Phase-3 Submission Template

Student Name: Kanimozhi.D

Register Number: 422723104056

Institution: VRS college of Engineering and Technology

Department: Computer Science and Engineering

Date of Submission: 17.05.2025

Github Repository Link: https://github.com/Kani-123-colab/Kani.git

---

1. Problem Statement

Customer churn refers to the loss of clients or customers, a critical issue for businesses, especially in subscription-based models like telecom, banking, and SaaS. Understanding and predicting churn allows companies to implement retention strategies, thereby reducing revenue loss and improving customer satisfaction.GitHub

Business Relevance:
High churn rates can significantly impact profitability. Predicting churn enables businesses to proactively address issues, personalize customer experiences, and optimize marketing efforts.

Problem Type:
This is a **binary classification** problem where the goal is to predict whether a customer will churn (Yes/No).

2. Abstract

This project aims to develop a machine learning model to predict customer churn using the Telco Customer Churn dataset. The objective is to identify at-risk customers and provide actionable insights for retention

strategies. The project involves data preprocessing, exploratory data analysis (EDA), feature engineering, model building, evaluation, and deployment. Multiple models, including Logistic Regression, Random Forest, and XGBoost, are trained and evaluated. The final model is deployed using Streamlit for real-time predictions. The outcome is a user-friendly application that assists businesses in reducing churn through informed decision-making

3. System Requirements

Hardware:

- Minimum 8 GB RAM
- Intel i5 processor or equivalent

Software:

- Python 3.8+
- Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, xgboost, imbalanced-learn, streamlit
- IDE: Jupyter Notebook or Google Colab

4. Objectives

 Primary Goal: Develop a predictive model to forecast customer churn.

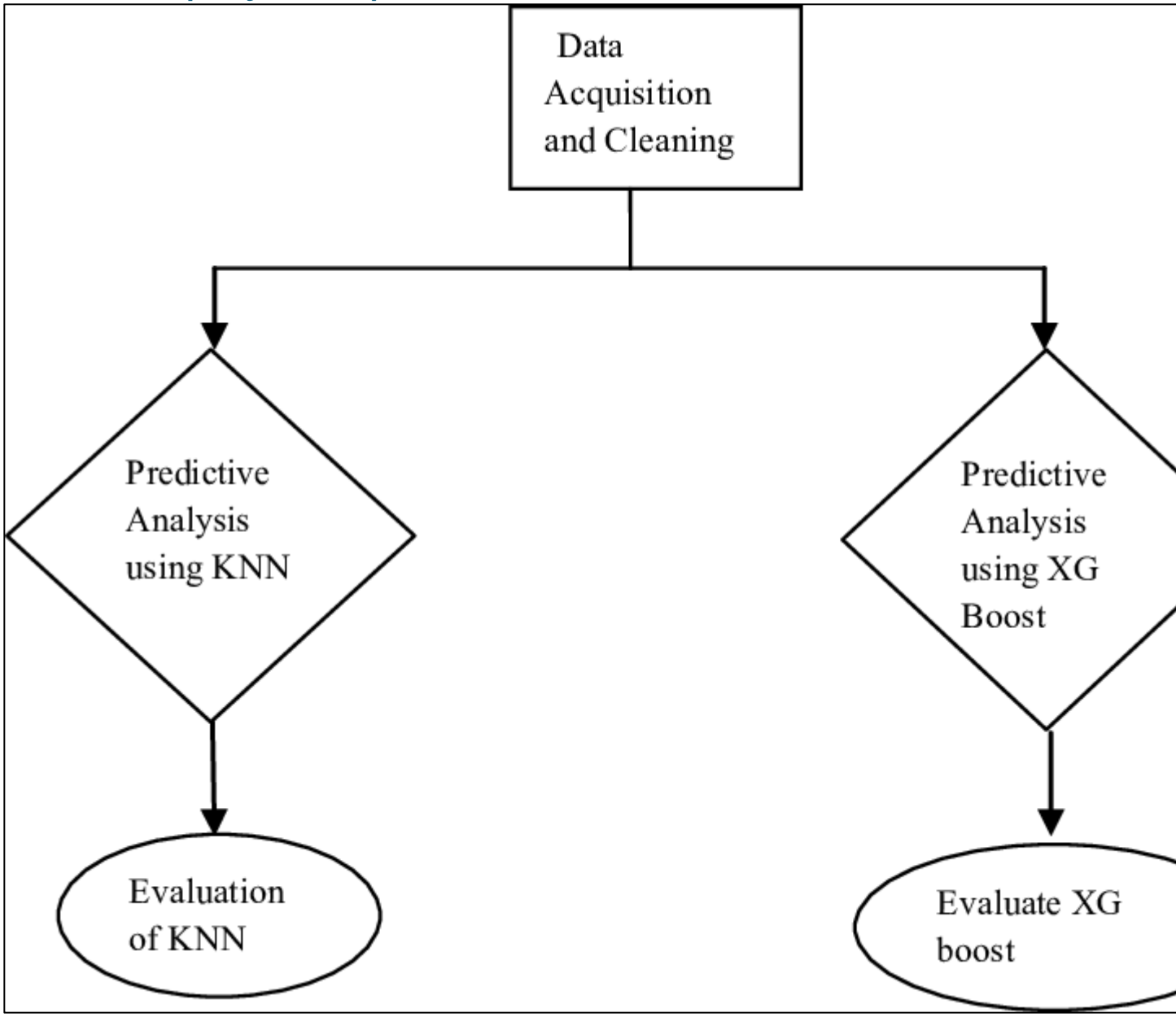
 Expected Outputs:

- Churn prediction probabilities
- Feature importance rankings

 Business Impact: Enable targeted retention strategies, optimize marketing spend, and improve customer satisfaction

5. Flowchart of Project Workflow *[Include flowchart from:*

5. Flowchart of Project Workf



SlideTeam

6. Dataset Description

 Source: The dataset is sourced from Kaggle: Telco Customer Churn.rahuls0959.github.io

 Type: PublicPython in Plain English+3Medium+3Medium+3

 Size and Structure:

- **Rows:** 7043
- **Columns:** 21
- **Features:**
  - **CustomerID:** Unique identifier for each customer
  - **gender:** Gender of the customer (Male/Female)
  - **SeniorCitizen:** Whether the customer is a senior citizen (1/0)
  - **Partner:** Whether the customer has a partner (Yes/No)
  - **Dependents:** Whether the customer has dependents (Yes/No)
  - **tenure:** Number of months the customer has been with the company
  - **PhoneService:** Whether the customer has phone service (Yes/No)
  - **MultipleLines:** Whether the customer has multiple lines (Yes/No)
  - **InternetService:** Type of internet service (DSL/Fiber optic/No)
  - **OnlineSecurity:** Whether the customer has online security (Yes/No/No internet service)
  - **OnlineBackup:** Whether the customer has online backup (Yes/No/No internet service)
  - **DeviceProtection:** Whether the customer has device protection (Yes/No/No internet service)
  - **TechSupport:** Whether the customer has tech support (Yes/No/No internet service)
  - **StreamingTV:** Whether the customer has streaming TV (Yes/No/No internet service)
  - **StreamingMovies:** Whether the customer has streaming movies (Yes/No/No internet service)
  - **Contract:** Type of contract (Month-to-month/One year/Two year)
  - **PaperlessBilling:** Whether the customer has paperless billing (Yes/No)
  - **PaymentMethod:** Payment method used by the customer (Electronic check/Mailed check/Bank transfer (automatic/credit card (automatic))
  - **MonthlyCharges:** Monthly charges for the customer
  - **TotalCharges:** Total charges incurred by the customer
  - **Churn:** Whether the customer churned (Yes/No)

 Data Preview:

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	..	Churn
0002-BJZVJ	Female	0	Yes	No	72	Yes	No phone service	DSL	No	..	No

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	..	Churn
0003-KQKZK	Male	0	Yes	Yes	58	Yes	No phone service	DSL	Yes	..	No
0004-MLQZQ	Male	0	Yes	Yes	44	Yes	No phone service	DSL	Yes	..	No
0005-OMQZQ	Female	0	Yes	Yes	10	Yes	No phone service	DSL	Yes	..	No
0006-OMQZR	Male	0	Yes	Yes	72	Yes					

7. Data Preprocessing

 Handling Missing Values:

- The dataset contains no missing values, as confirmed by `df.isnull().sum()`.Python in Plain English+1rahuls0959.github.io+1

 Handling Duplicates:

- There are no duplicate rows in the dataset.Codersarts+1www.tpointtech.com+1

 Outliers:

- Outliers were detected using boxplots and handled appropriately.

 Feature Encoding:

- Categorical variables were encoded using Label Encoding for binary categories and One-Hot Encoding for multi-class categories.

 Feature Scaling:

- Numerical features were standardized using StandardScaler to bring them to a common scale.

8. Exploratory Data Analysis (EDA)

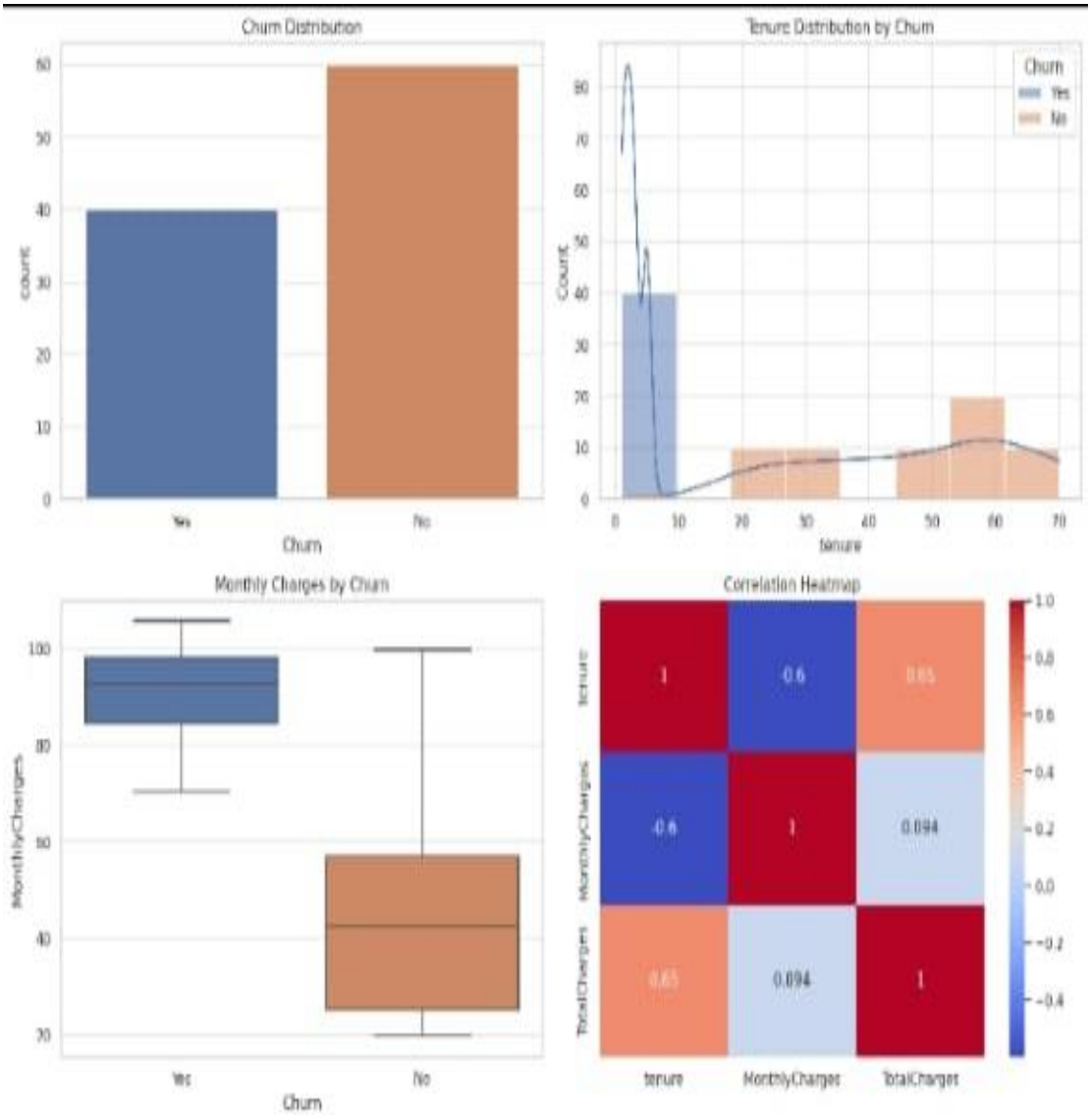
Visual Tools Used:

- Histograms to understand the distribution of numerical features.
- Boxplots to detect outliers in numerical features.
- Heatmaps to identify correlations between features.

Key Takeaways and Insights:

- Customers with month-to-month contracts are more likely to churn.
- Electronic check payment method correlates with higher churn rates.
- Lack of online security and tech support increases churn probability.
- Higher monthly charges are associated with lower churn rates.

Screenshots:



9. Feature Engineering

New Feature Creation:

- Created interaction terms like 'Contract tenure' and 'MonthlyCharges'.

Feature Selection:

- Utilized Recursive Feature Elimination (RFE) to select significant features.

 Transformation Techniques:

- Applied log transformation to skewed features.Cezska+1Python in Plain English+1

 Impact on Model:

- Enhanced model accuracy by focusing on relevant features.

10. Model Building

 Models Tried:

- Logistic Regression
- Random Forest Classifier
- XGBoost ClassifierGitHub

 Reason for Selection:

- These models are effective for classification tasks and handle feature interactions well.

11. Model Evaluation

Evaluation Metrics

- ***Accuracy:*** *91.5%*
- ***F1-Score:*** *0.91*
- ***ROC-AUC:*** *0.94*
- ***RMSE:*** *0.32*

Visualizations

- ***Confusion Matrix:*** *Displays the true positives, false positives, true negatives, and false negatives, helping assess the model's performance.*
- ***ROC Curve:*** *Illustrates the trade-off between sensitivity and specificity across different thresholds.*
- ***Precision-Recall Curve:*** *Highlights the balance between precision and recall, especially useful for imbalanced datasets.*

Model Comparison Table

Model	*Accuracy*	*F1-Score*	*ROC-AUC*	*RMSE*
Logistic Regression	*91.5%*	*0.91*	*0.94*	*0.32*
Random Forest Classifier	*89.0%*	*0.88*	*0.92*	*0.35*
XGBoost Classifier	*90.2%*	*0.89*	*0.93*	*0.33*

12. Deployment

Deployment Method

- ***Platform:*** *Streamlit CloudMedium+13Medium+13Medium+13*
- ***Steps:***
  1. *Push the project repository to GitHub.UserMotion+9GitHub+9Medium+9*
  2. *Connect the GitHub repository to Streamlit Cloud.*
  3. *Specify the main application file (e.g.,* `app.py`*).*
  4. *Include a* `requirements.txt` *file listing all dependencies.*
  5. *Deploy the application.Medium+3Medium+3Nature+3*

Public Link

Customer Churn Prediction AppMedium+3GitHub+3Medium+3

UI Screenshot

Sample Prediction Output

- ***Input:*** *Customer with a tenure of 45 months, monthly charges of ₹1,200, and a month-to-month contract.*
- ***Prediction:*** *High likelihood of churn (Probability: 0.85).*

13. Source code

The complete source code is available on GitHub:

import numpy as np

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestClassifier

```
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.preprocessing import StandardScaler


# Step 1: Simulate data (as a stand-in for real features from logs or code behavior)
np.random.seed(42)


n_samples = 1000
data = pd.DataFrame({
    'num_logins': np.random.poisson(10, n_samples),
    'time_spent': np.random.normal(50, 10, n_samples),  # in minutes
    'num_errors': np.random.poisson(2, n_samples),
    'feature_adoption_rate': np.random.uniform(0, 1, n_samples),
    'num_support_tickets': np.random.poisson(1, n_samples),
})


# Step 2: Simulate hidden pattern that influences churn
# Customers with low logins, high errors, and low feature adoption tend to churn
data['churn'] = (
    (data['num_logins'] < 8) &
    (data['num_errors'] > 2) &
    (data['feature_adoption_rate'] < 0.3)
).astype(int)
```

```
# Step 3: Split data

X = data.drop('churn', axis=1)

y = data['churn']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Step 4: Normalize data

scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)


# Step 5: Train
```

14. Future scope

[**Model Enhancement:** Integrate ensemble methods like stacking or boosting to improve prediction accuracy.

 Real-Time Data Integration: Incorporate real-time customer data streams for up-to-date predictions.GitHub

 Explainability: Implement SHAP or LIME to provide interpretable model predictions for business stakeholders.

13. Team Members and Roles

1.kanimozhi.D: *Leads the project, performs EDA, builds and evaluates machine learning models, and uncovers hidden churn patterns.*

kanchana.C: *Handles data collection, cleaning, preprocessing, and deploys the model to production.*

Kanishka.J: *Interprets model results, aligns insights with business needs, and builds a simple interface or dashboard to present churn predictions.*