**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**

**TITLE: Clustering-Based Book Recommender System Using AI/ML**

**Team members:**

**K.Lakshmipriya-2320030453 P. Sree Haneeka-2320030326**

**Abstract:**

The rapid growth of online information has necessitated the development of recommender systems to filter and present relevant content to users. This project focuses on creating a book recommender system that leverages clustering algorithms to group similar books and provide personalized recommendations. The system will utilize unsupervised machine learning techniques, such as K-means clustering, to categorize books based on multiple features like genre, author, and user ratings. By matching users to relevant book clusters, the system aims to enhance recommendation accuracy and improve user experience.

**Project Implementation Overview:**

1. **Data Collection**:  
   Gather datasets that contain information on books, including metadata like titles, authors, genres, publication dates, and user ratings. Sources like Goodreads or public datasets can be used.
2. **Data Preprocessing**:  
   Clean the data to handle missing values, normalize numerical data (e.g., ratings), and encode categorical variables (e.g., genre, author). Text data, such as book descriptions, may also be vectorized using techniques like **TF-IDF** if content-based recommendations are involved.
3. **Feature Engineering**:  
   Extract key features from the dataset to create a feature matrix. These features might include book popularity, genre, rating scores, or user reading behavior.
4. **Clustering Algorithm Implementation**:
   * Use the **K-means clustering** algorithm to group books based on the similarity of their features. The number of clusters (K) can be determined through the **Elbow method** or **Silhouette score**.
   * After clustering, the books within each group will share common attributes, making them easier to recommend to users with similar tastes.
5. **Recommendation Logic**:
   * When a user searches for or interacts with a book, the system assigns them to the most relevant cluster.
   * The system will then recommend books from that cluster to the user.
   * For users with insufficient interaction data, the system could rely on popularity-based or genre-based recommendations.
6. **Evaluation & Optimization**:
   * Use evaluation metrics like **Precision, Recall, F1-score**, and **Mean Average Precision (MAP)** to assess the accuracy and relevance of the recommendations.
   * Optimize the model by fine-tuning clustering parameters and adjusting features based on user feedback.
7. **Deployment**:  
   Deploy the recommendation system using a suitable framework, such as **Flask** or **Django**, with a web interface for users to interact with the system and receive book suggestions.

**Project Prerequisites:**

1. **Knowledge of Machine Learning**:.
   * Experience with **feature engineering** and **dimensionality reduction** techniques (e.g., PCA).
2. **Programming Skills**:
   * Proficiency in **Python**.
   * Familiarity with libraries such as **scikit-learn** (for machine learning), **pandas** (for data manipulation), **NumPy** (for numerical operations), and **Matplotlib/Seaborn** (for data visualization).
3. **Data Preprocessing Techniques**:
   * Handling missing data, encoding categorical variables, and normalizing numerical data.
   * Text vectorization techniques like **TF-IDF** if considering book descriptions.
4. **Familiarity with Recommender Systems**:
   * Understanding of recommendation strategies like **content-based filtering** and **collaborative filtering**.
   * Basics of **distance measures** (e.g., Euclidean, Cosine similarity) used in clustering.
5. **Web Development Basics (Optional)**:
   * If deploying the system, knowledge of frameworks like **Flask** or **Django**.
   * Understanding of APIs and front-end integration for user interaction.
6. **Development Environment**:
   * Set up of an IDE (e.g., **Jupyter Notebook**, **PyCharm**) and necessary Python libraries (scikit-learn, pandas, etc.).
   * Use of version control tools like **Git** for collaboration and code management.