Venkatesh Kashyap

+919395543140 | venkateshkashyap23@gmail.com | Linkedin

ABOUT ME

I am a self-taught developer, So I keep learning without limits. Every skill I've acquired as a developer has stemmed from solving specific problems encountered during my projects. Rather than seeking particular solutions, I prefer understanding the root cause of each issue and devising a solution of my own. This approach enables me to anticipate future problems and make necessary adjustments in the present.

PROJECTS

Good Movies 🕶️ | Github

- Good movies is a movie recommendation platform with a **Next.js frontend and a Flask backend**
- **Frontend**:The frontend fetches data from the TMDB API and manages user recommendations in Firebase, using **JWT** for data privacy. Movies are displayed as flip cards with options to add to a watchlist or view trailers. User feedbacks helps refine future suggestions
- **Backend**: The Flask backend is integrated with Firebase **firestore** to handle user-specific data and employs a **content-based recommender** system for personalised recommendations. API routes securely manage users and movie IDs to update recommendations.
- **Tech Stack**: **Typescript**, Python, Next.js, Flask, Firebase, **Node.js**.

Social Networking Backend | Github

- **User Registration and Authentication:** Implemented secure user signup and login using **Firebase Authentication**.
- **Discussion Posts :** Discussion posts support **CRUD** functionality with text, images, and hashtags. Users can comment on and like posts. Each comment can also be liked or replied to.
- **Search and Discovery:** Implemented Post search using hashtags
- **Data security** : Data security is ensured with **tokenized access keys** for all API interactions.
- **API endpoints** :Defined specific API routes for user and post-related interactions following **REST principles**
- **API testing** : Postman collection link
- **Tech Stack**: Next.js, Firebase, Firestore, Firebase Authentication, RESTful APIs

Content Based Recommender | Kaggle

- **Data Filtering**: Applied filtering based on multiple parameters such as runtime, vote count, vote average, and popularity to ensure a high-quality dataset.
- **Feature Engineering**: Extracted and processed genre and keyword information for each movie. Combine the relevant textual features into a single string or "**soup**" for each movie to ensure that all key descriptive elements are represented in a unified format.
- **Text Vectorization**: Utilised the CountVectoriser from the **Natural Language Processing** (NLP) toolkit in Scikit-Learn to transform the feature soups into a **token count matrix**.
- **Similarity Calculation**: Calculated cosine similarity between movies based on their token count matrices to recommend similar titles.
- **Tech Stack**: Python, Pandas ,Scikit-Learn and TMDB dataset.

TECHNICAL SKILLS

Languages:
Python, C++, Javascript, SQL, Typescript

Computer Networks:
OSI Model, TCP/ IP.

Frontend Development:
HTML, CSS, React, React Native, Tailwind, Tailwind, Bootstrap, Next.js, Nodejs.

BackendDevelopment:
Express.js,Node.js, MongoDB, Firebase, MySQL.

Others: WindowsBiometric Framework, Cryptography, Insomnia, Effective AI prompting.

EDUCATION

Electronics and Telecommunications Engineering

Jorhat Institute of Science and Technology
Aug'18 - Aug'22
Cumulative CGPA: 7.5

SOFT SKILLS

- Concept to code
- Voicing opinions
- Resource Utilisation
- Troubleshooting
- Active Observer
- Logical learner

RELEVANT COURSES

- Data Structure and Algorithm
- Computer Networks
- Problem Solving through Programming using C
- Advanced Programming
- Image Processing
- Machine Learning
- DBMS

CODING PRACTICES

GitHub | Leetcode | Freecodecamp