PLAN

E:\3rd year\_2nd sem\Application of ML in Industries\Tango\

│

├── app/

│ ├── \_\_init\_\_.py

│ ├── routes.py

│ ├── models/

│ │ └── \_\_init\_\_.py

│ ├── services/

│ │ ├── \_\_init\_\_.py

│ │ └── youtube\_analyzer.py

│ ├── templates/

│ │ ├── index.html

│ │ └── results.html

│ └── static/

│ ├── css/

│ │ └── styles.css

│ ├── js/

│ │ └── script.js

│ └── images/

│ └── logo.jpeg

│

├── config.py

├── requirements.txt

├── run.py

└── README.md

Requirements

# Flask and Web Development

Flask==2.3.2

Flask-SQLAlchemy==3.0.5

Flask-WTF==1.1.1

Flask-Login==0.6.2

Flask-Migrate==4.0.4

Flask-Cors==4.0.0

# YouTube Video Processing

pytube==15.0.0

youtube-transcript-api==0.6.1

google-api-python-client==2.95.0

google-auth==2.22.0

google-auth-oauthlib==1.0.0

google-auth-httplib2==0.1.0

# Data Processing and Analysis

pandas==2.0.3

numpy==1.24.3

# Machine Learning and NLP

scikit-learn==1.3.0

nltk==3.8.1

textblob==0.17.1

transformers==4.30.2

torch==2.0.1

# Web Scraping and HTTP Requests

requests==2.31.0

beautifulsoup4==4.12.2

# Utilities

python-dotenv==1.0.0

tqdm==4.65.0

Details

Flask and Web Development

Flask: The core web framework for building the platform.

Flask-SQLAlchemy: For database integration and management.

Flask-WTF: For handling forms (e.g., user input for YouTube links).

Flask-Login: For user authentication and session management.

Flask-Migrate: For database migrations (if you plan to use a database).

Flask-Cors: For handling Cross-Origin Resource Sharing (CORS) if your platform interacts with external APIs.

YouTube Video Processing

pytube: To download YouTube videos and extract metadata (e.g., title, description, views, etc.).

youtube-transcript-api: To fetch video transcripts for analysis.

google-api-python-client: To interact with the YouTube Data API for advanced metadata and analytics.

google-auth, google-auth-oauthlib, google-auth-httplib2: For authenticating with Google APIs.

Data Processing and Analysis

pandas: For data manipulation and analysis (e.g., processing video metadata or comments).

numpy: For numerical computations and data processing.

Machine Learning and NLP

scikit-learn: For machine learning tasks (e.g., fake video detection).

nltk: For natural language processing (e.g., analyzing comments or video descriptions).

textblob: For sentiment analysis of comments or video descriptions.

transformers: For advanced NLP tasks (e.g., fake news detection, content analysis).

torch: Required for using transformers (PyTorch backend).

Web Scraping and HTTP Requests

requests: For making HTTP requests (e.g., fetching external data).

beautifulsoup4: For web scraping (if needed for additional data collection).

Utilities

python-dotenv: For managing environment variables (e.g., API keys).

tqdm: For progress bars during data processing or downloads.