**AI-Driven News Aggregator**

**Description**: The AI-Driven News Aggregator will collect news articles from various sources, categorize them into different topics (e.g., technology, sports, politics), and summarize the content using AI techniques. This project will provide users with concise and categorized news, saving them time by delivering only the most relevant information.

**Features:**

1. **News Collection**:
   * Implement a web scraper to gather news articles from multiple sources.
   * Use APIs like NewsAPI or RSS feeds for structured news data.
2. **Categorization**:
   * Categorize news articles into predefined categories such as technology, sports, politics, etc.
3. **Summarization**:
   * Generate concise summaries of the news articles using NLP models.
   * Implement extractive or abstractive summarization.
4. **User Interface**:
   * Provide a simple and intuitive web interface for users to browse news by categories.
   * Enable users to search for specific news topics.
5. **User Personalization**:
   * Allow users to customize their feed based on preferred categories or keywords.
   * Use machine learning to recommend articles based on user preferences and behavior.

**Tech Stack:**

1. **Backend**:
   * **Spring Boot**: For creating RESTful APIs and handling the backend logic.
   * **Web Scraper**: Use libraries like JSoup (for Java) or BeautifulSoup (for Python) to scrape news websites.
2. **NLP Libraries**:
   * **SpaCy** or **NLTK** for text processing.
   * **Hugging Face Transformers** for implementing advanced NLP models for summarization and categorization.
   * **Scikit-learn** or **TensorFlow** for building and deploying classification models.
3. **Database**:
   * **MySQL** for storing news articles, user preferences, and other metadata.
4. **Frontend**:
   * **ReactJS** or **Angular** for building the user interface.
   * **Bootstrap** or **Material-UI** for styling and layout.
5. **Deployment**:
   * Use **Docker** for containerization.
   * Deploy the application on cloud platforms like **AWS**, **Heroku**, or **Google Cloud**.

**Steps to Develop:**

1. **Setup the Project**:
   * Create a new Spring Boot project with necessary dependencies (Spring Web, Spring Data JPA, etc.).
2. **Implement Web Scraper**:
   * Write a service in Spring Boot that uses JSoup to scrape news articles from selected websites.
   * Store the scraped articles in the database.
3. **Text Categorization**:
   * Preprocess the text data using NLP techniques.
   * Train a classification model using libraries like Scikit-learn or TensorFlow to categorize the articles.
   * Implement this model in the backend for real-time categorization.
4. **Text Summarization**:
   * Use pre-trained NLP models for summarization (e.g., BERT, T5).
   * Integrate the summarization service with the backend to process the articles.
5. **Build REST APIs**:
   * Develop RESTful APIs for fetching categorized and summarized news articles.
   * Implement endpoints for user preferences and search functionality.
6. **Frontend Development**:
   * Build the frontend with ReactJS or Angular.
   * Connect the frontend to the backend using the REST APIs.
7. **User Personalization**:
   * Implement a recommendation system that suggests articles based on user preferences.
   * Store user preferences and interaction data in the database.
8. **Testing and Deployment**:
   * Test the application thoroughly for performance and accuracy.
   * Use Docker to containerize the application and deploy it on a cloud platform.

**Potential Challenges:**

* Handling large volumes of data from multiple sources.
* Ensuring the accuracy and relevance of categorization and summarization.
* Maintaining real-time updates and scalability.

**Step 7: Implement a summarization algorithm.**

* **Use Hugging Face Transformers:** This library provides access to pre-trained summarization models like BART, T5, and Pegasus.
* **Other libraries:** Consider libraries like Gensim or Sumy, which offer various summarization techniques (e.g., extractive summarization, abstractive summarization).

### Project Description Key Features of the Website:

1. **News Categorization**:
   * Users can browse news articles categorized into different topics like Technology, Sports, Politics, etc.
2. **Keyword Search**:
   * A search bar allows users to find news articles by keywords.
3. **Summarization Feature**:
   * Users can view summarized versions of the articles.
   * The summary can be displayed in a side panel or a popup when the user clicks on a "Summarize" button.
4. **User Authentication**:
   * Users can register and log in.
   * Logged-in users can save their preferences, such as favorite categories or keywords.
5. **Personalized Feed**:
   * Based on saved preferences, users will receive a customized news feed highlighting their preferred categories.
6. **Responsive Design**:
   * The website will be mobile-friendly and work well on various devices.