Music che ha le caratteristiche name, difficulty, notes e tempo che sono rispettivamente: il nome, la difficoltà, la lista di note per comporre il livello e il tempo che È la velocità del livello.

Dopo viene lo script Interpretator che prendera un oggetto Music e lo usa per creare il livello.

Infine, c’ è lo script playerController che ha la keymap, ovvero la mappatura tra i tasti della tastiera e del pianoforte. Inoltre, questo script permette di cambiare le associazioni tra tasto e nota. Come ultimo, permette di suonare la nota / le note che l’utente sta premendo in quel momento.

## Design delle interfacce

Il design delle interfacce è stato fatto basandosi ai requisiti. Sono state create usando il sito web mockflow.

### Interfaccia Main Menu



Figura 8 - Interfaccia Main Menu

Questa è la pagina iniziale con qui si avvierà l’applicativo, si può vedere il titolo del programma e 4 bottoni tramite quali si potrà accedere alle altre pagine del software ovvero:

* Impostazioni col bottone in alto a sinistra
* Credits col bottone in basso destra
* Scelta livelli tramite il bottone “Levels”
* Andare all’ultimo livello sbloccato col bottone “Play”

### Interfaccia Impostazioni



Figura 9 - Interfaccia Impostazioni

Questa è la pagina delle impostazioni dove appunto, si possono variare certe impostazioni:

* Volume con la scrollbar in alto
* Il colore delle note col color picker
* Cancellare i salvataggi con il cestino
* Fare rebinding di tasti e note cliccando il tasto sul pianoforte e poi schiacciando il tasto sulla tastiera

Inoltre, in alto a sinistra c’è la freccia per tornare nel menu principale.

### Interfaccia Credits



Figura 10 - Interfaccia Credits

Questa è la pagina dei Credits dove ci saranno scritte le info su chi ha fatto quale parte del gioco tra cui lo sviluppatore, il progettista e tutti gli autori delle melodie.

### Interfaccia Scelta Livello

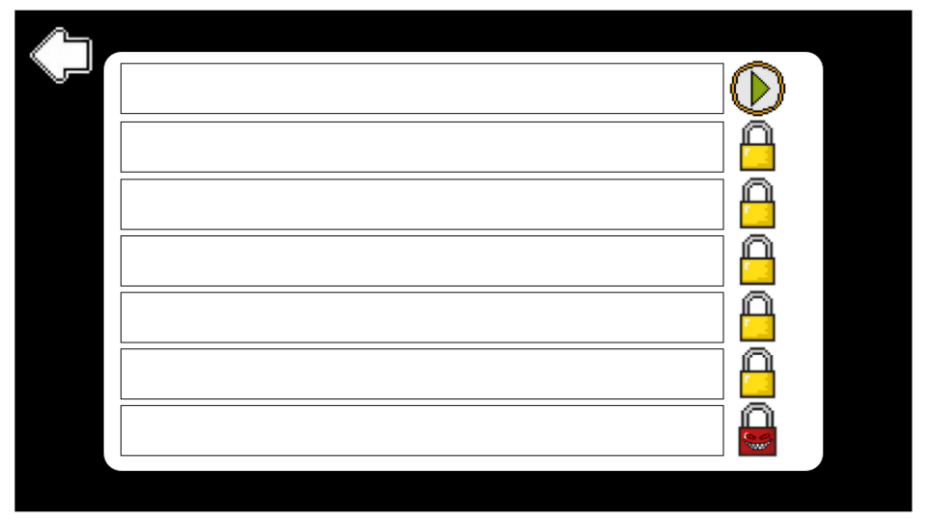


Figura 11 - Interfaccia Scelta Livelli

Questa è la pagina dove si sceglie il livello da giocare, in alto a sinistra c’è la freccia per tornare indietro nel menu principale. Sull’illustrazione è il design al momento dell’inizio del gioco quando È sbloccato solo il primo livello. Dentro i rettangoli ci saranno i nomi dei livelli.

### Interfaccia Gioco

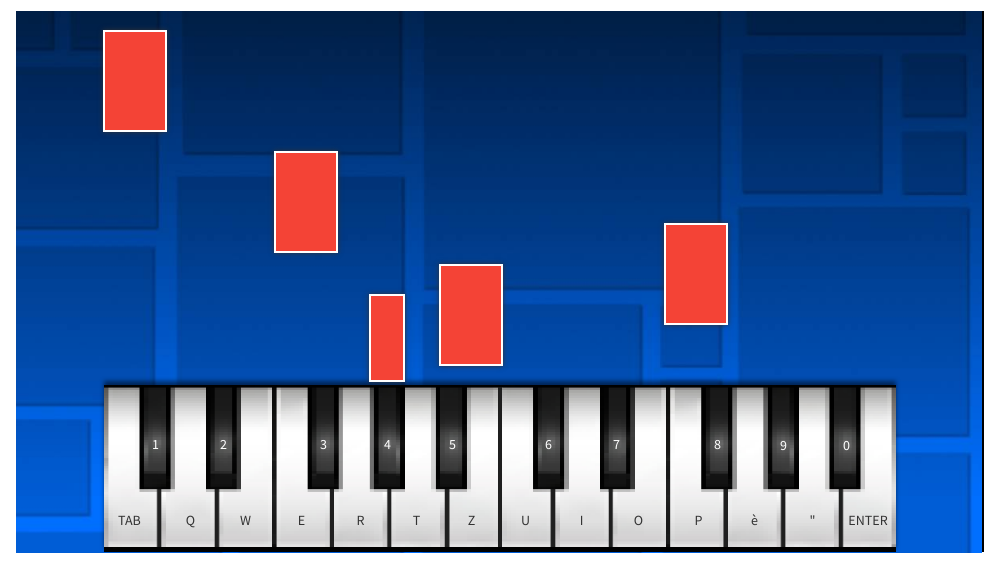


Figura 12 - Interfaccia Gioco

Questa è la pagina di gioco, dove l’utente dovrà schiacciare i tasti associati al pianoforte (si può vedere dalla sprite del piano forte che tasto schiacciare). Da sopra cadranno i blocchi che simboleggiano le note da schiacciare.

### Interfaccia Risultati



Figura 13 - Interfaccia Risultati

Questa è la pagina che rappresenta i risultati, sarà mostrata sopra la pagina di gioco alla fine. Si può vedere il proprio risultato che è illustrato dalle tre stelle (più stelle piene – meglio è). Ci sono anche 3 bottoni:

* Sinistro: riprovare il livello.
* Centrale: andare al prossimo livello.
* Destro: ritornare al menu principale.

# 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-012 | **Nome:** | Import a card with KIC, KID and KIK keys, but not shown with the GUI |
| **Descrizione:** | Import a card with KIC, KID and KIK keys with no obfuscation, but not shown with the GUI | | |
| **Prerequisiti:** | Store on local PC: Profile\_1.2.001.xml (appendix n\_n) and Cards\_1.2.001.txt (appendix n\_n).  PIN (OTA\_VIEW\_PIN\_PUK\_KEY) and ADM (OTA\_VIEW\_ADM\_KEY) user right not set. | | |
| **Procedura:** | 1. Go to “Cards manager” menu,  in main page click “Import Profiles” link, Select the “1.2.001.xml” file, Import the Profile 2. Go to “Cards manager” menu,  in main page click “Import Cards” link, Select the “1.2.001.txt” file, Delete the cards,  Select the “1.2.001.txt” file, Import the cards 3. Research the “41795924770” Card, Click the imsi card link Check the card details 4. Execute the SQL: SELECT imsi, dir, keyset, cntr, rawtohex(kickey), rawtohex(kidkey), rawtohex(kikkey), rawtohex(chv), rawtohex(dap)FROM otacardkey a where imsi='340041795924770' ORDER BY keyset; | | |
| **Risultati attesi:** | Keys visible in the DB (OtaCardKey) but not visible in the GUI (Card details) | | |

## 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

1. URL del sito (se troppo lungo solo dominio, evt completo nel diario),
2. Eventuale titolo della pagina (in italico),
3. Data di consultazione (GG-MM-AAAA).

**Esempio:**

* http://standards.ieee.org/guides/style/section7.html, *IEEE Standards Style Manual*, 07-06-2008.

# Indice delle figure

[Figura 1 - Use Case 8](#_Toc211601092)

[Figura 2: Esempio di diagramma di Gantt. 9](#_Toc211601093)

[Figura 3 - Swimlane Main Menu 10](#_Toc211601094)

[Figura 4 - Swimlanes Impostazioni 11](#_Toc211601095)

[Figura 5 - Swimlanes Credits 11](#_Toc211601096)

[Figura 6 - Swimlane Gioco 12](#_Toc211601097)

[Figura 7 - Interfaccia Main Menu 14](#_Toc211601098)

[Figura 8 - Interfaccia Impostazioni 14](#_Toc211601099)

[Figura 9 - Interfaccia Credits 15](#_Toc211601100)

[Figura 10 - Interfaccia Scelta Livelli 15](#_Toc211601101)

[Figura 11 - Interfaccia Gioco 16](#_Toc211601102)

[Figura 12 - Interfaccia Risultati 16](#_Toc211601103)

# 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