TECHNICAL DOCUMENTATION

Title: Customer Churn Prediction using Machine Learning

1. Project Overview and Objectives

Every telecom company faces the problem of losing customers, which reduces revenue and increases marketing costs.
The objective of this project is to **predict which customers are likely to churn** so that the companies can take proactive actions to retain them.

Key Goals:

- Analyze customer behavior and understand factors leading to churn.
- Build a machine learning model to predict churn.
- Deploy the model as a web application for easy usage.
- Provide actionable business recommendations based on model insights.

2. Setup and Installation Instructions

1. **Save the important project files**

   `app.py` → Flask application
   `model.pkl` → Trained model
   `scaler.pkl` → MinMax Scaler
   `pca.pkl` → PCA transformation
   `templates/` → HTML files like index.html and result.html

2. **Install required Python packages:**

   pip install flask numpy pandas scikit-learn matplotlib.pyplot pickle

3. **Run the Flask app:**

   python app.py

4. **Enter the customer details and check prediction:**

   Enter customer details include:

   - **Contract Type**(Month-to-Month/One-year/Two-year)
   - **Payment Method(**Electronic-Check/Credit-Card/Bank-Transfer)
   - **Paperless Billing**(Yes/No)
   - **Senior Citizen**(Yes/No)
   - **Tenure**(Months)

- **Monthly Charges**(Amount)
- **Total Charges**(Amount)

3. Code Structure Explanation

The project code is organized in a way that follows the complete data science workflow:

1. **Data Loading:**

   Load the whole csv file into the Dataframe**.**

2. **Data Understanding:**

   Perform EDA and Analysis Every Feature of the Dataset and Also Find the Relationship between them.

3. **Data Preprocessing and Feature engineering:**

   Handles missing values, encodes categorical features, scales numeric features, applies PCA.

4. **Training and Tunning:**

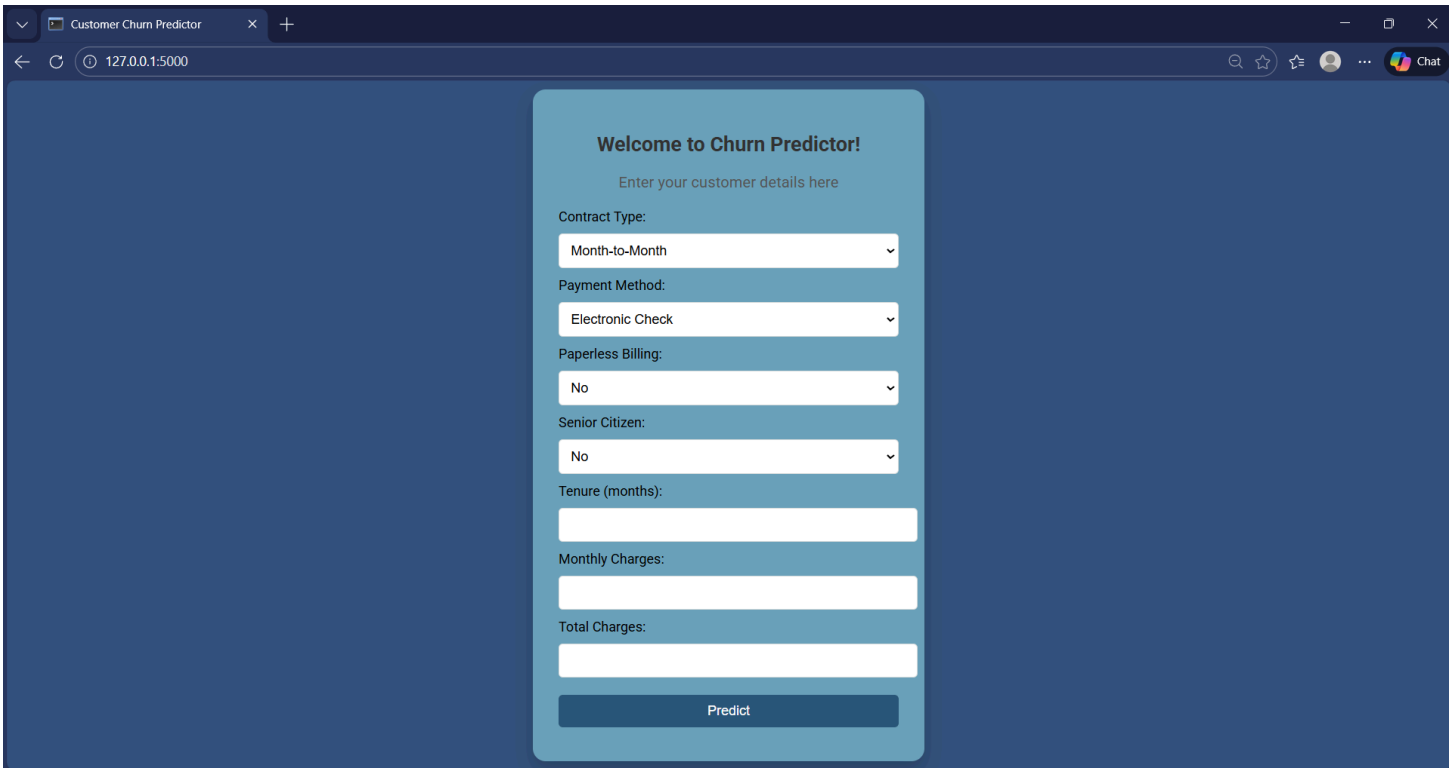
   Trains the model, tunes hyperparameters, saves the model, scaler, and PCA objects.

5. **Deployment (Flask Web App):**
   - `app.py` → Main application file.
   - `/` → Homepage: form to enter customer data.
   - `/predict` → Receives input, scales, applies PCA, predicts churn, shows result.

6. **Templates Files:**
   - `templates/index.html` → Homepage with dropdowns and input fields.
   - `templates/result.html` → Shows prediction results with user-friendly values.

4. Screenshots of Working Application

Website is looking like :



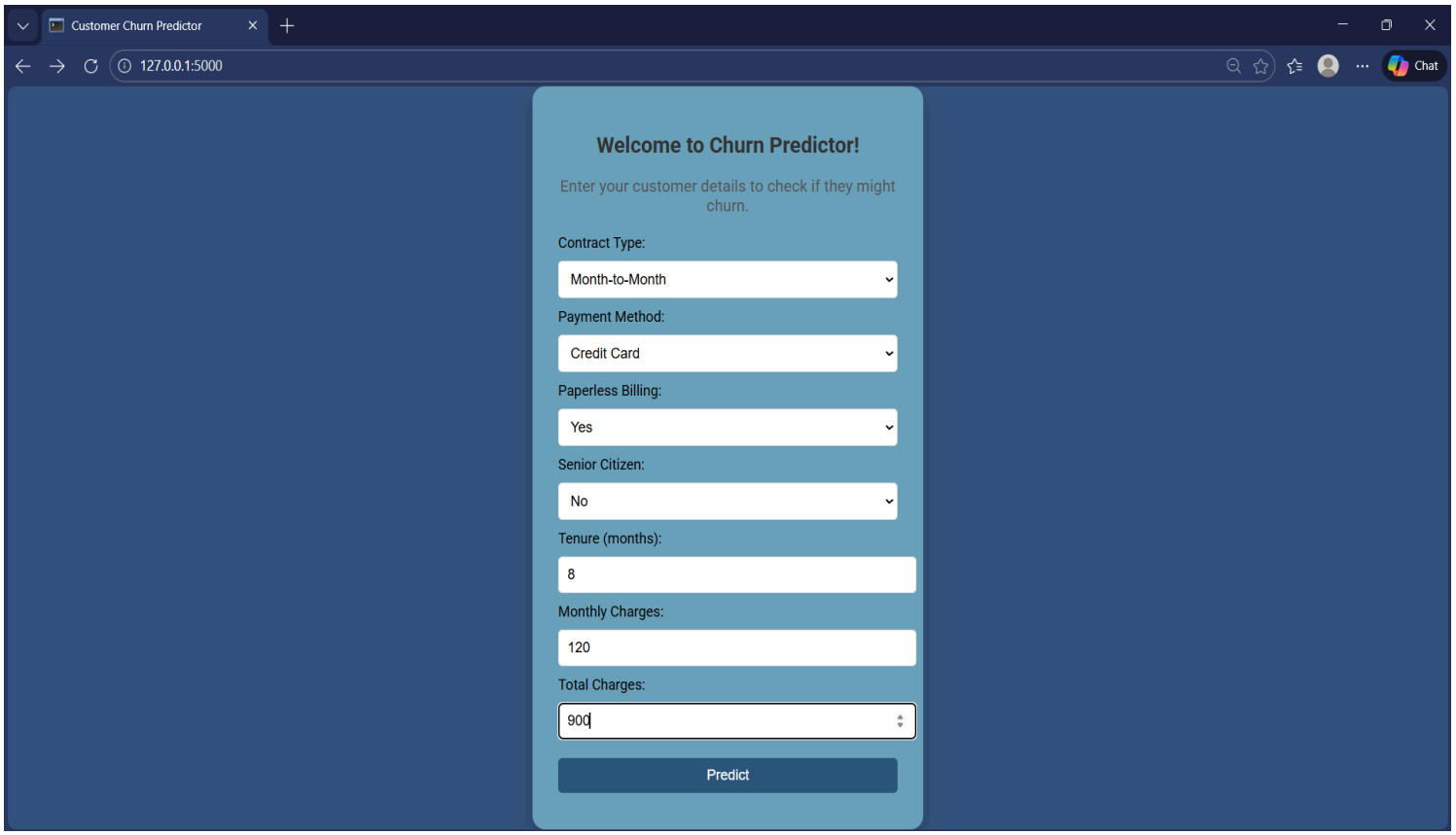
5. Explanation of How Technical Requirements are Met

Requirement	How It Is Met
Data Preprocessing	Handled missing values, encoded categorical features, scaled numeric features, applied PCA
Model Training	Used Decision Tree with hyperparameter tuning
Model Evaluation	Accuracy score, confusion matrix calculated and verified
Deployment	Flask app created with interactive form and result page
User-Friendly Input	Dropdowns and input fields with numeric mapping explained
Complete Workflow	From data collection → preprocessing → modeling → deployment → business insights

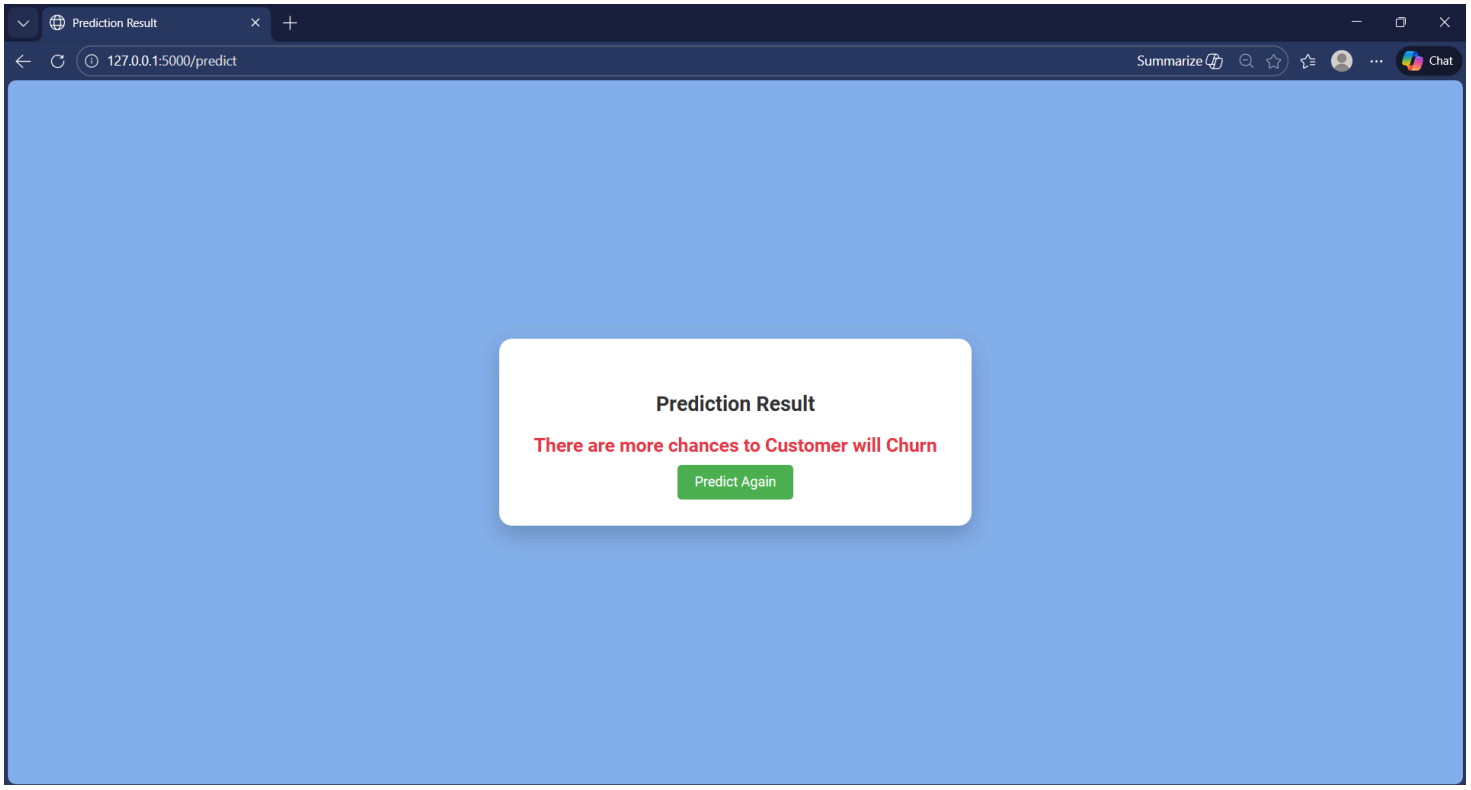
6. Testing Evidence by entering Customer details

Examples of test cases and validation

Homepage:



Result:



Testing Evid