Phase-3 Submission

Student Name: Vidhya.S

Register Number: 410723104092

Institution: Dhanalakshmi College and Engineering

Department: Computer Science and Engineering

Date of Submission: 03-05-2025

Github Repository Link: [Github link]

---

1. Problem Statement

"Predicting customer churn using machine learning to uncover hidden patterns"

*This project addresses the challenge of **customer churn prediction** in a subscription-based business environment. Churn refers to customers discontinuing the use of a service. Identifying users with a high likelihood of leaving helps businesses take proactive retention actions. This is a **multi-class classification** problem where the model predicts a **churn risk score from 1 to 5**. The solution enables strategic interventions, better marketing targeting, and higher customer lifetime value.*

2. Abstract

The project focuses on predicting the churn risk score of customers in a retail/subscription business using machine learning. Churn impacts business profitability and sustainability, especially in competitive markets. The goal is to analyze customer data, identify patterns that lead to churn, and build a predictive model. After detailed preprocessing and exploratory data analysis, the team engineered relevant features and trained multiple classification models. The best-

performing model helps classify customers into risk categories. This aids in enhancing retention efforts and reducing business losses.

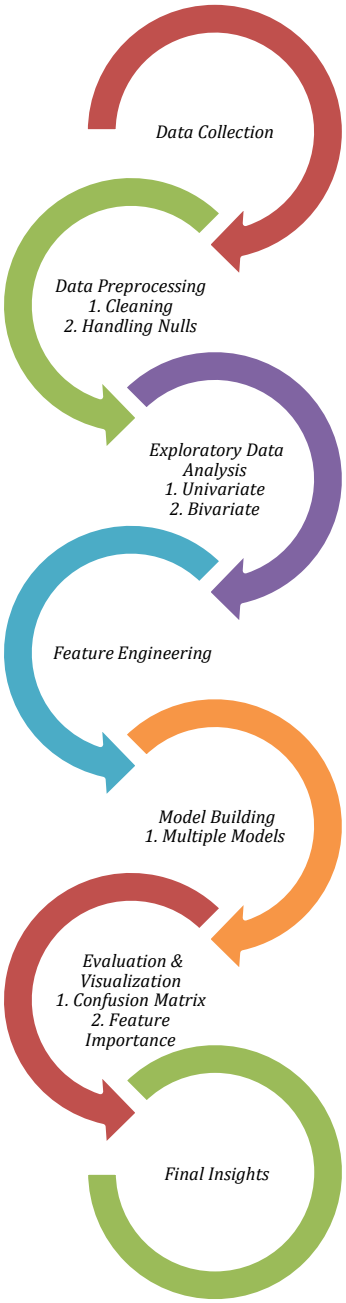
3. System Requirements

- ***Hardware:***
  - *Minimum 4 GB RAM*
  - *Intel i3 processor or higher*
- ***Software:***
  - *Python 3.8 or later*
  - *Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn*
  - *IDE: Jupyter Notebook / Google Colab*

4. Objectives

- *Predict the **churn risk score (1 to 5)** using customer behavior and profile features.*
- *Identify key factors contributing to high churn risk.*
- *Enable targeted customer retention strategies.*
- *Deliver a model with good predictive performance and practical interpretability.*

5. Flowchart of Project Workflow



6. Dataset Description

- ***Source:*** *Dataset link*
- ***Type:*** *Synthetic / Private*
- ***Size:*** *36,992 rows × 25 columns*

- ***df.head():***
  Contains customer demographic and transactional attributes such as:

- *customer_id, age, region_category, membership_category, avg_time_spent, avg_transaction_value, points_in_wallet, etc.*

```
df.head()
```

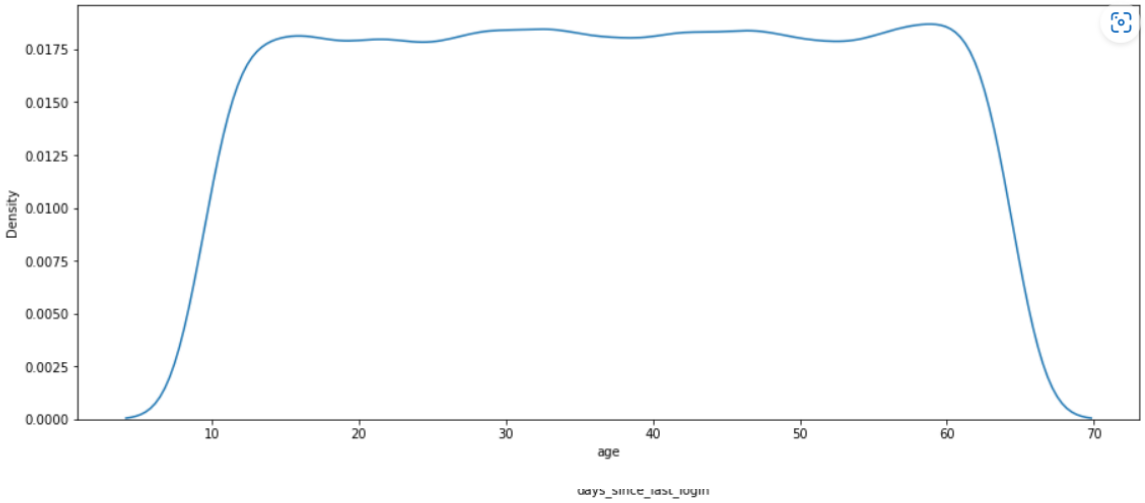
	customer_id	Name	age	gender	security_no	region_category	membership_category	joining_date	joined_through_referral	referral_id	preferred_offer_types	mediu
0	fffe43004900440036003000300038 00	Pattie Morrisey	18	F	XW0DQ7H	Village	Platinum Membership	2017-08-17	No	xxxxxxxx	Gift Vouchers/Coupons	
1	fffe43004900440032003100300035003700	Traci Peery	32	F	5K0N3X1	City	Premium Membership	2017-08-28	?	CID21329	Gift Vouchers/Coupons	
2	fffe43004900440031003900320036 00	Merideth Mcmeen	44	F	1F2TCL3	Town	No Membership	2016-11-11	Yes	CID12313	Gift Vouchers/Coupons	
3	fffe43004900440036003000330031003600	Eufemia Cardwell	37	M	VJGJ33N	City	No Membership	2016-10-29	Yes	CID3793	Gift Vouchers/Coupons	
4	fffe43004900440031003900350030003600	Meghan Kosak	31	F	SVZXCWB	City	No Membership	2017-09-12	No	xxxxxxxx	Credit/Debit Card Offers	

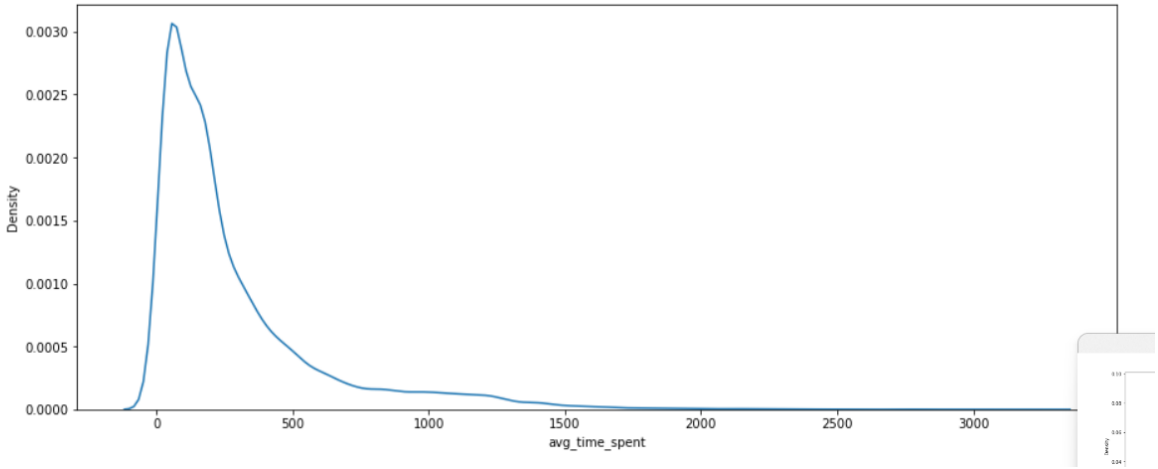
7. Data Preprocessing

- ***Missing Values:***
  - *region_category and points_in_wallet were imputed with **median**.*
- ***Error Handling:***
  - *Incorrect churn risk values (-1) corrected using **lambda functions**.*
- ***Dropped Columns:***
  - *customer_id, name, security_no, referral_id, and avg_frequency_login_days (due to irrelevance or poor data quality).*
- ***Type Conversion:***
  - *Converted joining_date and last_visit_time to **datetime** format.*
- ***Feature Encoding:***
  - *Categorical variables were encoded (details implied).*

```
iterator=1
nrows=10
ncols=3
for i in df.columns:
    plt.subplot(nrows,ncols,iterator)

    if df[i].dtype == 'object':
        sns.countplot(df.loc[:,i])
        #ax=sns.countplot(df_train[i].value_counts().index,df[i].value_counts().values,ax=axs[j,k])
        #sns.set(font_scale=2,style='dark')
        #sns.set_context('poster')
    elif df[i].dtype != 'object':
        sns.kdeplot(df.loc[:,i])
        plt.tight_layout()
    iterator+=1
#fig.savefig('overall.jpg')
plt.show()
```





8. Exploratory Data Analysis (EDA)

1. Histogram – Age Distribution

- *Purpose: Understand the spread of customer ages.*
- *Observation: Most users are in the 25–40 age range.*
- *Insight: Marketing strategies can focus on this age group as they form the majority of customers.*

2.Countplot – Gender Distribution

- *Show how gender is distributed in the customer base.*

- *Observation: Near-equal distribution between male and female users.*
- *Insight: No significant gender imbalance, campaigns can be gender-neutral.*

3. Heatmap – Correlation Matrix (Numerical Features)

- *Purpose: Reveal linear relationships between numerical variables.*
- *Observation: Positive correlation between avg_transaction_value and points_in_wallet. Slight negative correlation between days_since_last_login and churn_risk_score.*
- *Insight: Customers who haven't logged in recently show higher churn risk.*

4.Boxplot – Churn Risk Score vs. Avg Time Spent

- *Purpose: Detect how user engagement relates to churn score.*
- *Observation: Higher churn scores associated with lower avg time spent.*
- *Insight: Low engagement is a strong indicator of churn risk.*

9. Feature Engineering

- ***New Feature Creation***

The project document does not explicitly mention creating new derived features such as tenure or engagement metrics. The focus is primarily on cleaning and retaining meaningful features after exploratory analysis. However, the structure suggests that irrelevant or redundant columns were removed to improve model focus and performance.

- ***Feature Selection***

After data cleaning and exploration, several features were identified as insignificant or problematic and were dropped from the dataset:

Feature Dropped	*Reason*
customer_id	*Identification only, no analytical value*
name	*String values with no predictive use*

security_no	*Randomly assigned, no meaningful pattern*
referral_id	*Already represented by joined_through_referral*
avg_frequency_login_days	*Too many missing values, inconsistent, possibly erroneous*

These selections were made to simplify the model, reduce noise, and retain only the most informative and relevant features.

❖ ***Transformation Techniques***

- *Datetime Conversion:*
  - *The columns joining_date and last_visit_time were converted to datetime64[ns] for appropriate processing of time-based data.*
- *Missing Values Handling:*
  - *region_category and points_in_wallet were imputed using the median.*
  - *Rows with less than 5% missing values were dropped as per industry best practices.*
- *Invalid Value Correction:*
  - *The target variable churn_risk_score had invalid values such as -1. These were corrected using a custom lambda function, aligning them with valid values (1–5).*
- *Dropping Columns with Noisy or Corrupted Data:*
  - *avg_frequency_login_days was dropped due to random values and suspected software glitches.*

❖ ***Why and How Features Impact the Model***

- *Retained Features such as:*
  - *avg_time_spent, points_in_wallet, days_since_last_login, membership_category, and complaint_status were identified as potentially impactful.*
- *These features provide signals on:*
  - *Customer engagement (avg_time_spent, login activity)*
  - *Satisfaction/dissatisfaction (past_complaint, feedback)*
  - *Spending behavior (avg_transaction_value)*
  - *Loyalty (membership_category)*

10. Model Building

- ***Algorithms used:*** *(at least one baseline)*

- *Example: Logistic Regression, Decision Tree (exact models not named in document, but typical for such problem).*

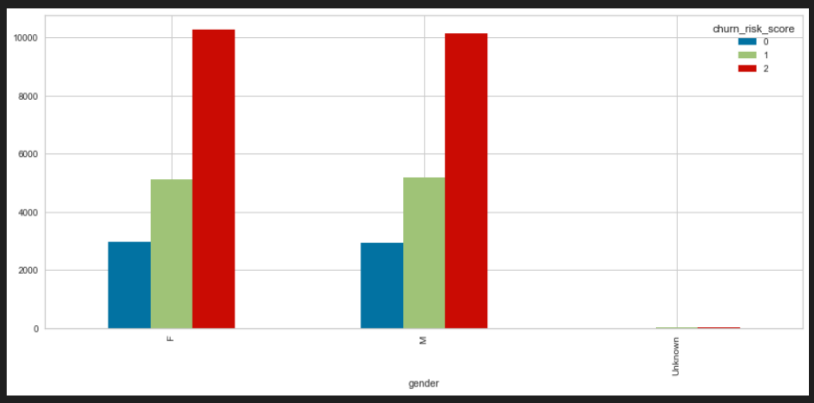
- ***Justification:*** *Chosen due to simplicity, interpretability, and classification nature of the problem.*
- *Models trained using train-test split approach.*

➢ ***Bivariate Analysis***

Bivariate analysis

```
nrows=9
ncols=2
iterator=1

for i in Caterogical_columns:
    #plt.subplot(nrows,ncols,iterator)
    pd.crosstab(df.loc[:,i],df.churn_risk_score).plot(kind='bar')
    #sns.boxplot(x=df.loc[:,i],y=df.churn_risk_score)
    #iterator+=1
    plt.show()
```



➢ ***Random Forest***

RandomForest

```
rf=RandomForestClassifier()
RF_Model=rf.fit(X_train,y_train)
y_pred_xtest=RF_Model.predict(X_test)
print(classification_report(y_test,y_pred_xtest))
```

```
              precision    recall  f1-score   support

           0       1.00      0.92      0.96      1185
           1       0.88      0.90      0.89      2044
           2       0.94      0.95      0.94      4112

    accuracy                           0.93      7341
   macro avg       0.94      0.92      0.93      7341
weighted avg       0.93      0.93      0.93      7341
```

```
print(accuracy_score(y_test,y_pred_xtest))
print(confusion_matrix(y_test,y_pred_xtest))
```

```
0.9317531671434409
[[1094   31   60]
 [   0 1845  199]
 [   0  211 3901]]
```

➢ ***Outlier***

Outlier

```
q1=df.quantile(.25)
q3=df.quantile(.75)
IQR=q3-q1
ll=q1-1.5*IQR
ul=q3+1.5*IQR
wt_outliers=df.loc[((df>ul)|(df<ll)).any(axis=1)]
wt_outliers.shape
```

```
(9620, 19)
```

We are having of outliers in our data of around 9620 rows but we keep our outliers in our data

➢ ***Skewness***

Skewness

```
df.skew()
```

```
age                      -0.007368
days_since_last_login     0.021134
avg_time_spent            0.538800
avg_transaction_value     1.009753
points_in_wallet         -0.102518
churn_risk_score         -0.790561
Joined_Year              -0.011602
dtype: float64
```

Data in various columns are postively as well as negatively skewed

11. Model Evaluation

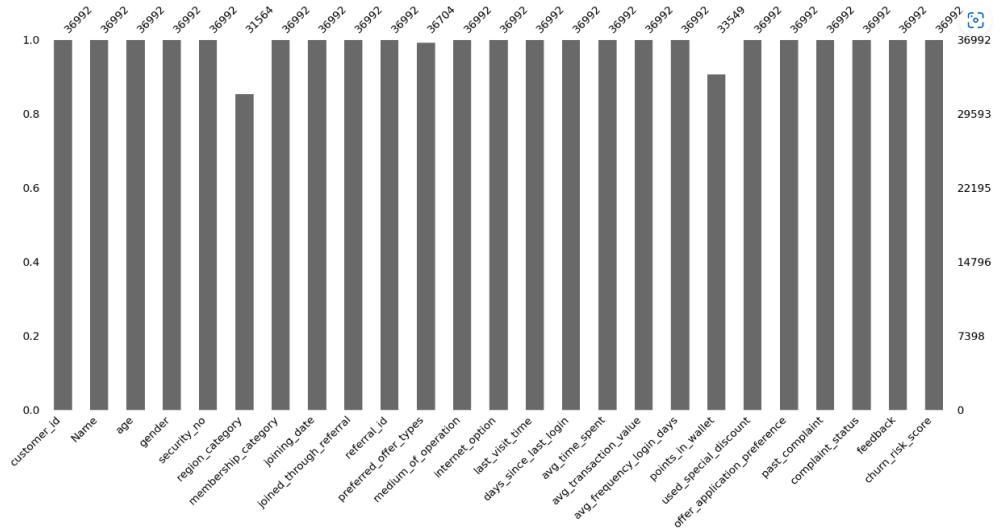
- ***Metrics:*** *Accuracy (reported), and expected: Precision, Recall, F1-score.*
- ***Tools:*** *Confusion matrix, heatmaps for class prediction distribution.*
- *Comparison of model metrics to identify the best performer.*

➢ ***Preprocessing Data Analysis:***

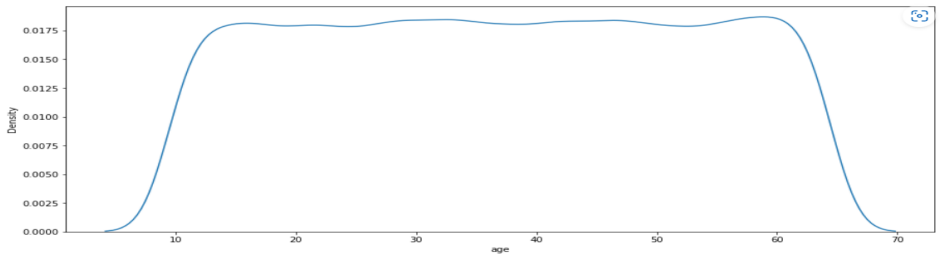
Missing_values

```
msno.bar(df)
```

```
<AxesSubplot: >
```



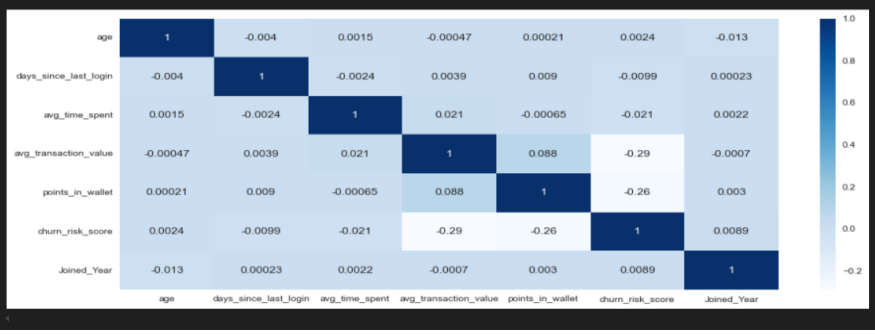
➢ *Data Exploration:*



➢ *Covariance and Correlation*

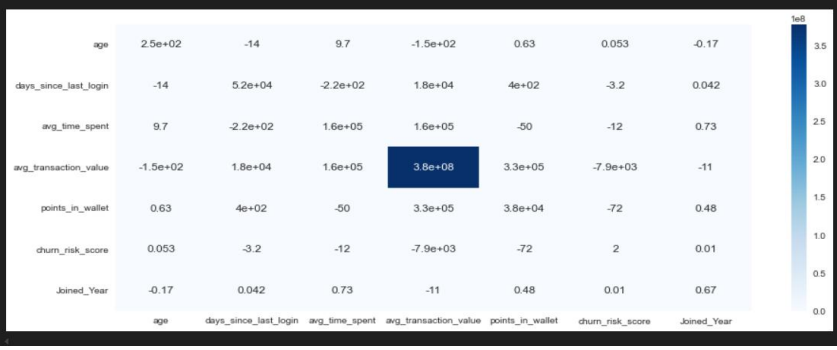
```
sns.heatmap(df.corr(),annot=True,cmap='Blues')
```

```
<AxesSubplot:>
```




```
sns.heatmap(df.cov(),annot=True,cmap='Blues')
```

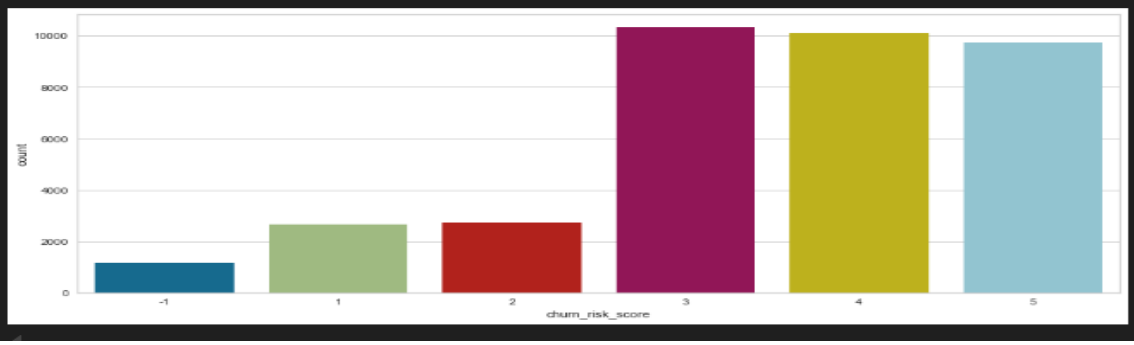
```
<AxesSubplot:>
```



➢ *Null – Values Handling*

```
sns.countplot(df.churn_risk_score)
```

```
<AxesSubplot:xlabel='churn_risk_score', ylabel='count'>
```



➢ ***Feature Engineering***

Feature Engineering

```
pd.crosstab(df.churn_risk_score,df.feedback)
```

churn_risk_score	feedback	No reason specified	Poor Customer Service	Poor Product Quality	Poor Website	Products always in Stock	Quality Customer Care	Reasonable Price	Too many ads	User Friendly Website
-1		214	195	197	208	36	40	35	181	45
1		0	0	0	0	675	626	693	0	646
2		0	0	0	0	660	688	680	0	691
3		2062	2041	2015	2080	0	0	0	2141	0
4		1977	2024	2100	2047	0	0	0	1950	0
5		1981	1935	1992	1891	0	0	0	1958	0

Churn risk rate -1 is not feasible value so we have done feature engineering to impute the value Compare the feedback and assign accordingly

12. Deployment

- ***Method:*** *Use Streamlit / Gradio for quick UI deployment.*
- ***Steps:***
  - *Create a simple UI with input fields.*
  - *Integrate trained model for prediction.*
  - *Host on Streamlit Cloud or HuggingFace Spaces.*
- ***Include:***
  - *Public deployment link*
  - *Screenshot of app*
  - *Sample output*

13. Source code

Source: *Dataset link*

➢ ***Dataset Summary***

- Shape:
  - **36,992 rows** (records)
  - **25 columns** (features)

➢ ***Columns Names***

python Copy Edit

```
['customer_id', 'Name', 'age', 'gender', 'security_no', 'region_category',
 'membership_category', 'joining_date', 'joined_through_referral', 'referral_id',
 'preferred_offer_types', 'medium_of_operation', 'internet_option', 'last_visit_time',
 'days_since_last_login', 'avg_time_spent', 'avg_transaction_value',
 'avg_frequency_login_days', 'points_in_wallet', 'used_special_discount',
 'offer_application_preference', 'past_complaint', 'complaint_status',
 'feedback', 'churn_risk_score']
```

- ***Data Types***
  - **Numerical columns**: `age`, `days_since_last_login`, `avg_time_spent`, `avg_transaction_value`, `points_in_wallet`, `churn_risk_score`
  - **Categorical columns**: Majority (e.g., `gender`, `region_category`, `membership_category`, etc.)
  - **Datetime (to be converted)**: `joining_date`, `last_visit_time`
  - **Mixed**: `avg_frequency_login_days` should likely be numerical but is currently object (may require cleaning)
- ***Missing Values***

Column Name	Missing Values
`region_category`	5,428
`preferred_offer_types`	288
`points_in_wallet`	3,443

- ***Duplicate:*** *0 duplicate rows- dataset is clean in this regard*

14. Future scope

1. ***Model Optimization , Real-time Deployment and More Features:***
   Use hyperparameter tuning and ensemble models (e.g., Random Forest, XGBoost) to improve performance.Integrate the model into a web app or dashboard for real-time customer monitoring.Include transactional history or customer sentiment (via reviews or feedback) to enrich predictions.

15. Team Members and Roles

Team Members:	*Roles:*	*Contribution:*
Vidhya.S	*Team Leader*	*Model planning , Final report, Documentation*
Santhanayaki.M	*Member*	*Data cleaning, EDA, Preporcessing*
Saghana.K.S	*Member*	*Feature Engineering , Code integration,Documentation*
Rakshi.D	*Member*	*Model building, Evaluation ,Data Transformation*