# STREAMGO
# Bd2

Labanca Paolo[1],Albanese Maria Giovanna[2]
[1]Università degli Studi di Salerno, p.labanca@outlook.it
[2]Università degli Studi di Salerno, maria.giovanna077@gmail.com

March 6, 2023

## 1 INTRODUZIONE

Attualmente un gran numero di utenti è alla ricerca di pattern di gestione dei dati che offrano scalabilità e dinamicità. Con le tecnologie relazionali, queste caratteristiche non possono essere sempre garantite, per questo motivo abbiamo bisogno di rivolgerci a una tecnologia più flessibile.
NoSQL raggruppa un insieme di modelli per la persistenza dei dati che funzionano in modo diverso dai database relazionali. Tale progetto si muove verso questo direzione, mirando all' utilizzi di una particolare dipologia di database NoSQL, di tipo document data store. Si discute il processo di sviluppo di StreamGo, una piattaforma di ricerca online dedicata al mondo dello streaming legalizzato. L'obbiettivo è quello di visualizzare tutte le informazioni di film e serie tv presenti su Netflix. Il documento si divide in 3 settori:

- Fase di pre-processing.
- Fase di back-end
- Fase di front-end

## 2 PRE-PROCESSING

Nella fase di pre-processing siamo andati a individuare e scaricare tre csv da Kaggle[1]. Questo set di dati è stato creato per elencare tutti gli spettacoli disponibili in streaming Netflix e analizzare i loro dati. Questi dati sono stati acquisiti a maggio 2022, contenenti dati disponibili negli Stati Uniti. I csv scaricati sono i seguenti: Credits.csv ( con 5 colonne e 7721 righe ),NetflixOriginals.csv ( con 6 colonne e 585 righe) ,Titles.csv ( con 15 colonne e 5807 righe ).

Dove, nel primo abbiamo tutti gli attori, con il loro personID di JustWatch,l'id del film, il nome ,il personaggio che interpretano e il ruolo, che puo essere attore o regista.
Nel secondo csv abbiamo altri film e serie presi da netflix, dove al suo interno abbiamo: il titolo, il genere, la data d'uscita, la durata, il IMDB Score cioè la media delle valutazioni e la lingua
Nel terzo abbiamo l'id del film, il titolo, il tipo di programma, cioè show o film, una breve descrizione, la durata, il genere, dove è stato prodotto, numero di stagioni, l'ID del titolo su IMDB, punteggio su IMDB[2], voti su IMDB, popolarità su IMDB e punteggio su IMDB.
Successivamente tramite Python[3] e la libreria Pandas[4], utilizzata per la manipolazione e l'analisi dei dati. Offre strutture dati e operazioni per manipolare tabelle numeriche e serie temporali. Tramite essa siamo andati a leggere i csv e abbiamo effettuato un primo merge, sul campo Title tra NetflixOriginals.csv e Titles.csv, dopo tale merge siamo andati a riempire i campi vuoti che comprendevano dei numeri con il valore 0. Dal csv risultante abbiamo eliminato le colonne non rilevanti che sono: tmdbScore, runtime, seasons, imdbScore, ageCertification.
Successivamnte è stato effettuato il marge, sull'id tra il csv definito in precedenza e credits.csv, al fine di avere solo le informazioni pricipali per gli attori presenti nei film selezionati e non tutti quelli presenti nel dataset,al termine di tale operzione sono state eliminate tutte le colonne non rilevanti, nello specifico riguardano: id, name, character e role. In fine i campi character che erano vuoti sono stati riempiti con il valore "other". Quindi nella fase conclusiva del preprocessing ci siamo ritrovati con due file: TitleTotalClean.csv ( con 13 colonne e 530 righe) e CreditsTotalClean.csv (con 4 colonne e 11727 righe) che sono collegate tra di loro tramite l'id, che lega gli attori con i/il film.
La struttura del progetto è la seguente:

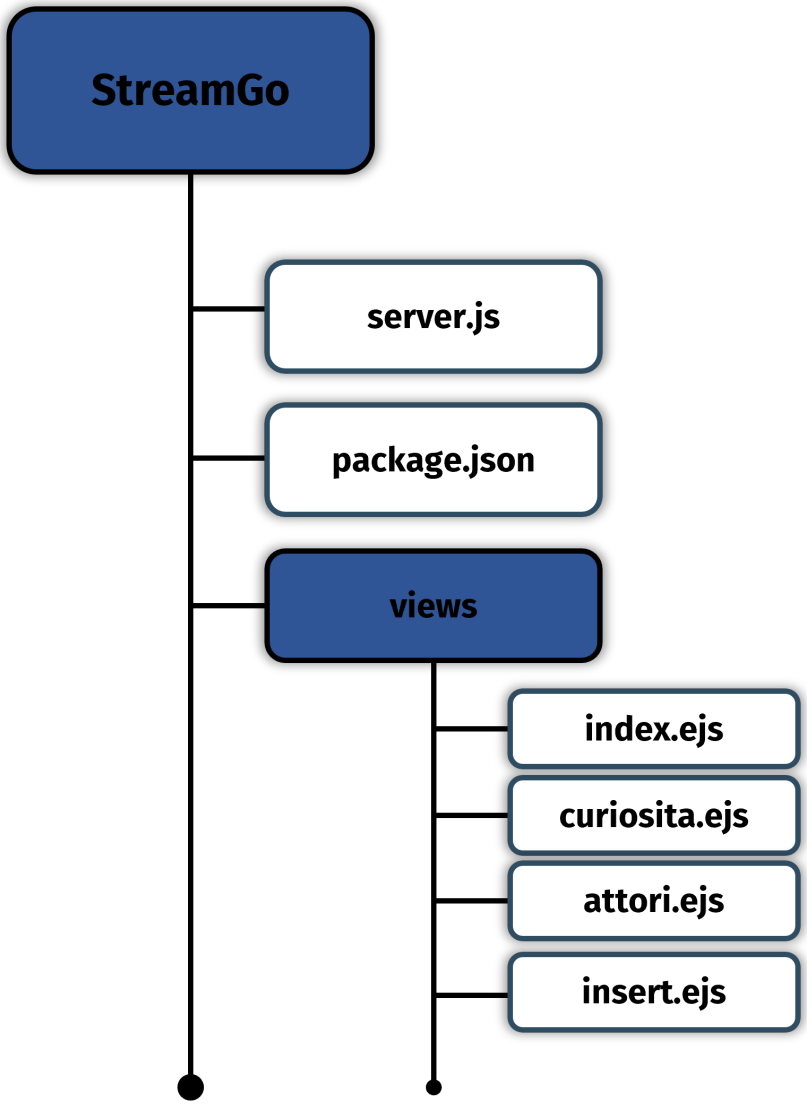


Figure 1: Struttura gerarchica StreamGo

# 3 BACK-END

Nella fase di back-end siamo andati a caricare i due csv in MongoDB[5] Atlas, andando a creare prima il database StreamGo e poi ad importare le due collection. MongoDB è un DBMS non relazionale, orientato ai documenti. Classificato come un database di tipo NoSQL, si allontana dalla struttura tradizionale basata su tabelle dei database relazionali in favore di documenti in stile JSON con schema dinamico (MongoDB chiama il formato BSON), rendendo l'integrazione di dati di alcuni tipi di applicazioni più facile e veloce, ma ci permette anche di mantenere i nostri dati in cloud. Successivamente abbiamo creato le nostre query in Nodejs[6] con l'ausilio del framework Express[7], che rientra nella famiglia dei runtime system open source multipiattaforma orientato agli eventi per l'esecuzione di codice JavaScript[8].La pagina Server.js rappresenta il front-end della nostra applicazione ed è siuddivisa in più parti: una prima parte in cui viene stabilita la connessione al DB e la porta di ascolto. Una seconda in cui vengono implementate le operazioni di manipolazione del DB.
Le query che sono state integrate sono le seguenti:

- Ricerca globale
- Ricerca per titolo
- Filtra per genere e lingua
- Filtra per data di produzione, scelta tra gli ultimi 2/5/10/20 anni
- Filtra per la maggiore valutazione
- Inserire un film
- Filtra per attori di un film
- Media del punteggio dei film per ogni anno
- Numeri dei film raggruppati per la lingiua

# 4 FRONT-END

Nella fase di front-end, tramite HTML[9] un linguaggio di markup. Nato per la formattazione e impaginazione di documenti ipertestuali disponibili nel web, utilizzato principalmente per il disaccoppiamento della struttura logica di una pagina web e la sua rappresentazione, gestita tramite gli stili CSS[10] per adattarsi alle nuove esigenze di comunicazione e pubblicazione all'interno di Internet. Grazie all'utilizzo di tali tecnologie, abbiamo realizzato le varie pagine in formato ".js" con le quali l'utente ha la possibilità di interagire con l'applicazione. Le pagine sono raggruppate all'interno della cartella vievs e sono le seguenti:

- La pagina inziale index con cui l'utente si interfaccia

- La pagina dedicata alla visualizzazione degli attori per un determinato film
- La pagina contenente il form per l'inserimento di un nuovo film o serie tv
- La pagina dedicata alle curiosità in cui vengono visualizzati due grafici,
  Nel primo vengono analizzate le valutazioni dei Titoli presenti ne DB in base all'anno di produzione
  Nel secondo vengono analizzati i titoli e vengono raggruppati in base alla lingua, al fine di valutare quali sono le lingue dominanti fra i film e le serie tv prposte

Le pagine sono state realizzate con l'ausilio di Bootstrap[11] e, per i grafici abbiamo utlizzato chart.js[12]

# References


[1] link site, https://www.kaggle.com/

[2] link site, https://www.imdb.com/

[3] link site, https://www.python.org/

[4] link site, https://pandas.pydata.org/

[5] link site, https://www.mongodb.com/

[6] link site, https://nodejs.org/

[7] link site, https://expressjs.com/

[8] link site, https://www.javascript.com/

[9] link site, https://www.html.it/

[10] link site, https://www.html.it/css/

[11] link site, https://getbootstrap.com/

[12] link site, https://www.chartjs.org/