Project Synopsis

AI-Powered News Summarizer Portal

Problem Statement:

With the exponential growth of online news and information, students often face **information overload** while trying to stay updated. Reading lengthy articles consumes significant time, and manually filtering key insights is inefficient. There is a growing need for a platform that delivers **concise, accurate, and reliable news summaries**, enabling students to consume information quickly and effectively.

Proposed Solution:

The AI-Powered News Summarizer Portal is a **web-based platform** that utilizes state-of-the-art **Natural Language Processing (NLP) models** to generate concise summaries of lengthy news articles. The backend is powered by **Django**, managing user authentication, news storage, and summarization APIs. The frontend, built with **React**, provides students with a clean, user-friendly interface to browse, read, save, and share AI-generated news summaries. The summarization is driven by advanced **transformer-based models** such as BERT or Pegasus, ensuring high-quality summaries that retain contextual meaning.

Objectives:

- To help students **save time** by providing concise, AI-driven summaries of news articles.
- To build a platform where students can **store, organize, and share** summarized articles.
- To enhance **information accessibility** by simplifying complex or lengthy content.
- To showcase the application of **transformer-based NLP models** in real-world information management.

Methodology:

- **Backend (Django):**
  - USER authentication system (login/signup) with role-based access.
  - REST APIs for fetching articles, generating summaries, and managing user preferences.
  - Integration with external news APIs (e.g., NewsAPI.org) to fetch live articles.
- **AI Summarization Engine:**
  - Use retrained transformer models such as **BERT (extractive summarization)** or **Pegasus (abstractive summarization)**.
  - Implement text preprocessing, tokenization, and summary generation pipelines.
  - Optimize model inference for scalability (via Hugging Face Transformers or TensorFlow).
- **Frontend (React):**
  - Build a responsive UI for browsing articles and reading AI-generated summaries.
  - Enable features for saving, bookmarking, and sharing summaries.
- **Database (MongoDB + MySQL)**:
  - Store user data and saved summaries.

Expected Outcome:

- A **fully functional AI-driven news portal that generates summaries with high accuracy and contextual relevance.**
- **Time-efficient information consumption for students and researchers.**
- Features that allow users to **save, organize, and share** news summaries seamlessly.