POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Gestionale

Classe L8 – Ingegneria dell’Informazione



**Sviluppo di un software per la generazione automatica di una maratona cinematografica**

**Relatore Candidato**

Prof. Fulvio Corno Grandi Emanuele

269431

A.A. 2021/2022

**Sommario**

[1 Proposta di progetto 1](#_Toc111389941)

[1.1 Studente proponente 1](#_Toc111389942)

[1.2 Titolo della proposta 1](#_Toc111389943)

[1.3 Descrizione del problema proposto 1](#_Toc111389944)

[1.4 Descrizione della rilevanza gestionale del problema 1](#_Toc111389945)

[1.5 Descrizione dei data-set per la valutazione 2](#_Toc111389946)

[1.6 Descrizione preliminare degli algoritmi coinvolti 3](#_Toc111389947)

[1.7 Descrizione preliminare delle funzionalità previste per l’applicazione software 4](#_Toc111389948)

[2 Descrizione del problema affrontato 4](#_Toc111389949)

[3 Descrizione del data-set utilizzato per l'analisi 4](#_Toc111389950)

[3.1 Introduzione 4](#_Toc111389951)

[3.2 Tabella Data\_tot 5](#_Toc111389952)

[6](#_Toc111389953)[4 Descrizione ad alto livello delle strutture dati e degli algoritmi utilizzati 6](#_Toc111389954)

[4.1 Descrizione delle strutture dati: i packages e le classi 7](#_Toc111389955)

[4.1.1 it.polito.tdp.PF 7](#_Toc111389956)

[4.1.2 it.polito.tdp.PF.db 7](#_Toc111389957)

[4.1.3 it.polito.tdp.PF.model 7](#_Toc111389958)

[4.2 Descrizione degli algoritmi utilizzati 8](#_Toc111389959)

[4.2.1 Sezione 1: Ricerca 8](#_Toc111389960)

[4.2.2 Sezione 2: Ricorsione 10](#_Toc111389961)

[5 Interfaccia e video dimostrativo 12](#_Toc111389962)

[5.1 Sezione Ricerca 12](#_Toc111389963)

[5.2 Sezione Maratona 13](#_Toc111389964)

[6 Risultati sperimentali e esempi 14](#_Toc111389965)

[6.1 Esempio 1 14](#_Toc111389966)

[6.2 Esempio 2 15](#_Toc111389967)

[6.4 Esempio 1 16](#_Toc111389968)

[6.5 Esempio 2 17](#_Toc111389969)

[6.6 Esempio Playlist 18](#_Toc111389970)

[7 Valutazioni sui risultati ottenuti 18](#_Toc111389971)

# 

# Proposta di progetto

## Studente proponente

s269431 Grandi Emanuele

## Titolo della proposta

Sviluppo di un Software per la generazione automatica di una maratona cinematografica

## Descrizione del problema proposto

L’obbiettivo del software è quello di permettere all’utente di generare automaticamente una maratona cinematografica, elaborata rispettando i vincoli dettati dall’utente stesso e comunicati al software mediante l’interfaccia applicativa.

In particolare, fornisce all’utente la possibilità di determinare una durata massima (in termini di minuti per la ricerca semplice e ore per la maratona) della sequenza dei Film (vincolo di solito assente nella ricerca dei Film nelle piattaforme di streaming più utilizzate); i generi privilegiati e il periodo di uscita dei titoli.

E’ possibile, inoltre, definire un vincolo sulla valutazione minima della critica, nell’elaborazione della maratona desiderata.

Il software permette, per quanto riguarda le Serie Tv, di esprimere una preferenza per una determinata durata media degli episodi.

Funzioni implementate sono quelle di ricerca semplice (senza generare una maratona) di Film e Serie Tv (forniti i vincoli da parte dell’utente) e di aggiunta dei Film e Serie a una playlist personale “Guarda più tardi”, per tenere traccia dei titoli da vedere in un secondo momento.

## Descrizione della rilevanza gestionale del problema

Il software si focalizza, in primo luogo, su una globale ottimizzazione dei tempi.

L’utente, infatti, grazie al vincolo sulla durata massima dei Film, delle Serie e della maratona, potrà effettuare una ricerca su misura, in base al proprio programma giornaliero, al proprio tempo libero e in generale alle proprie esigenze in termini di tempistiche.

I titoli proposti dal software si incastreranno perfettamente nelle finestre temporali stabilite dal fruitore, evitando di dover interrompere e metà la visione di un film o di perdere tempo dietro a Serie Tv dagli episodi troppo lunghi.

Oltre a fornire informazioni generali sui titoli proposti, il software segnalerà all’utente la piattaforma di streaming in cui il titolo è attualmente disponibile (tra quelle più utilizzate), semplificando ulteriormente la ricerca.

Il processo di generazione della maratona cinematografica, infine, porterà chiaramente alla generazione di molteplici possibilità per l’utente, che potrà per esempio stabilire una valutazione minima (in termini di voto della critica e del pubblico) dei titoli che la compongono.

## Descrizione dei data-set per la valutazione

Il software si appoggerà a un data-set da me costruito ed elaborato a partire da alcuni data-set forniti dalla piattaforma Kaggle.

In particolare, sono stati utilizzati data-set che contengono informazioni su Film e Serie Tv presenti nelle principali piattaforme di streaming (Netflix, Disney +, Amazon Prime video), combinati con altri set di dati, per poter fornire all’utente informazioni sui titoli come: una breve descrizione del titolo, anno di uscita, ratings (dal Sito IMDb), informazioni sul genere e sul tipo di contenuti, le fondamentali informazioni sulla durata e la presenza o meno su una data piattaforma.

I dataset sono aggiornati a Novembre 2021.

Il database completo è presente nel repository (imdb.sql) ed è composto da 5 tabelle: netflix\_titles, amazon\_prime\_titles, disney\_plus\_titles, imdb\_copy e l’ultima data\_tot, che rappresenta il data-set effettivo utilizzato per l’applicazione (contenente però solamente i film e non le serie tv, poiché non considerate nella creazione della maratona), frutto della combinazione degli altri 4.

**Le righe totali del database (data\_tot) sono 1272 e gli attributi sono:**

* Nome;
* Anno di Uscita;
* Valutazione Imdb;
* Durata in minuti;
* Genere (uno o più);
* Livello di Nudità;
* Livello di Violenza;
* Livello di Alcool;
* Livello di Paura;
* Breve Descrizione del Titolo;
* Piattaforma in cui è presente (N o P o D).

## Descrizione preliminare degli algoritmi coinvolti

La prima parte sarà costituita da algoritmi di ricerca per la visualizzazione di record e l'elaborazione dei dati estratti.

Per la seconda parte, Il software lavorerà sul database costruito con l’obbiettivo di costruire una sequenza di titoli, vincolata alle preferenze dell’utente.

Per fare ciò si baserà su un algoritmo ricorsivo, che a partire da un titolo casuale che rispetti tutti i filtri, genererà una maratona che dovrà essere ottimizzata sulla base della durata massima stabilita, della valutazione minima dei titoli richiesta dall’utente e su un serie di ulteriori parametri, che si potrà far variare a piacere (livello di paura del Film, livello di violenza…).

In tal modo, ogni volta che l’utente vorrà generare una maratona, il software cercherà di variare la proposta se possibile.

Si tratta quindi di un problema dello zaino (knapsack problem).

## Descrizione preliminare delle funzionalità previste per l’applicazione software

L’utente si troverà davanti a una serie di vincoli da determinare nella ricerca o nella generazione della maratona ad hoc.

Nella prima tab, l’utente potrà scegliere tramite una combobox se includere Film o Serie Tv, spuntare le piattaforme desiderate tramite checkbox e dopodichè impostare una serie di filtri, quali il genere (combobox), il periodo di uscita (2 combobox) e la durata massima dei film (durata massima media degli episodi per le Serie Tv) per la ricerca dei Titoli.

Il software evidenzia il titolo con la valutazione migliore (in caso di valutazione uguale, viene scelto il primo in ordine alfabetico) e fornisce breve descrizione (in inglese) di esso.

Sempre nella prima tab, dopo aver effettuato la ricerca, l’utente potrà aggiungere uno o più titoli, tra quelli proposti, ad una playlist “Guarda più Tardi”.

Nella seconda Tab, quella predisposta alla creazione di una maratona, oltre agli stessi vincoli presenti nella prima parte, sono presenti degli slider per stabilire il livello massimo di violenza, paura, nudità e alcool, la durata totale massima in ore della maratona e la valutazione minima dei titoli che ne faranno parte.

L’applicazione mostrerà la valutazione media e la durata totale della maratona proposta.

Per quanto riguarda la creazione della maratona ho preferito escludere le serie Tv, in quanto risulta poco sensato a mio parere inserire uno o più episodi di una serie all’interno di una maratona, in sequenza con dei Film.

I risultati proposti dal software mostreranno il nome del Film o della Serie tv, l’anno di uscita e la Piattaforma in cui sono disponibili (N, P, D) (Agg. Novembre 2021).

# Descrizione del problema affrontato

Il problema affrontato è principalmente di ottimizzazione.

L’idea parte dall’osservazione del fatto che sulle principali piattaforme di streaming di Film e Serie tv non è possibile effettuare in modo semplice ed immediato una ricerca per durata massima dei Titoli, così come risulta macchinoso limitare la ricerca ad un determinato periodo temporale di uscita dei Titoli.

Il software si pone l’obbiettivo di adattarsi alle esigenze degli utenti e ai loro impegni giornalieri, che limitano le finestre temporali in cui potersi godere un buon Film o cominciare una Serie Tv.

Attraverso un algoritmo ricorsivo, l’applicazione genera in automatico una maratona ad hoc, a partire dalle preferenze dell’utente, che stabilirà la durata massima totale, il periodo di uscita del Film, la valutazione minima dei Titoli che ne faranno parte e regolerà alcuni parametri come il livello di Violenza o Paura del Film.

L’applicazione ovviamente non è pensata come un elemento sostitutivo di tali piattaforme, bensì come un elemento di supporto.

L’utente sarà informato della piattaforma in cui sono presenti i titoli, velocizzando la ricerca e la successiva fruizione.

Infine, ho ritenuto non significativa l’implementazione della ricerca per Attori o Registi, sia per problemi di completezza dei dataset, sia perché tale tipo di ricerca è facilmente eseguibile sulle piattaforme di streaming.

# Descrizione del data-set utilizzato per l'analisi

## Introduzione

Per creare il software sono state utilizzate parti di un database fornito dal sito Kaggle, il quale afferma che è possibile usufruirne liberamente.

I dati forniti erano tutti in formato CSV, di conseguenza è stata svolta una conversione in formato SQL per permetterne l’importazione attraverso il software HeidiSQL.

A tal proposito è stato creato nell'ambiente di HeidiSQL un nuovo database chiamato "imdb" contenente le tabelle e i parametri necessari ad accogliere il contenuto attraverso il comando import CSV.

Al fine di non affaticare l'applicazione con eccessivi calcoli si è preferito snellire il database eliminando dati errati e/o ridondanti, nonchè oggetti inutili ai fini del software.

Il database "imdb" è composto da 5 tabelle, come scritto in precedenza.

Le tabelle imdb\_copy, netflix\_titles, amazon\_prime\_titles,\_disney\_plus\_titles sono quelle originarie dal sito Kaggle (utilizzate per la ricerca semplice di titoli), tuttavia al fine di non affaticare l'applicazione con eccessivi calcoli si è preferito snellire il database eliminando dati errati e/o ridondanti, nonchè oggetti inutili ai fini del software (regista, attori, Paese, id Titoli, anno aggiunta nelle piattaforme) combinando i dati utili nella tabella data\_tot (più semplice da utilizzare per la ricorsione).

Per questa ragione descriverò solamente tale tabella, riassunto delle informazioni utili delle altre.

* Amazon\_prime\_titles
* Data\_tot
* Disney\_plus\_titles
* Imdb\_copy
* Netflix\_titles

Immagine che contiene testo

Descrizione generata automaticamente

Figura 3.1 Data-set completo

## Tabella Data\_tot

La tabella è composta da 1272 elementi, che rappresentano i Film presenti nel database

Di seguito sono elencate le colonne contenenti informazioni utili:

1. \item “Nome” rappresenta il titolo del Film in inglese.
2. \item “Anno” identifica univocamente insieme al Nome il Titolo.
3. \item “Voto” contiene la valutazione Imdb del Titolo (da 2 a 9)
4. \item “Durata” indica durata in minuti del Film.
5. \item “Genere” include il genere o i generi in cui il Titolo rientra.
6. \item “Nud” contiene un valore da 0 a 3 che indica la presenza di Nudità nel Film. (0 nessuna, 1 blanda, 2 moderata, 3 significativo)
7. \item “Viol” contiene un valore da 0 a 3 che indica la presenza di Violenza nel Film. (0 nessuna, 1 blanda, 2 moderata, 3 significativa)
8. \item “Alc” contiene un valore da 0 a 3 che indica la presenza di Alcool nel Film. (0 nessuna, 1 blanda, 2 moderata, 3 significativa)
9. \item “Fri” contiene un valore da 0 a 3 che indica il livello di Paura che solitamente il Titolo suscita alla visione . (0 nessuno, 1 blando, 2 moderato, 3 significativo)
10. item “Descr” contiene una breve descrizione in inglese del Film.
11. item “Piattaf” indica la piattaforma in cui è presente il Titolo (N netflix, P prime video, D disney +).

## Immagine che contiene tavolo Descrizione generata automaticamente

Figura 3.2 Tabella data\_tot

# Descrizione ad alto livello delle strutture dati e degli algoritmi utilizzati

L’applicazione è stata sviluppata in linguaggio java seguendo il pattern MVC (Model-View-Controller) ed il pattern DAO (Data-Access-Object) tramite i quali sono stati separati l’accesso al database, i processi algoritmici per la manipolazione dei dati e l’interfaccia utente.

## Descrizione delle strutture dati: i packages e le classi

I packages sono utili a tenere separate le varie fasi, abbiamo infatti tre packages: uno per il model, uno per il controller e uno per il database.

### it.polito.tdp.PF

Il package si occupa dell’interfaccia utente e della procedura di avvio dell’applicazione tramite tre classi:

**EntryPoint**: necessario per l'avvio dell'applicazione stessa

**Main**: gestisce avvio interfaccia nel quale verranno inseriti i dati necessari.

**FXMLController**: riceve dati in Input e collega l'interfaccia con la logica applicativa tramite l’oggetto model. I occupa quindi di interpretare e validare i dati inseriti.  
In questa classe avranno luogo gli algoritmi che andranno a comporre l’UI che mostrerà anche gli output.

### it.polito.tdp.PF.db

Questo package ci permette di effetuare la connessione alla base dati e di interrogare la suddetta attraverso query scritte in linguaggio SQL.

E’ composto dalla classe **DBConnect**, **IMDBDao** :

* **DBConnect** In questa classe avviene la connessione con il database tramite pooling che permette di risparmiare tempo di apertura e chiusura di una connessione grazie alla classe HikariDataSource.
* **IMDBDao** Questa classe riceve i parametri dal Model (che nella maggior parte dei casi vengono passati dal Controller) e interroga il database tramite query SQL. Crea delle strutture dati idonee all’organizzazione dei dati raccolti e li restituisce al Model.

### it.polito.tdp.PF.model

Questo package contiene la logica applicativa e svolge la maggior parte delle operazioni e dei calcoli.

È’ composto da 2 classi:

* **Model** È la classe portante del progetto, qui sono contenuti tutti i dati e le strutture necessari al corretto svolgimento dei processi. Il model collega il database e l’interfaccia utente e al suo interno viene creato il grafo semplice che rappresenta i film appartenenti allo stesso genere e che rispettano tutti i filtri stabiliti dall’utente (periodo temporale, valutazione minima, livelli di violenza…).
* **Titolo** I cui attributi sono le 11 colonne della tabella data\_tot.

Immagine che contiene testo

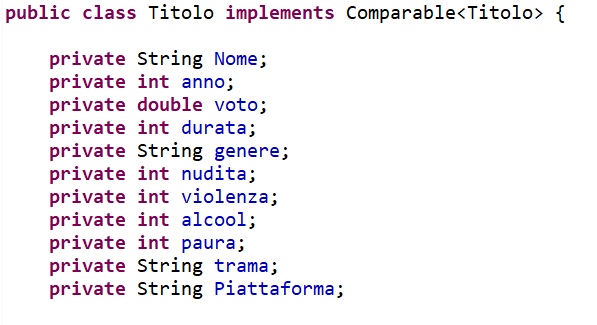
Descrizione generata automaticamente 

Figura 4.1 Packages e relative classi Classe Titolo

## Descrizione degli algoritmi utilizzati

### Sezione 1: Ricerca

Nella prima tab, l’utente potrà scegliere tramite una combobox se includere Film o Serie Tv, spuntare le piattaforme desiderate tramite checkbox e dopodichè impostare una serie di filtri, quali il genere (combobox), il periodo di uscita (2 combobox) e la durata massima dei film (durata massima media degli episodi per le Serie Tv) per la ricerca dei Titoli.

Sempre nella prima tab, dopo aver effettuato la ricerca, l’utente potrà aggiungere uno o più titoli, tra quelli proposti, ad una playlist “Guarda più Tardi”.

L’utente in questa prima parte può scegliere se effettuare la ricerca per genere e per periodo o solo per uno dei due filtri, ciò ha reso necessario creare più metodi nel IMDBDao.

Nel DAO

Immagine che contiene testo

Descrizione generata automaticamente

Immagine che contiene testo

Descrizione generata automaticamente

Immagine che contiene testo

Descrizione generata automaticamente

3 metodi ripetuti anche per le altre 2 tabelle: Amazon\_Prime\_titles e Disney\_plus\_titles.

### Sezione 2: Ricorsione

La ricorsione si basa sulla creazione di un grafo semplice i cui vertici sono tutti quei film che durano meno della durata massima della maratona impostata dall’utente e che rispettano tutti i filtri impostati (vertici ottenuti tramite metodo nel DAO, che lavora sul data-set finale data-tot). I vertici del grafo saranno poi tutti collegati fra loro e verranno utilizzati per la creazione della maratona ottimale a partire da uno di essi scelto in modo casuale (Math.random), nel metodo Crea Cammino (), il quale si appoggia sul metodo ricorsivo cerca().

NEL DAO

Immagine che contiene testo

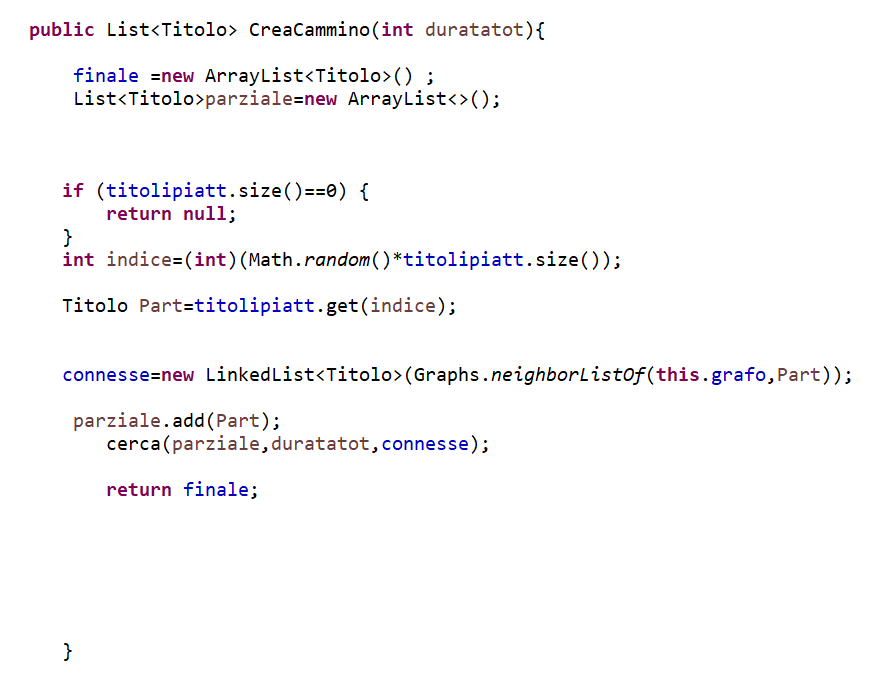
Descrizione generata automaticamente

NEL MODEL

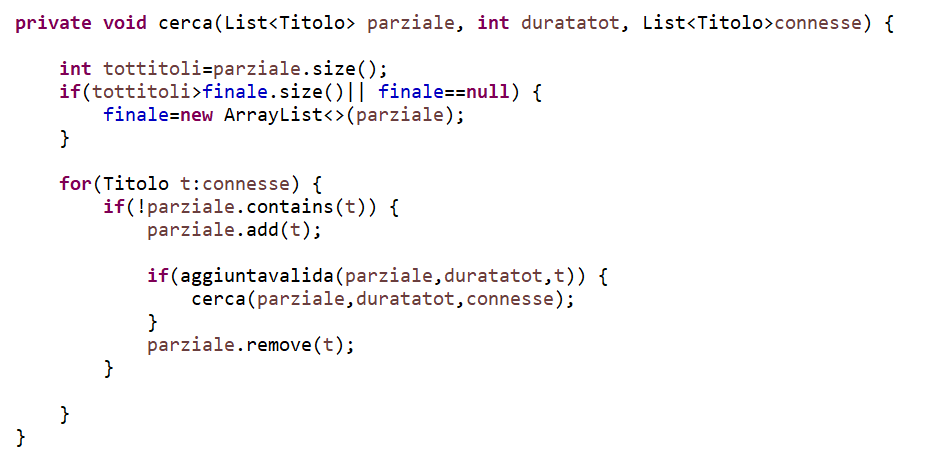
Immagine che contiene testo

Descrizione generata automaticamente

Metodo CreaCammino ()



Metodo cerca ()

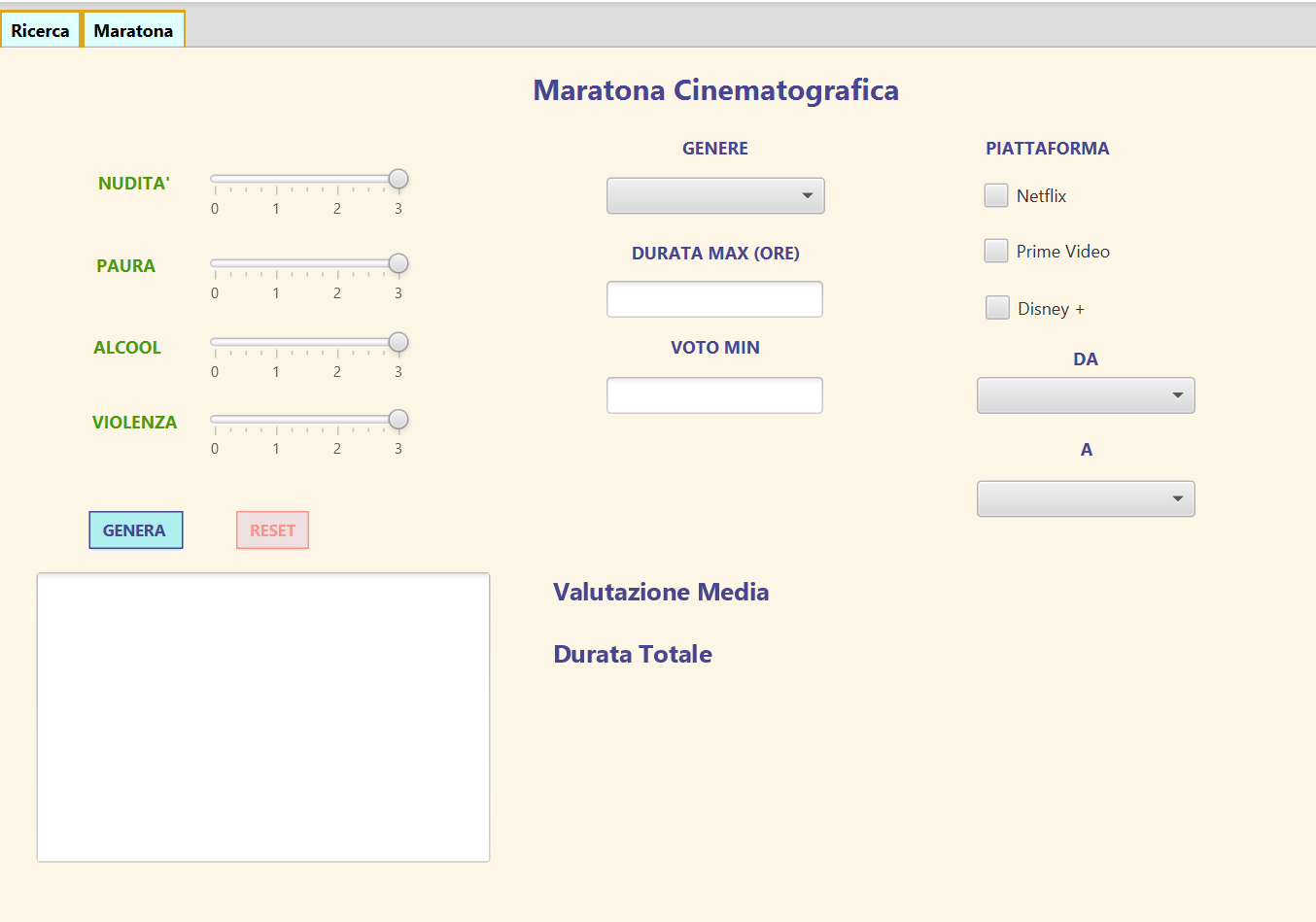


# Interfaccia e video dimostrativo

## Sezione Ricerca



## Sezione Maratona



Qui potrete trovare un video dimostrativo: <https://www.youtube.com/watch?v=wbMAJAll2gM>

# Risultati sperimentali e esempi

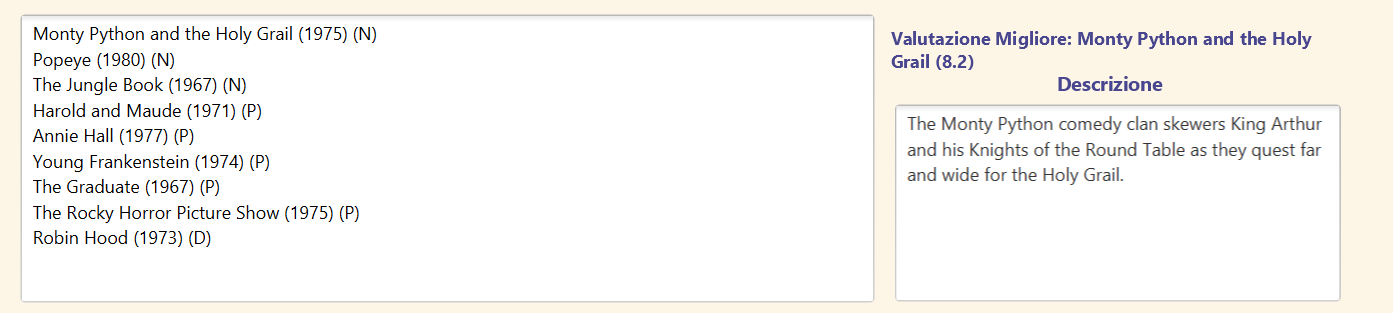
## Esempio 1

**Ricerca**

**Input:**

****

**Output:**

****

**Valutazione**

**Pur scegliendo tutte e 3 le piattaforme, una durata massima di 120 minuti (2 ore) e un genere molto comune(Comedy), il numero di Film consigliati è relativamente limitato.**

**Questo è dovuto al periodo scelto (1950-1980), che limita molto la scelta all’interno del data-set.**

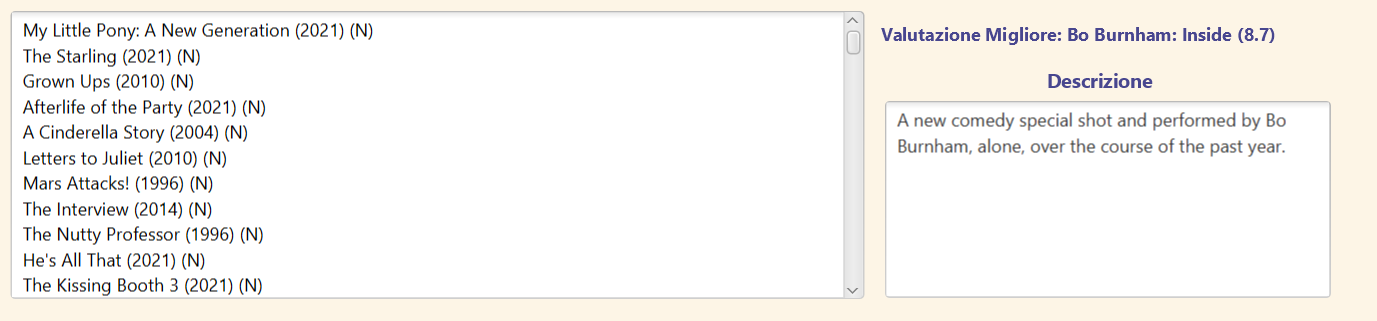
## Esempio 2

**Ricerca**

**Input:**

****

**Output:**



**Valutazione**

**Modificando solamente il periodo rispetto all’esempio precedente, la lista di titoli proposta aumenta notevolmente (1990-2021), poiché la maggior parte dei titoli nel database appartiene a tale periodo.**

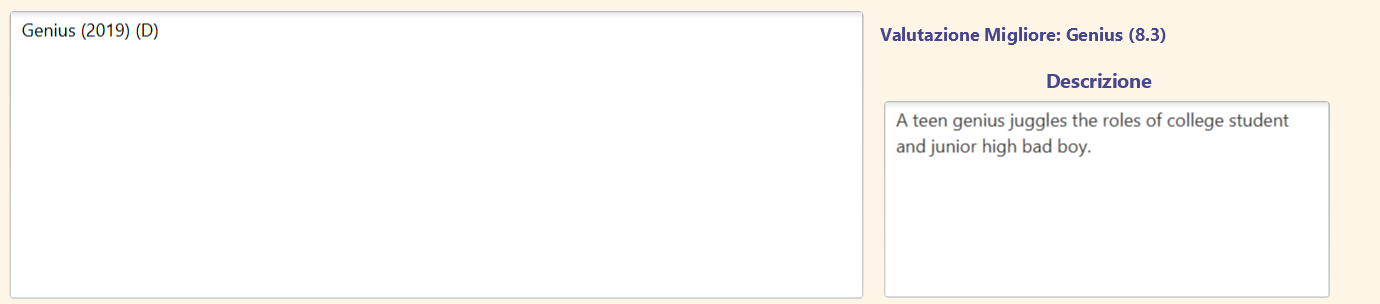
**6.3 Esempio 3**

**Ricerca**

**Input:**

****

**Output:**

****

**Valutazione**:

**Includendo solamente le serie tv e la piattaforma Disney +, nonostante il periodo specificato sia quello a cui corrispondono più film, il software è in grado di suggerire solo un titolo, in base ai filtri stabiliti.**

**Tutto ciò perché nel database i titoli presenti su Disney + sono relativamente limitati rispetto alle altre 2 piattaforme.**

**Inoltre, nel data-set utilizzato le serie tv costituiscono circa il 30% del totale dei Titoli.**

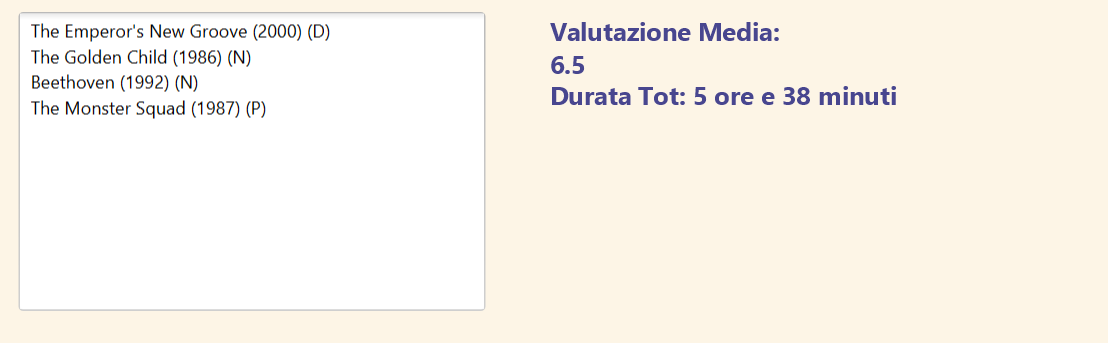
## Esempio 1

**Ricorsione**

**Input:**

****

**Output:**



**Valutazione**

**Applicando tali filtri e impostando una durata massima non elevata (7 ore) la creazione della maratona è immediata.**

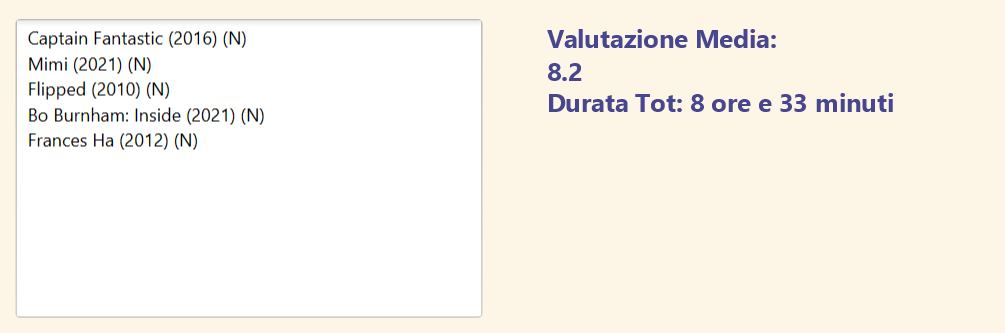
## Esempio 2

**Ricorsione**

**Input:**



**Output:**



**Valutazione:**

**Aumentando la durata totale a 10, lasciando i parametri massimi degli slider a 3, scegliendo il periodo da cui provengono la maggior parte dei Titoli (1990-2021), selezionando tutte le piattaforme, pur scegliendo una valutazione minima molto alta (8), il software impiega circa 4 secondi per creare la maratona (procedura non più immediata).**

## Esempio Playlist

La cmbTitoli () per l’aggiunta dei titoli alla playlist viene riempita con i titoli suggeriti dal software nel punto precedente.

Non appena il titolo viene aggiunto alla playlist viene immediatamente rimosso dalla combobox.



# Valutazioni sui risultati ottenuti

La pulizia dei data-set e la scelta degli attributi effettivamente presenti in tutti i record ha portato ad una riduzione della dimensione del database finale.

L’avere a disposizione un database relativamente limitato di Titoli ha rappresentato uno svantaggio per quanto riguarda la prima parte, votata alla ricerca semplice, ma d’altra parte ha rappresentato un requisito fondamentale per permettere all’algoritmo ricorsivo di fornire risultati in tempi accettabili nella seconda parte.

Mentre nella prima parte ho potuto lasciare una certa libertà nella compilazione dei vincoli (l’utente può selezionare il genere e il periodo temporale o solo uno dei 2; il periodo scelto deve essere di almeno 20 anni, per avere più risultati), nella seconda, per esigenze dettate dalla scelta della ricorsione (che può dilatare i tempi di creazione della maratona) , l’utente è obbligato a compilare tutti i campi ed è consigliato farlo cercando di limitare il campo di ricerca dei titoli.

Nella creazione della maratona, infatti, il periodo temporale scelto non deve essere maggiore di 35 anni e la durata massima deve essere minimo di 3 ore (per cercare di non avere solo un Film consigliato) ed è preferibile non superi le 11 ore (ovviamente in relazione alla ristrettezza degli altri vincoli).

Inoltre, il database finale (ripulito) è popolato per la maggior parte da titoli presenti su Netflix, penalizzando la ricerca per le altre 2 piattaforme.

Si nota inoltre uno sbilanciamento per quanto riguarda il periodo di uscita dei titoli, a favore dei titoli più recenti (più numerosi).

Tenendo presente le precedenti considerazioni riguardo i filtri e il database di partenza, il software può rappresentare uno strumento molto utile e divertente da utilizzare.

Immagine che contiene testo, clipart, segnale

Descrizione generata automaticamente

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc-sa/4.0/>