Team 83

Software Architecture and Design Document

INDEX

1. Introduction and Goals

Introduction

Trumio University is a standalone platform designed to address key challenges faced by both students and clients in the field of project collaboration and skill development. It features a diverse repository of courses created by Trumio and other contributors. The platform is integrated with an AI recommender system, including a Course Recommender and a News Flash section.

Key Features and Components:

- Course Recommender: An AI-based system that assists users in finding courses tailored to their skills, interests, and ongoing projects.
- News Flash: A news recommender system that keeps students updated with the latest industry trends and pushes news relevant to their ongoing projects.
- Client Recommender: An AI feature that recommends potential bidders to clients based on a star rating system, considering the experience and past projects of the bidders.

Goals

Project Goals:

- Upskilling Students: Provide students with access to a diverse range of certified courses aligning with their domain and project requirements.
- Relevant Material: Ensure ongoing projects are supported with relevant news and updates, improving the quality of project output.
- Client Confidence: Boost the confidence of clients by offering skilled candidates with verified certifications addressing concerns about students' technical and project management skills.

Course Recommender Goals:

- Tailored Recommendations: Provide personalised course recommendations based on skills, interests, career goals, and ongoing project requirements.
- Skill Enhancement: Ensure students possess the necessary skills to complete their projects successfully.

News Flash Goals:

- Daily Updates: Keep students informed with daily updates on trending and relevant industry news.
- Project-specific News: Push news related to the student's current projects, ensuring they stay informed about changes and opportunities.

Client Recommender Goals:

- Candidate Selection: Provide clients with a list of potential candidates for a project based on relevant experience.
- Trust Building: Implement a rating system based on previous projects and bidder experience to build trust between clients and bidders.

Benefits:

- Students: Gain access to certified courses, stay updated with industry trends, and increase career prospects.
- Clients: Receive higher-quality work output, confident in the skills and expertise of recommended students or bidders.
- Platform Trust: Establish trust between clients and bidders, ensuring projects are assigned to the most suitable candidates.

2. Stakeholders

1. **Clients:**

- Description: Individuals or organisations posting projects on the platform.
- Involvement: Engage in project collaboration and seek skilled students or bidders.

2. **Students/Bidders:**

- Description: Individuals seeking skill development, project collaboration opportunities, and actively participating in project bidding.
- Involvement: Enroll in courses, collaborate on projects, and bid for projects on behalf of themselves or other students.

3. **Platform Administrators:**

- Description: Individuals responsible for managing and maintaining the Trumio University platform.
- Involvement: Ensure smooth operation, address technical issues, and oversee platform activities.

3. Architecture Constraints

Hosting:

- On-Premise Design: The platform is tailored for on-premise use, restricting the adaptability of cloud-based solutions.
- Consideration: Architectural decisions should align with on-premise hosting, potentially impacting scalability and resource allocation dynamics.

Database Storage:

- MongoDB Adoption: MongoDB is the chosen database storage solution.
- Consideration: While MongoDB offers flexibility, managing large datasets, particularly embeddings.csv from AWS S3, may present challenges.

Resource Handling:

- Startup Impact: Retrieving embeddings.csv from AWS S3 can affect startup time.
- Consideration: Prioritize models with minimal resource requirements to streamline startup and maintain system efficiency.

Scalability:

- React and Flask Selection: Frontend and backend rely on React and Flask, facilitating straightforward scalability.
- Consideration: Leverage the scalability features of React and Flask during deployment to manage increased user loads effectively.

Power Efficiency:

- Low Power Models: The system is designed to prioritize low power consumption.
- Consideration: Continue the selection and optimization of models with an emphasis on resource efficiency, ensuring minimal impact on overall system performance.

4. Solution Strategy

The project employs a comprehensive solution strategy, leveraging AI for recommendation systems and integrating third-party APIs for enriched functionality.

Course Recommender:

- Data Collection: Used the dataset for udemy-courses from Kaggle to gather course data and obtained a CSV file, forming the primary dataset.
- Preprocessing: Conducted standard preprocessing, including text concatenation, contraction expansion, and lowercasing.

- Embedding Computation: Applied a pre-trained multilingual Universal Sentence Encoder ('distiluse-base-multilingual-cased-v1') to compute course embeddings.
- Storage Optimization: Stored all embeddings in a CSV file for efficient retrieval during inference.
- Tech Stack Processing: Applied the same preprocessing techniques to user-provided tech stacks, computed embeddings, and utilized cosine similarity for relevant course recommendations.
- Consistent Approach: Applied the same approach and model to recommend courses based on client-defined skills in the ticket-course recommender.

News Flash:

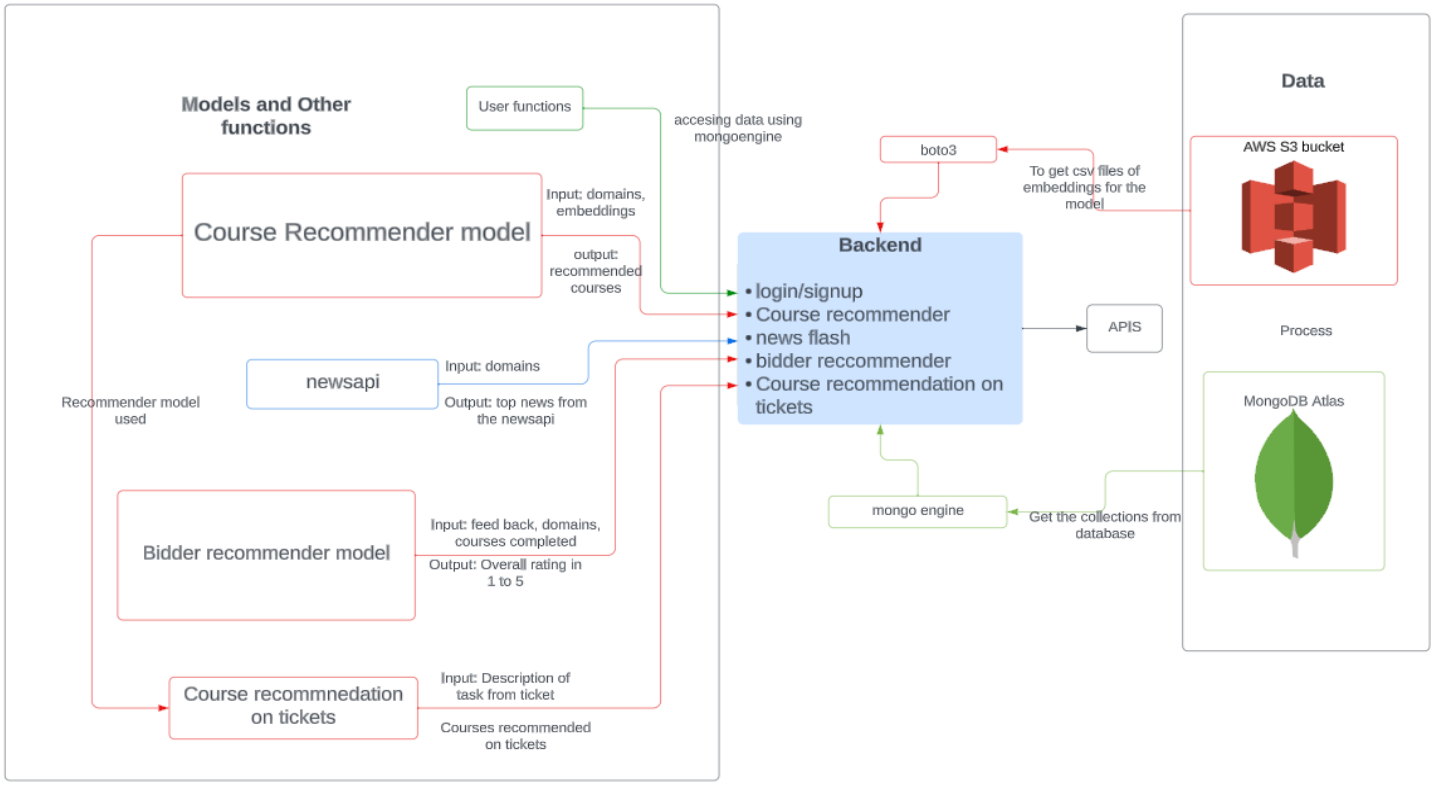
- API Integration: Employed NewsAPI to retrieve the latest news from 54 countries and various media sources.
- Methodology: Utilized the 'get_everything' method, performing a keyword search based on user-provided tech stacks.
- Filtering Criteria: Implemented language filtering (English) and a time cap of 1 week to display relevant news from the current week.

Bidder Recommendation to Clients:

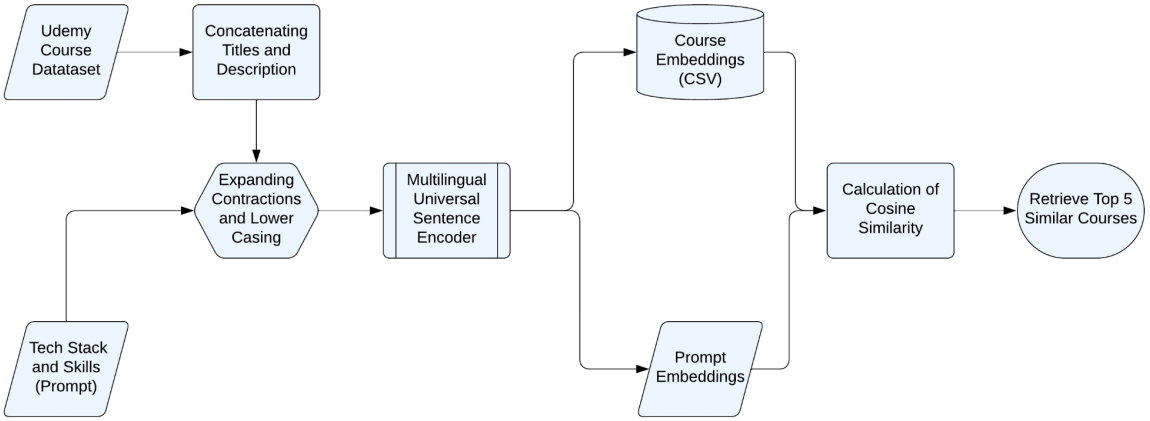
- Data Generation: Generated dummy data for students, including project history, client feedback, and completed courses.
- Model Selection: Implemented the BERTweet model, which is basically a RoBERTA model fine-tuned on English Tweets, for sentiment analysis.
- Scoring System: A SoftMax function was applied to sentiment scores, creating a discrete probability distribution.
- Rating System: Converted the 'positive' label probability into a star-based rating system (1 to 5), offering a comprehensive project suitability score.
- Information Presentation: Provided clients with details on completed projects, courses, and relevant badges for informed candidate selection.

5. Backend Architecture

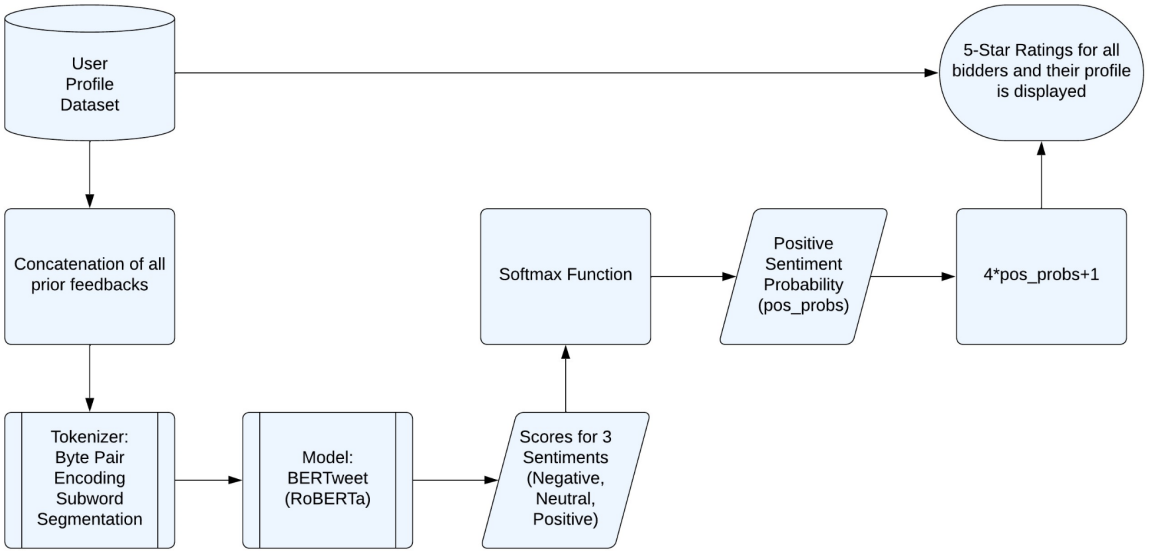
Overall architecture:



Course Recommender:



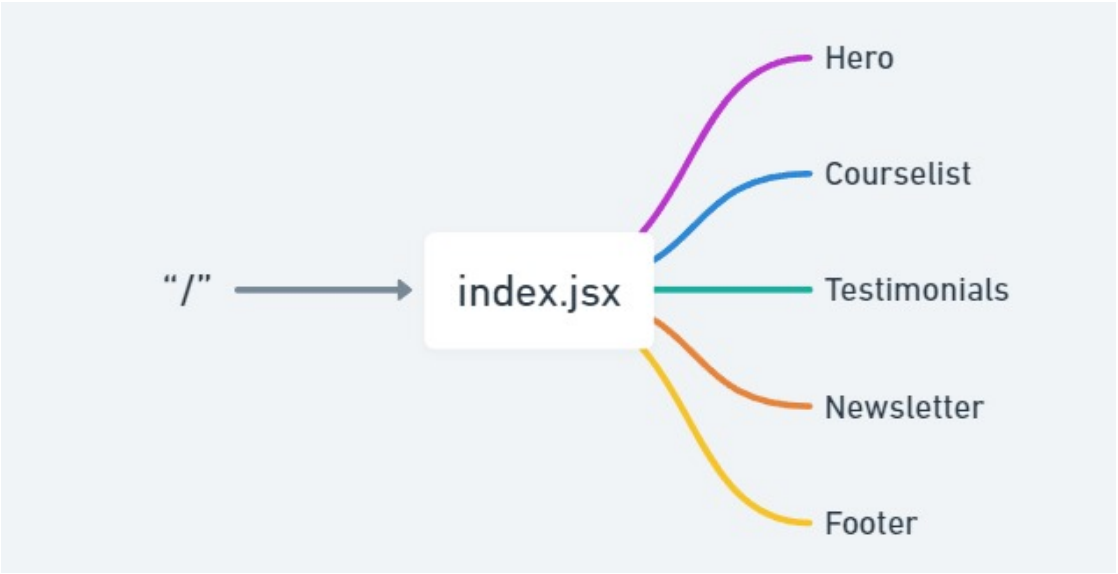
Bidder Recommender:

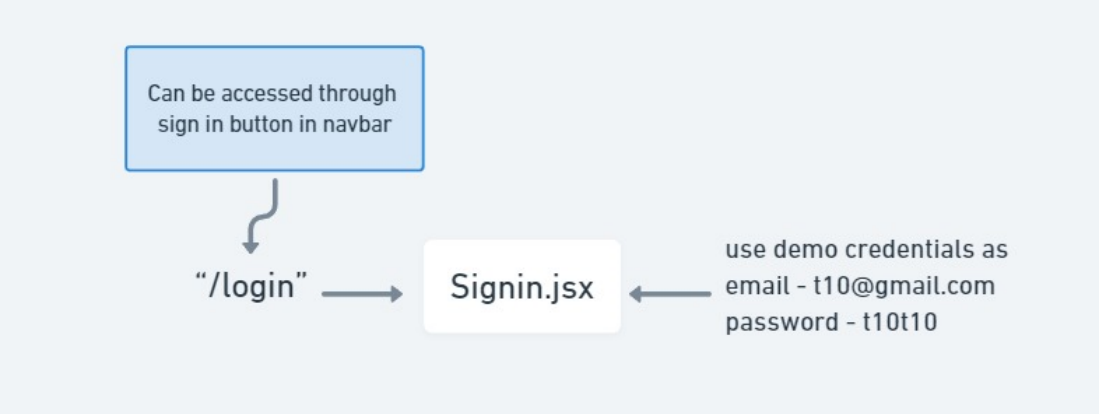


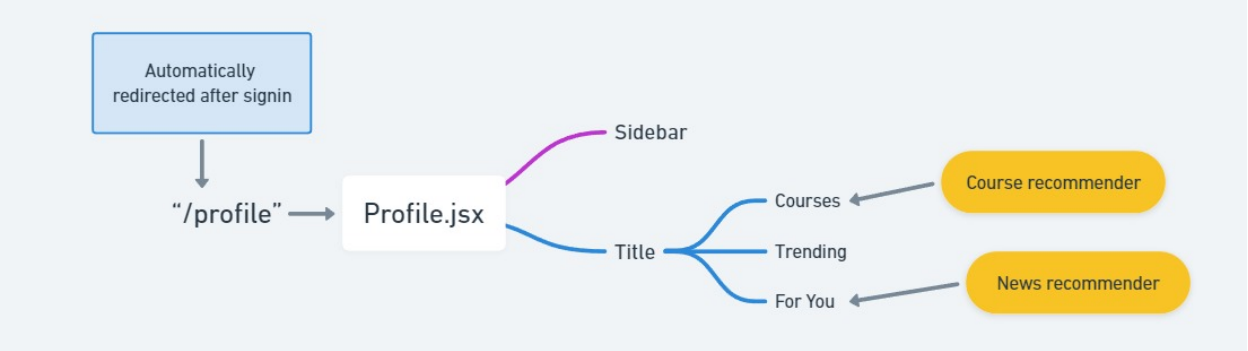
Dependencies:
Flask, flask-smorest, python-dotenv, pymongo, mongoengine, numpy, pandas, sentence-transformers==2.2.2, torch==2.1.0, flask_bcrypt, flask_jwt_extended, flask_cors, newsapi-python, boto3, requests==2.27.1

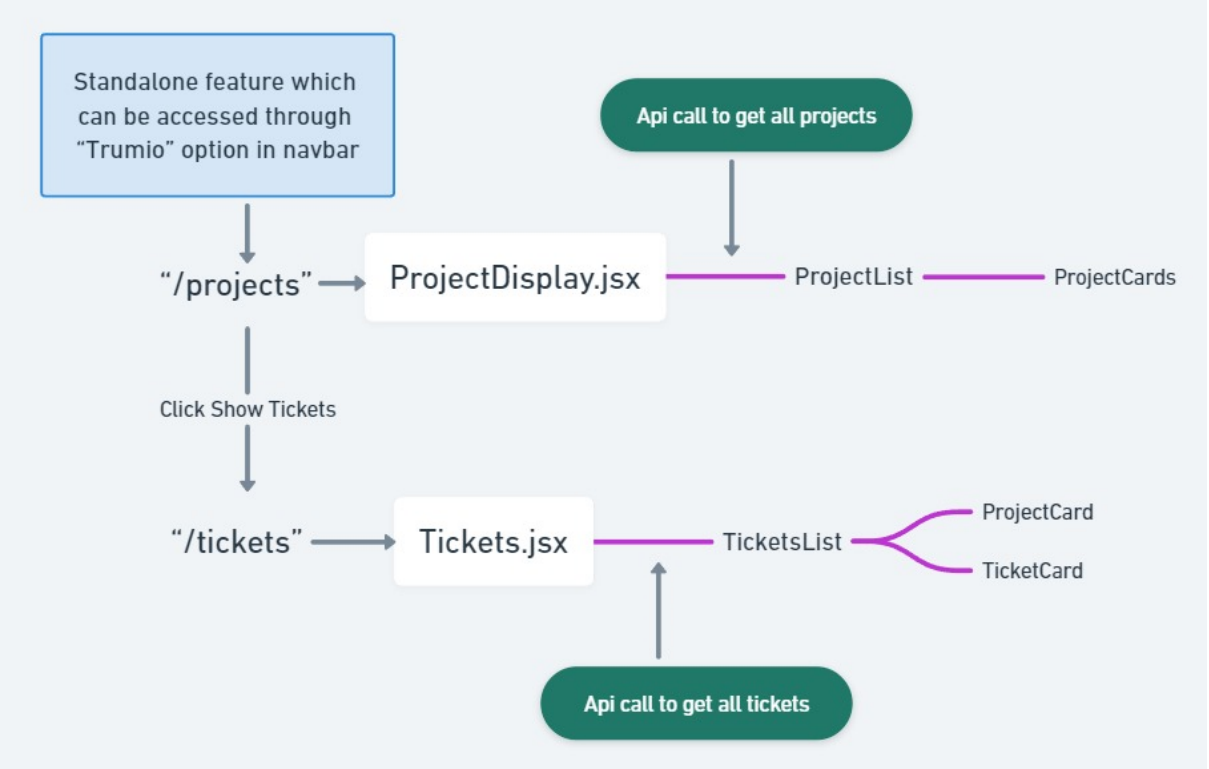
6. Frontend Architecture

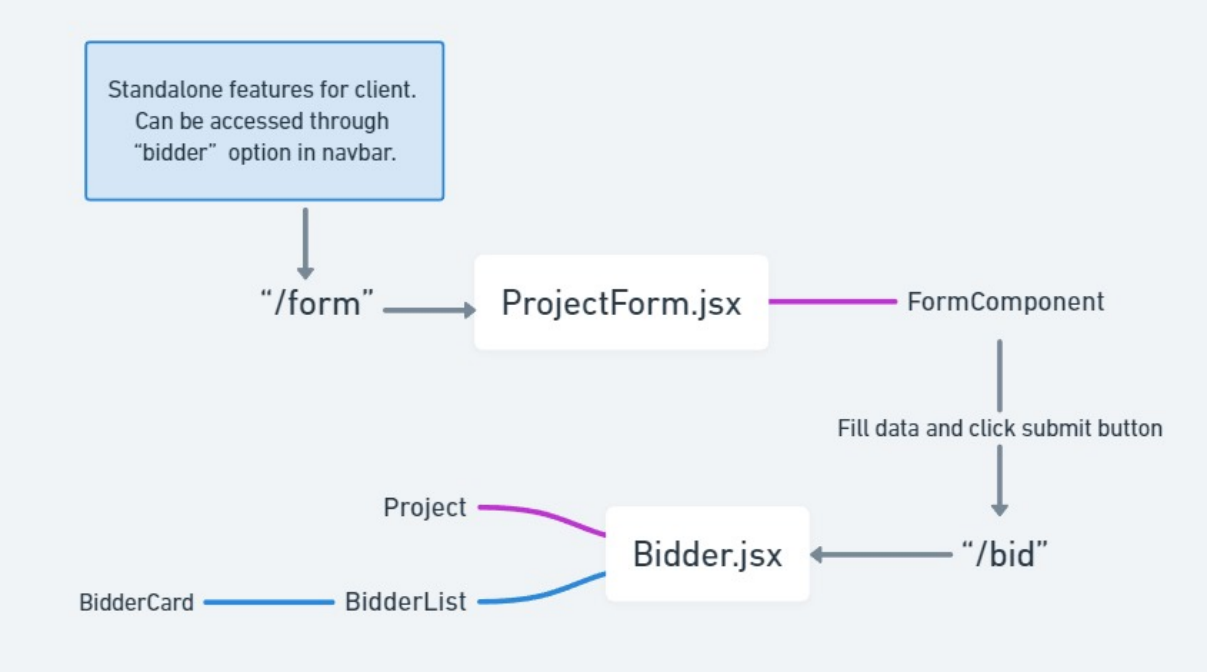
- App.js - Contains all routing used in the frontend.
- **Routes and Component structures**
  (flow chart of all routes and Pages folder components) -




Can be accessed through sign in button in navbar
"/login"
Signin.jsx
use demo credentials as
email - t10@gmail.com
password - t10t10


Automatically redirected after signin
"/profile"
Profile.jsx
Sidebar
Title
Courses
Trending
For You
Course recommender
News recommender


Standalone feature which can be accessed through "Trumio" option in navbar
Api call to get all projects
"/projects"
ProjectDisplay.jsx
ProjectList
ProjectCards
Click Show Tickets
"/tickets"
Tickets.jsx
TicketsList
ProjectCard
TicketCard
Api call to get all tickets



7. Glossary

- **Trumio University**:Standalone platform for project collaboration and skill development.
- **Course** Recommender:Recommends tailored courses based on skills and interests.
- **News Flash**: Recommender system for daily updates and project-specific news.
- **Client Recommender:** Recommends bidders to clients using a star rating system.
- **Clients**:Post projects and seek skilled students or bidders.
- **Students/Bidders**:Seek skill development and bid for projects.
- **Platform Administrators**:Manage and maintain Trumio University.
- **On-Premise Design:** Designed for on-premise use.
- **MongoDB Adoption:** Chosen as the database storage solution.
- **Startup Impact:**Retrieving embeddings.csv can impact startup time.
- **Scalability:** React and Flask for scalable frontend and backend.
- **Power Efficiency**:Models designed for low power consumption.
- **Data Collection (Course Recommender)**:Used Kaggle dataset, standard text preprocessing.
- **Embedding Computation (Course Recommender)**: Used a pre-trained multilingual Universal Sentence Encoder.
- **API Integration (News Flash)**:Employed NewsAPI for global news.
- **Data Generation (Bidder Recommendation)**: Generated dummy data for students.
- **Model Selection (Bidder Recommendation)**: Implemented BERTweet model for sentiment analysis.
- **Information Presentation (Bidder Recommendation)**:Provides clients with relevant information for candidate selection.