PHASE-2

ProjEct titlE: PrEdicting cuStomEr cHurn uSing mAcHinE lEArning to uncovEr HiddEn PAttErnS

**Student Name:** Manogaran.T

**Register Number:** 511723205003

**Institution:** Pallavan college of Engineering

**Department:** INFORMATION TECHNOLOGY

**Date of Submission:** 27/04/2025

**Github Link**:[: **https://github.com/manogaran0309/naanmudhalvan**](https://github.com/manogaran0309/naanmudhalvan)

# 1. Problem Statement

Customer churn, or the loss of customers over time, is a critical issue for businesses in competitive industries. Retaining existing customers is often more cost-effective than acquiring new ones. However, identifying which customers are likely to churn can be difficult due to the complexity and volume of customer behavior data.

This project aims to build a machine learning-based predictive system that identifies customers likely to churn based on their behavior, transaction history, demographics, and service usage patterns. The goal is to provide early warning indicators so that businesses can take proactive measures to improve retention and customer satisfaction.

# 2. Project Objectives

Develop a machine learning model to predict customer churn.

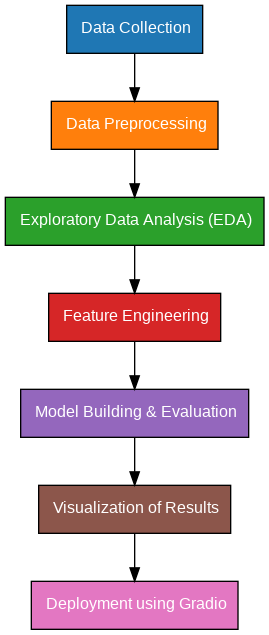
Identify key behavioral and demographic features influencing churn.

Provide actionable business insights to reduce churn rate.

Ensure model accuracy, interpretability, and business usability.

Deploy the model using a user-friendly interface (e.g., Streamlit or Flask).

# 3. Flowchart of the Project Workflow



# Data Description

**Dataset Name: Telco Customer Churn Dataset (or similar business dataset)**

**Source: Kaggle / Company CRM**

**Type of Data: Structured tabular data**

**Records and Features: ~7,000+ records, with 20-30 features**

**Target Variable: Churn (Yes/No) Static or Dynamic: Static dataset**

**Attributes Covered:**

**Customer Demographics: Gender, Age, SeniorCitizen, etc.**

**Service Usage: Internet, Phone, Streaming, etc.**

**Account Info: Tenure, Contract type, Monthly charges, Total charges**

**Target: Churn (Yes or No)**

**Dataset Link: https://www.kaggle.com/datasets/blastchar/telco-customer-churn**

# Data Preprocessing

Checked for missing/null values.

Encoded categorical variables using Label Encoding and One-Hot Encoding.

Scaled numerical features using StandardScaler.

Handled class imbalance using SMOTE (Synthetic Minority Oversampling Technique). Removed outliers based on Z-scores and boxplots.

# Exploratory Data Analysis (EDA)

Univariate Analysis:

Distribution plots for churned vs. retained customers Bar plots for contract type, tenure, and payment method Bivariate/Multivariate Analysis:

Heatmap showing correlation between features

Boxplots comparing churn vs. non-churn across various metrics Key Insights:

Month-to-month contract customers are more likely to churn.

Customers with high monthly charges and short tenure show high churn rate. Electronic check payment users are more prone to churn.

# Feature Engineering

Created new features: e.g., Average charges per tenure, Service count

Reduced multicollinearity by dropping highly correlated features

Performed feature selection using Recursive Feature Elimination (RFE) and feature importance rankings

# Model Building

Algorithms Used:

Logistic Regression

Random Forest

XGBoost

Support Vector Machine (SVM)

Train-Test Split: 80% training, 20% testing Evaluation Metrics:

Accuracy

Precision, Recall, F1-Score

Confusion Matrix

ROC-AUC Score

Model Selection Rationale:

Logistic Regression: Easy interpretability and baseline.

Random Forest & XGBoost: High accuracy, handles feature interactions well. SVM: Effective with high-dimensional space and limited samples.

# Visualization of Results & Model Insights

ROC Curves for model comparison

Feature importance charts from tree-based models

Confusion matrix visualization

Churn probability histogram

Dashboard displaying churn insights for management

# 10. Tools and Technologies Used

Programming Language: Python 3

Data Handling & EDA: pandas, numpy, matplotlib, seaborn, plotly

Modeling: scikit-learn, xgboost, imbalanced-learn

Deployment: Streamlit or Flask

Development Environment: Google Colab / Jupyter Notebook

Version Control: Git, GitHub

# 11. Team Members and Contributions

|  |  |
| --- | --- |
| **Team members** | **contribution** |
| KESAVAN .S | Data Collection and Preprocessing |
| MANOGARAN .T | Exploratory Data Analysis |
| BOOPATHI .M.K | Feature Engineering & Model  Development |
| SATHISH KUMAR .S | Evaluation and Deployment |
| HEMANATHAESWARAN .S | Documentation and Reporting |