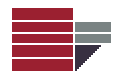
**Università della Calabria**

**Dipartimento di Matematica e Informatica**



**Corso di Laurea Triennale/Magistrale in Informatica**

Tesi di Laurea

**Applicazione per la gestione della dieta**  ​

|  |  |  |
| --- | --- | --- |
|  | Relatore | Candidato: |
|  | Prof. Carmine Dodaro | Bogdan Sorin Miauca |
|  |  | Matricola 189987 |

Anno Accademico 2020/2021

[1 Introduzione 2](#_Toc74058861)

[1.1 Cos’e l’applicazione per la gestione della dieta 2](#_Toc74058862)

[1.2 Motivazioni 2](#_Toc74058862)

[2 Tecnologie e strumenti utilizzati per lo sviluppo del progetto 2](#_Toc74058864)

[2.1 Linguaggio Java 2](#_Toc74058862)

[2.2 JavaFX e JavaFX Scene Builder 2](#_Toc74058863)

[2.3 SQLite 2](#_Toc74058862)

[2.4 CSS per JavaFx 2](#_Toc74058863)

[2.5 Pattern MVC e Singleton 2](#_Toc74058863)

[2.6 IText 5 2](#_Toc74058862)

[2.7 Metabolismo Bassale , Livello di Attivita Fisica e calcolo del Fabbisogno Giornaliero 2](#_Toc74058862)

[3 Descrizione del progetto realizzato 2](#_Toc74058865)

[3.1 Interfaccia 2](#_Toc74058862)

3.1.1 Scena 1

3.1.2 Scena 2

3.1.3 Scena 3

3.1.4 Scena 4

3.1.5 Scena 5

3.1.6 Scena 6

3.1.7 Scena 7

[3.2 Logica 2](#_Toc74058863)

[3.3 Data 2](#_Toc74058862)

[4 Conclusioni 2](#_Toc74058866)

[5 Bibliografia e sitografia 2](#_Toc74058867)

# Introduzione

## Cos’e l’applicazione per la gestione della dieta

L’applicazione per la gestione della dieta e un software con lo scopo di gestire le diete di varie persone , calcolare in base alle caratteristiche fisiche della persona le calorie giornaliere da consumare per raggiungere un certo obbiettivo e calcolare i macro nutrienti di cui il corpo ha bisogno.

L’applicazione utilizza strumenti dal mondo informatico per la progettazione ed implementazione e strumenti del mondo nutrizionale per il calcolo delle calorie e dei macronutrienti. Nel contesto informatico utilizza un’interfaccia grafica per interagirsi con l’utente finale , dove vengono inseriti i dati di cui l’applicazione ha bisogno per il calcolo delle calorie e dei macro nutrienti. L’interfaccia ha varie scene o finestre alcune con il compito di mostrare i risultati , altre di prendere le informazioni , come alimenti da inserire nella dieta , informazioni sulla persona nonché sulla sua struttura fisica e gli obbiettivi da ottenere.

Dietro l’interfaccia abbiamo un database in cui troviamo già salvati i vari alimenti con le loro proprietà ed in cui verranno salvate in seguito le persone a cui verrà fatta la dieta con le loro diete giorno per giorno.

Dietro l’interfaccia troviamo la logica che e composta da strutture dati ed algoritmi , insieme a calcoli e nozioni prese dal mondo della nutrizione. Grazie ad essa i dati vengono calcolati correttamente e salvati nel database.

Dal mondo della nutrizione invece abbiamo nozioni come metabolismo bassale che serve a dirci quante calorie vengono consumate in situazioni di inerzia del corpo e livello di attività fisica che ci dice quante calorie vengono consumate facendo una certa attività.

Le formule del calcolo di queste nozioni insieme agli algoritmi e alle strutture dati compongono la logica dell’applicazione.

Tra la logica e l’interfaccia troviamo il controller che serve a far interagire l’una con l’altra , possiamo vederlo come un ponte che collega la logica e l’interfaccia utente , chiamato anche pattern MVC.

L’applicazione implementa anche delle feature in grado di farci vedere con dei grafici come cambia la costituzione fisica di una persona nel tempo ed il suo fabbisogno giornaliero ed a stampare in formato PDF la sua dieta settimanale.

## Attività svolte e risultati ottenuti

Fornire una breve sintesi delle attività che sono state svolte durante il periodo di tesi e dei principali risultati ottenuti (applicazioni o sistemi sviluppati). (circa 1 pagina)

# Tecnologie e strumenti utilizzati per lo sviluppo del progetto

## Linguaggio Java

# Descrizzione :

# Java e un linguaggio di programmazione di alto livello orientato agli oggetti . Una delle cose che lo rendono uno tra i piu usati linguaggi di programmazione e il fatto che puo girare quasi su ogni tipo di dispositivo che abbia installato java . Infatti java non viene eseguito sul nostro dispositivo ma su una macchina virtuale , chiamata Java Virtual Machine per poi mandare il risultato alla nostra macchina. Il fatto che si una macchina virtuale invece che direttamente il nostro dispositivo ha un costo in termini di prestazione , ma un guadagno in termini di portabilita, cio’e il codice non va cambiato da dispositivo a dispositivo.

# Esempio di un algoritmo in java che calcola i numeri primi da 1 a 100.

# 

# Se facessimo partire questo algoritmo su qualsiasi dispositivo , ovviamente abbastanza veloce , funzionerebbe su ognuno di questi.

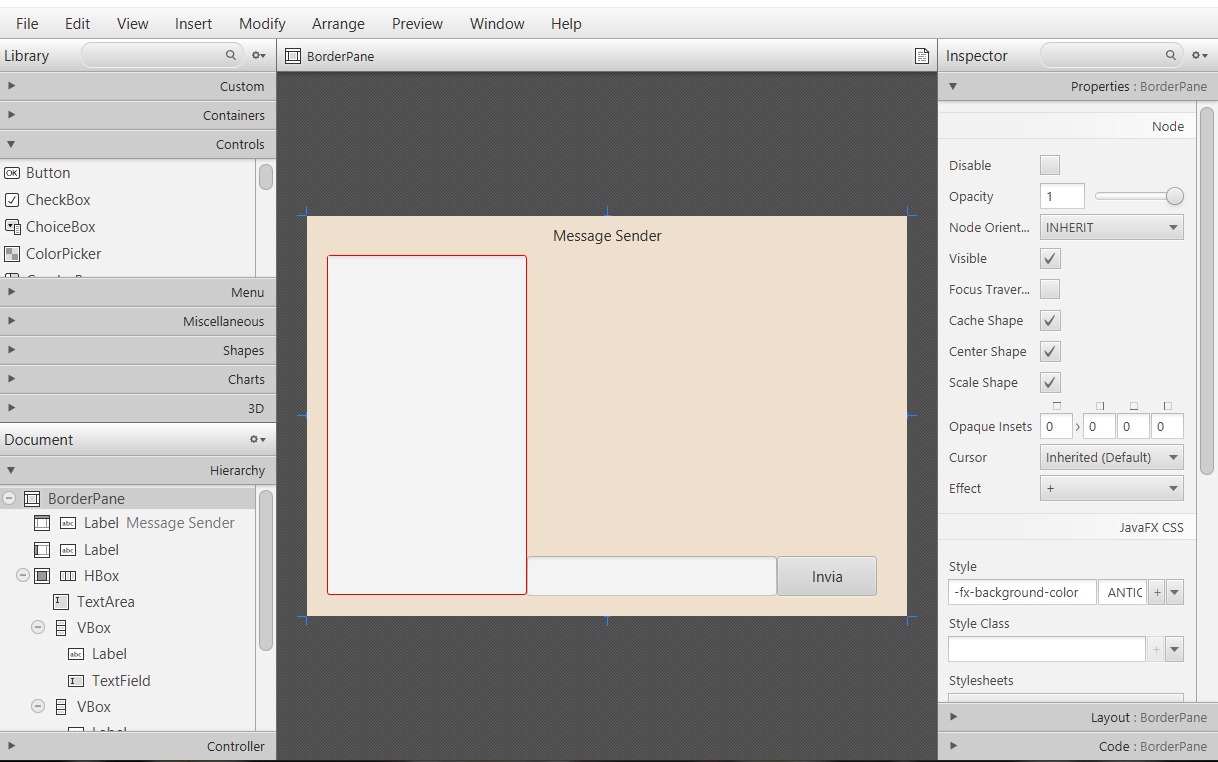
## JavaFX e JavaFX Scene Builder

# Descrizzione :

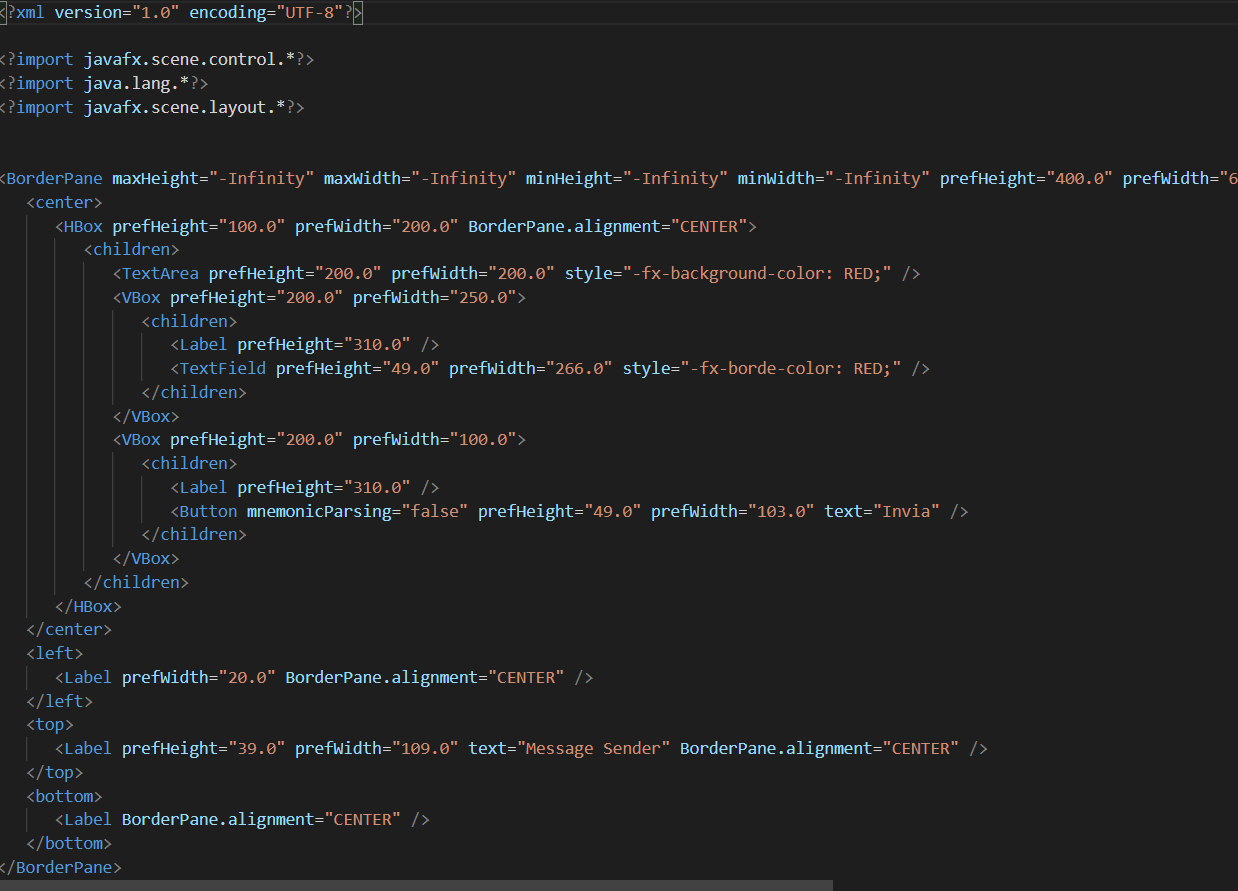
# come descritto in [1]:

“JavaFX è una famiglia di software applicativi, basato sulla piattaforma Java, per la creazione di applicazioni per computer. Con JavaFX è possibile realizzare delle applicazioni per computer, cellulari, dispositivi portatili di vario genere, televisori e altri tipi di piattaforme. JavaFX introduce un linguaggio di scripting dichiarativo ad oggetti estremamente pratico per lo sviluppo di applicazioni grafiche . Lo spazio grafico di lavoro di JavaFX è un oggetto Stage , radice di ogni applicazione JavaFX, che può intercambiare oggetti Scene. JavaFX Scene Builder e un software grafico per le applicazioni JavaFX che rende la creazione di un’interfaccia grafica facile e senza aver bisogno di scrivere codice , perche lo genera in automatico , e lo fa importare come documento FXML che puo essere usato insieme a Java per la creazione del progetto. In seguito viene mostrato come e un’interfaccia creata con JavaFx Scene Builder in formato FXML.

**Interfaccia grafica creata con JavaFx Scene Builder**



**Il file FXML generato da JavaFX Scene Builder**



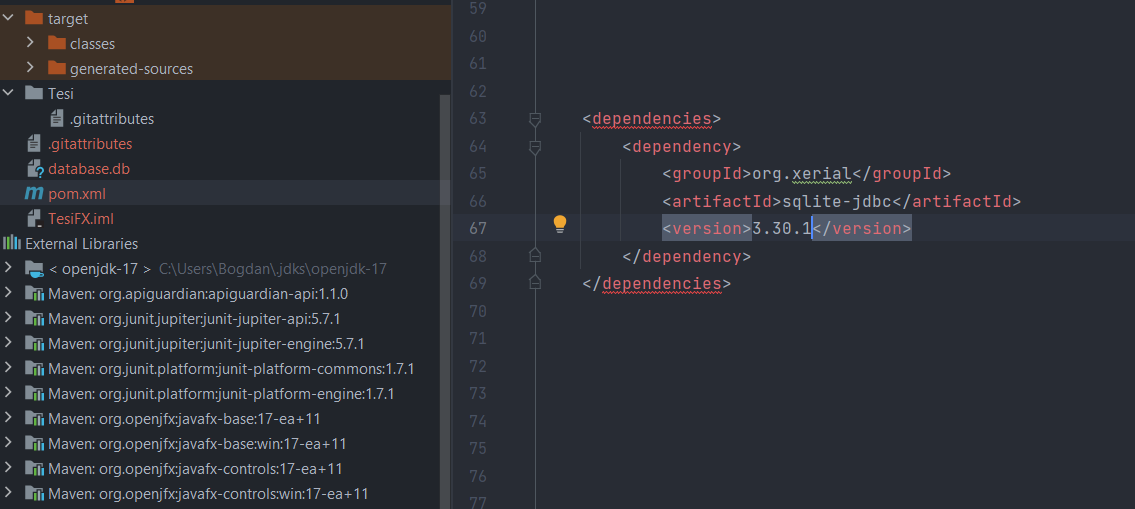
Le due immagini sono la stessa cosa , solo che una rappresenta ciò che vediamo e l’altra e la sua costruzione. Quindi JavaFx Scene Builder agevola la creazione dell’interfaccia grafica risparmiandoci di scrivere codice.

## SQLite

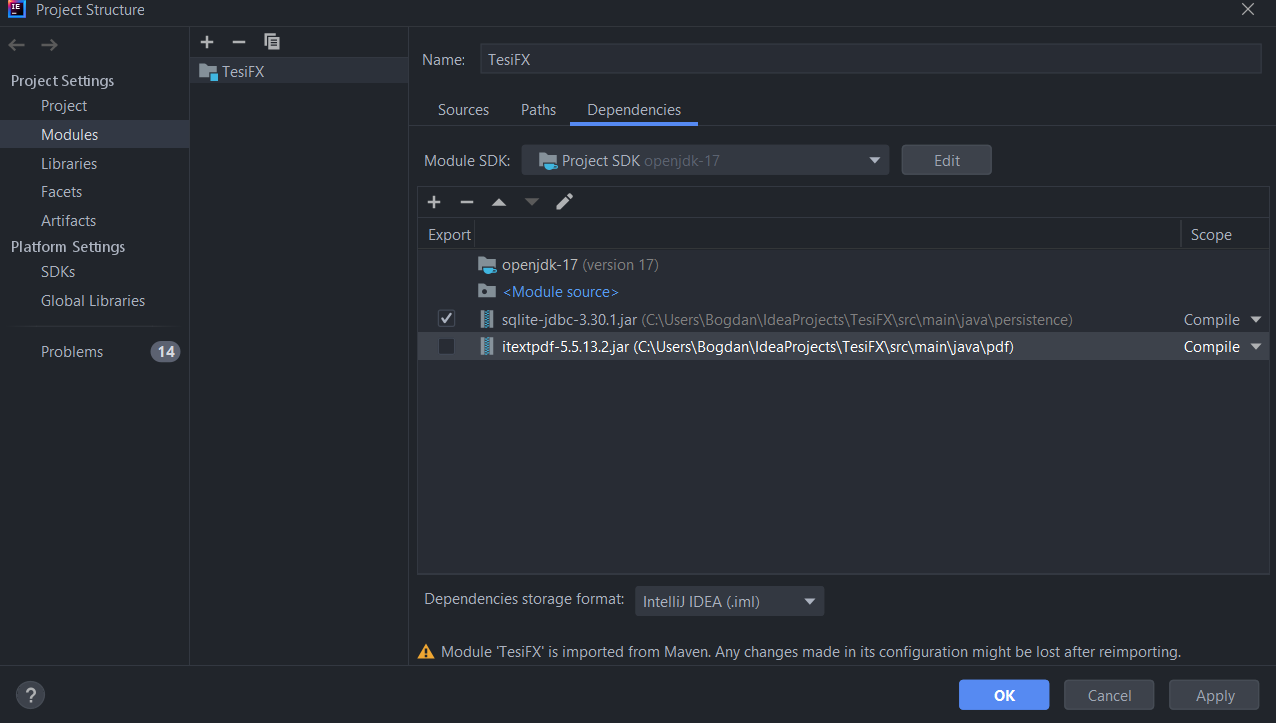
# Descrizzione :

SQLite e una libreria che implementa un database manager system SQL , e uno tra i piu usati per le applicazioni perche permette di creare un database incorporato in un unico file.

Puo essere importato in qualsiasi tipo di progetto , in un progetto con Maven tramite la dipendenza scrivendolo nel POM file e lui in automatico scaricherà il jar oppure importando a mano il jar. Una volta importato non ci serve altro che usare il database. Il linguaggio utilizzato per fare richieste sul database e per la maggior parte SQL. Le foto piu avanti mostreranno i due modi per importare SQLite in un progetto e come usarlo.

**Qui SQLite viene importato tramite dipendenza nel POM file** 

**Qui importiamo a mano il jar di SQLite**



**Esempio che seleziona nome ,cognome ed età da un database**



## CSS per JavaFx

# Descrizzione :

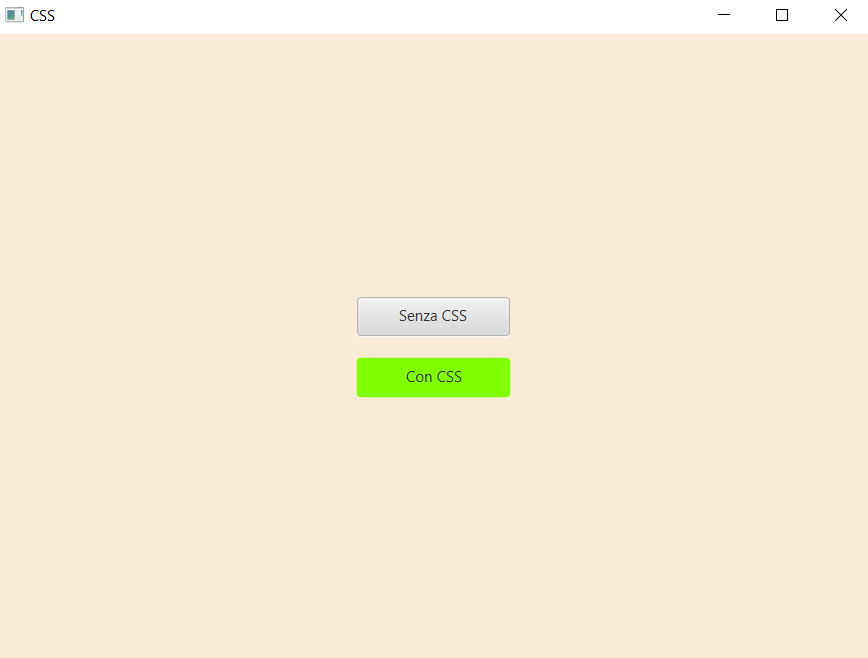
## CSS è un linguaggio di Styling che separa il contenuto di una pagina e la sua presentazione , il suo acronimo significa Cascading Style Sheets . Il linguaggio serve a dare dello stile al contenuto da noi inserito in una pagina ed e usato prevalentemente nelle pagine web , ma un’applicazione JavaFX puo anch’essa contenere del codice CSS solo che il codice CSS usato in un’applicazione JavaFX ha una sintassi un po' differente ma offre le stesse funzionalità. In un’applicazione o anche pagina web , ogni pagina dovrebbe avere un suo CSS , non e obbligatorio farlo ma e consigliato per avere un codice pulito e facile da interpretare . se si mettesse tutto in un unico file CSS non riusciremmo a distinguere a quale pagina appartiene un certo stile.

## Il formato di un file CSS è il seguente ed e lo Style.css che cambia il file FXML

**Immagine che contiene testo, screenshot, monitor, schermo

Descrizione generata automaticamente**

**Qui si vedono due pulsanti e lo sfondo , ed e ciò che implementa il file di sopra**



## Pattern MVC e Singleton

# Descrizzione :

Un pattern non è altro che una soluzione progettuale da applicare in determinate situazioni, quindi un modo di risolvere un determinato problema nell’istante della progettazione di un sistema.

Il pattern MVC o Model-View-Controller non e altro che un modo per separare l’architettura del nostro progetto , non è un tool , una classe o un’interfaccia che possiamo utilizzare .ma soltanto una rappresentazione astratta che separa la logica dall’interfaccia e le collega attraverso il controller. Quindi il pattern MVC serve a rendere piu semplice e leggibile il nostro progetto , rendendo la logica e l’interfaccia indipendenti , ma che lavorano insieme tramite il controller che prende l’input dell’utente dall’interfaccia e la passa alla logica , viceversa prende il risultato dato dalla logica e lo passa all’interfaccia.

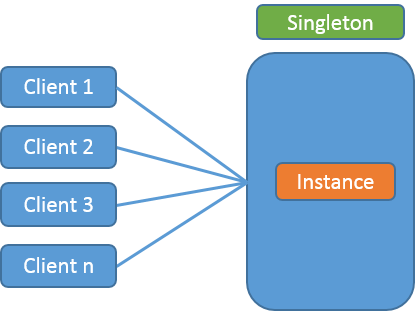
Il controller può essere anche visto come un altro pattern chiamato Mediator il cui ruolo e stare tra due o piu oggetti o interfacce e farle comunicare attraverso di lui , una cosa simile alla torre di controllo di un aeroporto , gli aerei non comunicano direttamente tra loro ma attraverso la torre di controllo, cosi come accade nel pattern MVC.

**Qui sotto abbiamo un esempio di come funziona il pattern MVC**

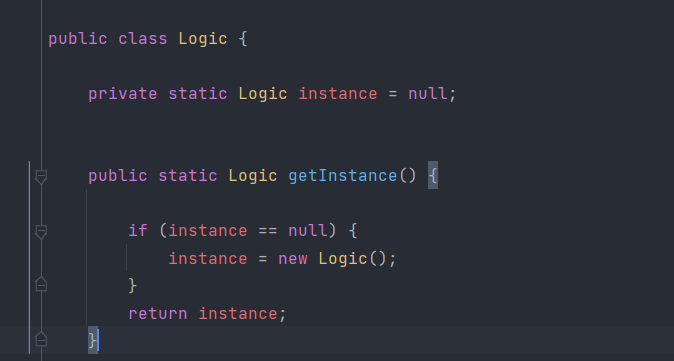


Il pattern singleton è un pattern creazionale il cui scopo e avere un’unica e sola istanza di un oggetto , questo fa si che un certo oggetto non possa essere creato piu di una volta , ma una volta creato tutti lavorano soltanto su di lui e non su un’altra sua copia. Si pensi ad esempio al protagonista di un film , il protagonista di solito è soltanto uno , l’antagonista anche , quindi il pattern singleton fa in modo che ci sia soltanto un protagonista , soltanto un antagonista , un solo narratore e cosi via a seconda di che oggetti abbiamo bisogno ma sempre fatto in modo che ne esistano solo uno di ogni tipo.

**Un esempio dell’architettura progettuale del pattern singleton**



**Qui invece si vede come viene implementato il pattern singleton**



## IText 5

# Descrizzione :

## IText 5 e una libreria java per la creazione e la manipolazione di file in formato PDF , e una tra le piu usate librerie quando si lavora con file in formato PDF. Come SQLite puo essere importato nel sistema come jar oppure tramite dipendenza nel file POM per poi essere scaricato ed installato automaticamente.

## Di seguito viene riportato un esempio di utilizzo di IText 5

## 

## 

## Il risultato del esempio precedente è il seguente

## Immagine che contiene testo Descrizione generata automaticamente

## Metabolismo Basale , Livello di Attività Fisica e Fabbisogno Giornaliero

## Descrizione:

## Il metabolismo basale è quanta energia spende un organismo vivente nello stato di riposo , è un concetto usato nel calcolo della dieta di una persona. Il metabolismo basale si calcola utilizzando il peso della persona , la sua età e la sua altezza e dei valori già prestabiliti della formula , che cambiano in base al sesso. Una formula molto usata nel calcolo del metabolismo basale è l’equazione di Harris-Benedict.

## Di seguito vengono riportate le due formule per il calcolo del metabolismo basale , la prima per gli uomini e la seconda per le donne

## MBuomo = 88,362 + (13,397 X peso (kg)) + (4,799 X altezza (cm)) – (5,677 X età (anni))

## MBdonna = 447,593 + (9,247 X peso (kg)) + (3,098 X altezza (cm)) – (4,330 X età (anni))

## Come si nota i valori cambiano a seconda del sesso , ma la formula resta la stessa

## Il livello di attività fisica , chiamato anche metabolismo cinetico comprende quanta energia si spende giornalmente facendo sport o attività lavorativa , e l’opposto del metabolismo basale , uno calcola l’energia spesa a riposo , l’altro in movimento. Piu attività fisica si svolge e più energia si consuma , quindi questo valore cresce in base a quanto e intensa l’attività fisica svolta durante la giornata . Il livello di attività fisica e soltanto un valore che varia in base all’attività svolta.

I livelli di attività fisica vengono riportati di seguito

LAF Sedentario = 1,45

LAF Leggermente attivo = 1,60

LAF Attivo = 1,75

LAF Molto attivo = 2,10

Fabbisogno calorico giornaliero è quanta energia ha bisogno ogni individuo per arrivare a fine giornata , in poche parole comprende il metabolismo bassale e il livello di attività fisica. La formula per calcolare il fabbisogno giornaliero di calorie è la seguente

Fabbisogno calorico = MB\*LAF

Questa modalità che utilizza il metabolismo bassale e il livello di attività fisica per calcolare il fabbisogno giornaliero di calorie è tra le più accurate .

# Descrizione del progetto realizzato

## Interfaccia

## Descrizione:

## L’interfaccia grafica è completamente creata utilizzando JavaFX Scene Builder e JavaFx , è composta da sette scene , alcune usate interamente per prendere l’input dell’utente , altre usate soltanto per mostrare i risultati. Lo stile usato è uno stile semplice ed appropriato per far in modo che l’utente che utilizzerà il sistema abbia una certa famigliarità con esso e non abbia difficoltà nel interagirsi con l’interfaccia e i suoi componenti. Per fare ciò si e utilizzato la tecnica KISS e ciò’è keep it simple stupid che dice di utilizzare un design semplice perché offre piu affidabilità. I colori utilizzati sono anch’essi semplici ed utilizzati in modo da aiutare l’utente nell’interazione con il sistema. Anche per i messaggi di errore o dialogo si è fatta cura a renderli piu semplici ed appropriati per gli utenti.

### Scena 1

La prima scena che trova quando si avvia il sistema è la Scena 1 , questa scena offre la possibilità di aggiungere un nuovo utente oppure di cercare uno già esistente per rimuoverlo o aggiornarlo. L’interfaccia è composta da container usati per posizionare i componenti nella scena , e componenti come pulsanti come quello di aggiunta , rimozione o aggiornamento , una barra di ricerca in cui mostra le persone salvate nel sistema e da anche la possibilità di ricercarne una digitando il suo nome , ed una lista in cui mostra le persone.

**Di seguito viene mostrato come si trova la scena quando viene avviato il sistema**



**Cliccando sulla barra di ricerca i risultati vengono mostrati nel seguente modo**

## 

## Selezionando una persona invece ci vengono mostrati i seguenti p

## 3.1.2 Scena 2

Descrivere l’architettura del progetto, la sua implementazione e i risultati ottenuti.

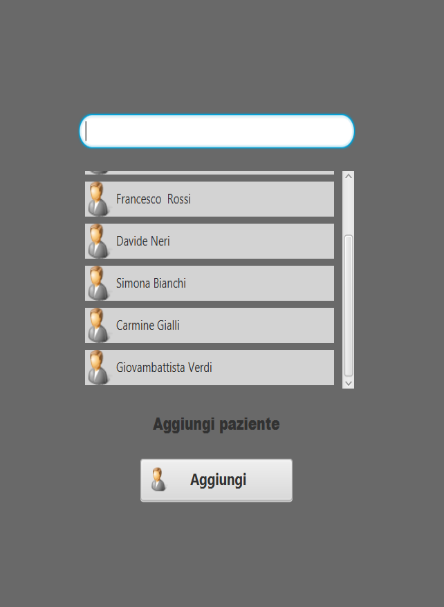
Java e un linguaggio di programmazione fantastico [2].

L’interfaccia grafica e stata creata utilizzando JavaFX Scene Builder per poi essere importata sotto forma di codice in formato FXML che e il formato di un progetto JavaFX.

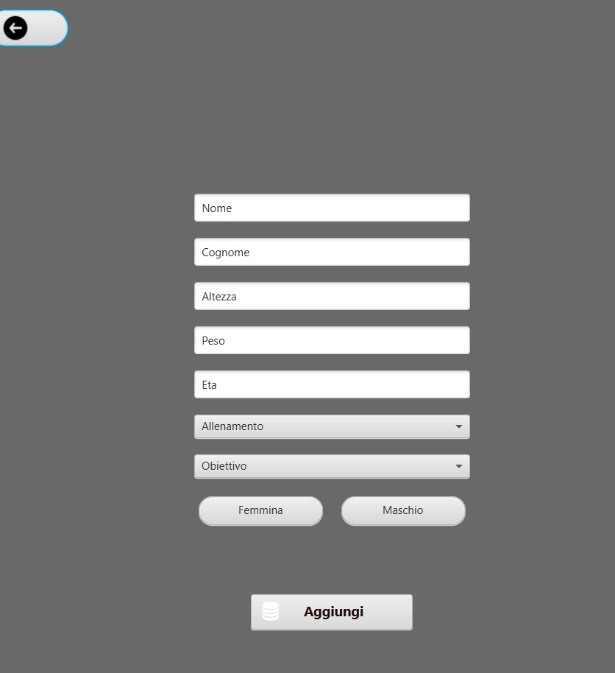
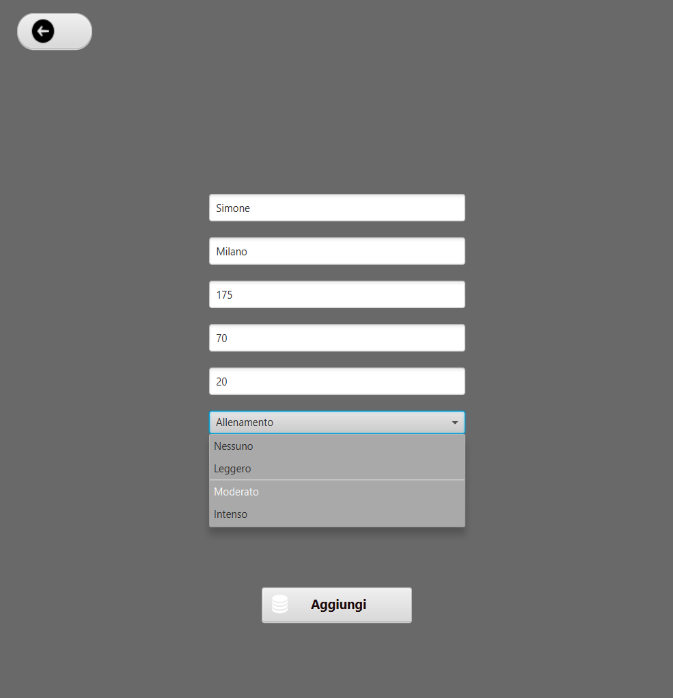
 Immagine che contiene testo

Descrizione generata automaticamente

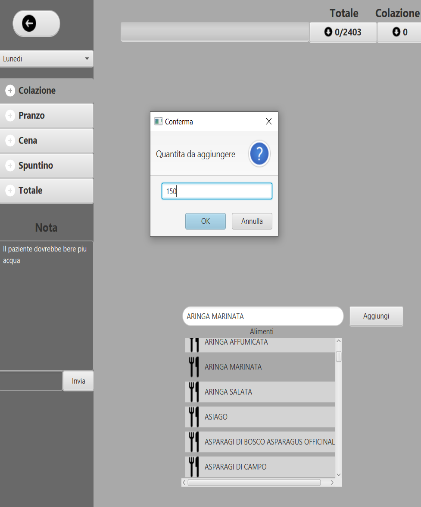
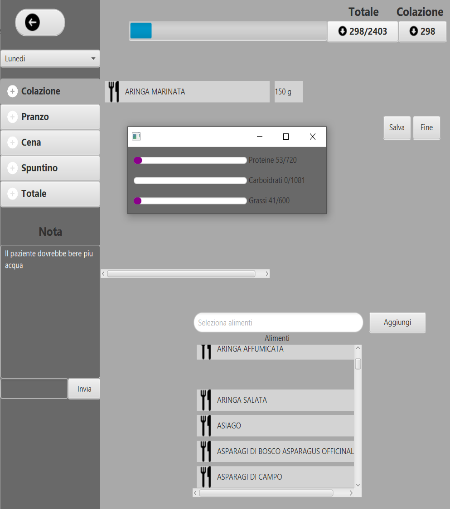
L’interfaccia e composta da sette “Scene” , la prima o quella iniziale serve ad aggiungere una nuova persona oppure cercarne una tra le esistenti , cliccandoci sopra la persona scelta si può rimuovere oppure aggiornare

Cliccando su aggiungi ci manda ad un’altra “Scene” dove si inseriscono i dati della persona e si seleziona allenamento e obbiettivo

Cliccando su aggiungi si va ad una pagina successiva dove si crea la dieta della persona , in questo caso Simone Milano oppure si clicca sul pulsante in alto a sinistra e si torna alla pagina successiva. Qui troviamo una finestra dove poter scegliere il giorno ed il pasto per cui scegliere gli alimenti , possiamo aggiungere una nota , oppure vedere il riepilogo degli alimenti inseriti per poi stampare la dieta in formato PDF . Abbiamo anche un valore di calorie da raggiungere che si incrementa ogni volta che aggiungiamo un alimento , con il valore specifico del alimento , possiamo scegliere la quantità di grammi , chilogrammi , litri , millilitri o cucchiaini del alimento scelto e cliccando sul valore delle calorie ci esce anche una sottospecifica della quantità di macronutrienti aggiunti in proteine , grassi e carboidrati , anche questi calcolati in base ai dati della persona . Cliccando fine torniamo alla pagina iniziale e i dati inseriti si salvano.

  Immagine che contiene testo

Descrizione generata automaticamente

# Conclusioni

Riassumere il lavoro svolto, eventuali problemi incontrati e possibili sviluppi futuri (circa 1 pagina)

Il lavoro in prattica salva una persona , che sara il nostro utente a cui dovra essere fatta una dieta , ci fa scegliere gli alimenti che comporranno la sua dieta , sette giorni la settimana e per ogni pasto , ci fa stampare la sua dieta in formato PDF . vedere in suoi cambiamenti nel tempo ed aggiornare i dati inseriti ogni qualvolta si vuole.

# Bibliografia

Inserire i riferimenti bibliografici del materiale utilizzato per la realizzazione della tesi.

[1] sito web di javafx. URL: https…. Consultato in data 09/11/2021.

[2] Seconda citazione.