📱 SMS Spam Detector - README

# 📌 Project Overview

Spam Guardian is a machine learning-based web application developed using Streamlit. It classifies SMS and email messages as spam or not spam using natural language processing (NLP) and a trained classification model. The application highlights message characteristics and offers safety recommendations based on the prediction.

# 🚀 Features

* • Streamlit-powered web interface for real-time predictions
* • Custom CSS styling for enhanced user experience
* • Message analysis: word count, character count, exclamations, URL count
* • Probability prediction and spam confidence score
* • Visual indicators and recommendations based on analysis
* • Detection of suspicious patterns like urgency, financial terms, contact requests, etc.

# 📁 Project Structure

├── mail\_data.csv → Dataset file  
  
├── model.pkl → Trained classification model  
  
├── vectorizer.pkl → TF-IDF vectorizer for text  
  
├── sms-Detection.ipynb → Jupyter Notebook (EDA, training, evaluation)  
  
├── Sms-app.py → Streamlit app script  
  
└── README.docx → This documentation file

# ⚙️ Installation & Setup

Follow the steps below to set up and run the project locally:

* • Clone the Repository:

git clone https://github.com/your-username/sms-spam-detector.git

* • Navigate to the Project Directory:

cd sms-spam-detector

* • (Optional) Create a Virtual Environment:

python -m venv venv  
source venv/bin/activate # macOS/Linux  
venv\Scripts\activate # Windows

* • Install Dependencies:

pip install -r requirements.txt

* • Run the Streamlit App:

streamlit run Sms-app.py

# 🧾 Dataset

The project uses the SMS Spam Collection dataset, which contains labeled SMS messages as 'spam' or 'ham'. The dataset is loaded from 'mail\_data.csv' and processed during model training.

# 🧠 Model & Techniques

The model uses the following techniques and tools:

* • Natural Language Processing: tokenization, stop word removal, stemming
* • Vectorization using TF-IDF
* • Model: Naive Bayes Classifier
* • Evaluation metrics: accuracy, confusion matrix, precision, recall

# ⚙️ How It Works

1. User inputs a message into the text box.  
2. The message is cleaned (lowercased, tokenized, stop words removed, stemmed).  
3. Transformed text is vectorized using TF-IDF.  
4. Model predicts whether it's spam or not, and displays the confidence.  
5. Visual indicators and tips are shown based on the analysis.