Spam Detection – Project Report

1. Project Overview

The goal of this project is to build a **machine learning model** that can detect **spam messages** from non-spam (ham) messages. It involves natural language processing (NLP) for text preprocessing and classification using supervised learning algorithms.

---

2. Data Preprocessing

- **Dataset:** The dataset used includes labeled SMS messages as either spam or ham.
- **Steps Taken:**
    - Imported the dataset using pandas.
    - Checked for null/missing values.
    - Added a new column to convert the target labels into numerical format: ham → 0, spam → 1.
    - Cleaned the text by:
        - Removing punctuations
        - Lowercasing
        - Removing stop words
        - Tokenization and stemming using NLTK

---

3. Feature Extraction

- Utilized **CountVectorizer** to transform the cleaned text into a numerical format.
- Split the dataset into **training** and **testing** sets using train_test_split.

---

4. Model Building & Training

Multiple models were trained for comparison:

- **Multinomial Naive Bayes (MNB)**
- **Support Vector Machine (SVM)**
- **Random Forest Classifier**
- **Logistic Regression**

Each model was trained using the vectorized features and evaluated on test data.

---

5. Model Evaluation

- Metrics used: **Accuracy**, **Confusion Matrix**, **Precision**, **Recall**, and **F1 Score**.

- Results showed **Naive Bayes** performing quite well due to its effectiveness in text classification problems.

---

6. Conclusion

- The project demonstrates a classic NLP problem pipeline — from cleaning raw text to evaluating machine learning models.
- **Naive Bayes** is a simple yet powerful model for spam detection.
- With more data or advanced NLP methods (e.g., TF-IDF, deep learning), performance may further improve.