Strategic Framework for Data Analytics Project: Customer Churn Prediction Using Python

# 1. Project Overview

## 1.1 Project Title:

Customer Churn Prediction and Retention Strategy Using Data Analytics

## 1.2 Objective:

To identify customers at risk of leaving a business (churning) using predictive analytics, enabling proactive retention strategies.

## 1.3 Rationale:

Retaining existing customers is significantly more cost-effective than acquiring new ones. Predicting churn enables businesses to understand why customers leave and implement measures to retain them, thus improving profitability.

## 1.4 Expected Insights:

- Key factors contributing to customer churn  
- High-risk customer segments  
- Predictive churn model with actionable retention recommendations  
- Metrics to evaluate retention campaign effectiveness

# 2. Strategic Framework

## 2.1 Problem Definition:

How can we use historical customer data to predict which customers are likely to churn and why?

## 2.2 Data Acquisition Plan:

- Source: Public datasets (e.g., Telco Customer Churn from Kaggle), Simulated company CRM/exported transactional data  
- Data Types: Customer demographics, subscription or usage history, interaction logs and support calls, feedback or complaints (if available)

## 2.3 Data Preprocessing Strategy:

- Handle missing/null values  
- Encode categorical variables (e.g., using LabelEncoder, OneHotEncoder)  
- Normalize numerical features  
- Feature selection and correlation analysis

## 2.4 Exploratory Data Analysis (EDA):

- Churn distribution (pie/bar chart)  
- Relationship between features and churn  
- Heatmap for feature correlation

## 2.5 Model Development:

Supervised Learning Approach (Classification)  
- Algorithms: Logistic Regression, Random Forest, XGBoost, Support Vector Machine  
- Model evaluation metrics: Accuracy, Precision, Recall, F1-Score, AUC-ROC Curve, Confusion Matrix

## 2.6 Interpretation of Results:

- Identify top features causing churn  
- Visualize churn probabilities  
- Generate actionable reports

## 2.7 Outcome Implementation Plan:

- Create a dashboard to monitor churn  
- Share findings with stakeholders for targeted campaigns  
- Integrate with CRM for real-time churn scoring

# 3. Timeline and Project Roadmap (Total 30–35 Hours)

|  |  |
| --- | --- |
| Task | Estimated Time |
| Project Planning & Research | 2 hours |
| Data Collection | 2 hours |
| Data Cleaning & Preprocessing | 4 hours |
| Exploratory Data Analysis (EDA) | 4 hours |
| Feature Engineering & Selection | 3 hours |
| Model Development & Training | 6 hours |
| Model Evaluation | 3 hours |
| Visualization & Reporting | 3 hours |
| Dashboard/App Development (Optional) | 4 hours |
| Final Documentation | 2 hours |
| Buffer for Testing & Adjustments | 2 hours |

# 4. Required Python Tools and Libraries

Data Handling: pandas, numpy  
Visualization: matplotlib, seaborn, plotly  
Data Preprocessing: sklearn.preprocessing  
Machine Learning Models: scikit-learn, xgboost  
Evaluation: sklearn.metrics  
Dashboard (optional): streamlit, dash

# 5. Expected Challenges & Mitigation Strategies

Challenge: Imbalanced classes (low churn ratio)  
Mitigation: Use oversampling (SMOTE) or class weighting  
  
Challenge: Noisy/incomplete data  
Mitigation: Apply preprocessing techniques and drop bad samples  
  
Challenge: Overfitting  
Mitigation: Use cross-validation and regularization techniques  
  
Challenge: Model interpretability  
Mitigation: Use SHAP values or feature importance plots