

# High Level Design (HLD)

## Credit Card Default Prediction

## Document Version Control

Date Issued	Version	Description	Author
23 July 2023	1	Initial HLD – V1.0	Nitin Udmale
			Akshay Nikam

# Contents

Document Version control.....	2
Abstract.....	4
1. Introduction.....	5
1.1 Why this High-Level Design Document ? .....	5
1.2 Scope.....	5
1.3 Definitions.....	5
2. General Descriptions.....	6
2.1 Product Perspective.....	6
2.2 Problem statement.....	6
2.3 Proposed Solution.....	6
2.4 Further Improvements.....	6
2.5 Technical Requirements.....	7
2.6 Data Requirements.....	8
2.7 Tools Used.....	9
2.7.1 Hardware Requirements.....	10
2.8 Constraints.....	10
2.9 Assumptions.....	11
3. Design Details.....	12
3.1 Process Flow.....	12
3.1.1 Model Training Evaluation.....	12
3.2 Event log.....	12
3.3 Error Handling.....	13
3.4 Performance.....	13
3.5 Reusability.....	13
3.6 Application Compatibility.....	13
3.7 Resource Compatibility.....	13
3.8 Deployment.....	13
4. KPIs (Key Performance Indicators).....	14
5. Conclusion.....	14

## Abstract

Credit card default is a significant issue, affecting consumers and financial institutions. This study focuses on developing predictive models using machine learning and data analytics to identify potential defaulters. Early intervention can help mitigate losses and maintain a healthy credit ecosystem, fostering trust between customers and financial institutions. The application of these models aids in informed decision-making and proactive measures to reduce the impact of credit card defaults. As the financial landscape evolves, the development of effective credit card default prediction models becomes crucial to ensure the stability and sustainability of the credit industry. Leveraging data-driven insights, financial institutions can take preventive actions, reducing the impact of credit card defaults on both consumers and the economy at large.

# 1. Introduction

## 1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
  - Security
  - Reliability
  - Maintainability
  - Portability
  - Reusability
  - Application compatibility
  - Resource utilization
  - Serviceability

## 1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

## 1.3 Definitions

CC	Credit Card
IDE	Integrated Development Environment

## 2. General Description

### 2.1 Product Perspective

The Credit Card Default Prediction system is an advanced machine learning-based solution that analyzes historical data to predict credit card defaults. It helps financial institutions make informed decisions, prevent losses, and maintain a stable credit ecosystem.

### 2.2 Problem statement

Financial threats are displaying a trend about the credit risk of commercial banks as the incredible improvement in the financial industry has arisen. In this way, one of the biggest threats faces by commercial banks is the risk prediction of credit clients. The goal is to predict the probability of credit default based on credit card owner's characteristics and payment history.

### 2.3 Proposed Solution

The solution is a real-time machine learning model that analyzes customer data to predict credit card defaults. Leveraging sophisticated algorithms and extensive data analytics, it empowers financial institutions to make well-informed decisions regarding credit card approvals, credit limits, and risk management strategies.

By identifying high-risk customers in real-time, the system provides timely alerts and insights, enabling financial institutions to offer targeted credit counseling and financial assistance. This proactive approach not only reduces the likelihood of credit card defaults but also fosters a more responsible and stable credit industry.

### 2.4 Further Improvements

Improvements for Credit Card Default Prediction:

1. Enhanced Features: More relevant data variables for accuracy.
2. Real-Time Data Integration: Up-to-date information.
3. Model Ensemble Techniques: Improved performance.
4. Interpretability: Transparent predictions.
5. Customer Segmentation: Tailored credit offerings.
6. Continuous Monitoring: Sustained accuracy.
7. Fraud Detection Integration: Detecting fraud.
8. Collaboration with Agencies: Comprehensive risk assessment.

## 2.5 Technical Requirements

1. High-performance Computing System: The system requires a high-performance computing infrastructure to handle the massive volumes of data and perform complex machine learning computations efficiently.
2. Secure Data Storage and Reliable Connectivity: Robust data storage solutions and reliable internet connectivity are essential to ensure the safety and accessibility of sensitive customer information.
3. Robust Server Infrastructure: A sturdy server infrastructure is necessary to host and deploy the predictive model, enabling real-time predictions and proactive risk management.
4. Data Security Measures: Stringent data security measures must be in place to safeguard sensitive customer data and comply with data protection regulations.
5. Power Supply Redundancy: To ensure uninterrupted operations, the system should have a backup power supply to mitigate the risk of downtime due to power failures.
6. Scalable Design: The system should be designed with scalability in mind, allowing it to handle increasing data volumes and accommodate future growth without compromising performance.
7. Monitoring and Maintenance Tools: Regular monitoring and maintenance tools are crucial to optimize system performance and identify any issues promptly.
8. Seamless Integration: The system must seamlessly integrate with existing credit card processing systems used by financial institutions to facilitate real-time data feeds and updates.
9. User-friendly Interface: A user-friendly interface is essential for financial institution staff to interact with the system easily and access predictive insights effortlessly.
10. Training and Support: Comprehensive training and ongoing support for staff are essential to ensure they can effectively use the system and interpret the predictions accurately.

By fulfilling these technical requirements, the Credit Card Default Prediction System can effectively predict credit card defaults, assist financial institutions in proactive risk management, and enable well-informed decision-making to prevent credit card defaults efficiently and safeguard the interests of both consumers and financial institutions.

## 2.6 Data Requirements

Data Requirements for Credit Card Default Prediction:

For the Credit Card Default Prediction project, the following data requirements must be fulfilled:

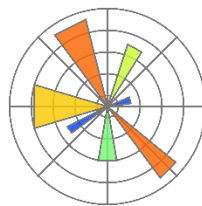
1. **Dataset Size:** The dataset should be substantial, containing a sufficient number of credit card records to ensure accurate predictions. It is recommended to have a sizable dataset with thousands of records.
2. **Balanced Classes:** The dataset should be well-balanced, meaning an equal representation of both default and non-default cases. This balance ensures the model's ability to make accurate predictions for both scenarios.
3. **Feature Columns:** The dataset should contain relevant feature columns, such as 'LIMIT\_BAL' (credit limit), 'SEX', 'EDUCATION', 'MARRIAGE', 'AGE', and payment history columns, along with the 'ID' column to uniquely identify records.
4. **Numerical Data:** The data in columns like 'LIMIT\_BAL', 'AGE', 'BILL\_AMT', 'PAY\_AMT' should be represented as numerical values.
5. **Categorical Data:** Columns like 'SEX', 'EDUCATION', 'MARRIAGE', and payment history columns may contain categorical data that needs appropriate encoding for model training.
6. **Preprocessing:** The dataset may require preprocessing steps such as handling missing values, outlier detection, and normalization/scaling to ensure the data is in a suitable format for machine learning.
7. **Data Quality:** The dataset must be clean and consistent, with minimal noise and errors, to avoid biases and inaccuracies in model predictions.

By meeting these data requirements, the Credit Card Default Prediction model can be trained effectively and yield reliable predictions, enabling financial institutions to make informed decisions to manage credit risk efficiently.



## 2.7 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn and Flask are used to build the whole model.



- VS code is used as IDE.
- For visualization matplotlib is used.
- For frontend HTML, CSS is used.
- For backend Flask is used.
- Gunicorn library is used to create WSGI HTTP server.
- GitHub is used for as version control system.
- Docker is used for easy and fast deployment and sharing of application
- Evidently AI library is used to check data drift.

## 2.7.1 Hardware Requirements

1. Computer/Server: You will need a computer or server with sufficient processing power and memory to handle the machine learning tasks. A modern multi-core processor (e.g. i7) with at least 8GB of RAM should be sufficient for small to medium-sized datasets and models. For larger datasets and more complex models, a more powerful machine with higher memory capacity would be beneficial.
2. Storage: Adequate storage space is necessary to store datasets, trained models, and other relevant files. An SSD (Solid State Drive) is recommended for faster data access and processing.
3. GPU (Graphics Processing Unit): Though not strictly necessary, having a compatible NVIDIA GPU with CUDA support can significantly speed up the training process for large machine learning models. GPUs are especially useful for deep learning models that involve neural networks.

Credit card default prediction is primarily a data science and machine learning task, and the hardware requirements will mostly revolve around the computational capabilities needed to train the models and process the data efficiently.

## 2.8 Constrains

1. User-Friendly Interface: The Credit Card Default Prediction app must have a user-friendly interface that is easy to navigate and understand. Users should be able to interact with the app without the need for specialized technical knowledge.
2. Automation: The app should be as automated as possible. Users should not be required to manually perform complex data processing or model training tasks. Instead, the app should handle data preprocessing, feature engineering, model training, and prediction processes seamlessly.
3. No Knowledge of Workings: Users should not be required to have any understanding of the underlying machine learning algorithms, data science concepts, or credit risk modeling methodologies. The app should abstract away the technical details and present the results in a straightforward manner.

4. **Reliability and Accuracy:** The Credit Card Default Prediction app should provide reliable and accurate predictions. It should be thoroughly tested and validated to ensure it produces dependable results for real-world scenarios.
5. **Real-Time or Near Real-Time Predictions:** Depending on the use case, the app may need to provide real-time or near real-time predictions for credit card transactions. The prediction process should be efficient and responsive to ensure timely decision-making.
6. **Compatibility:** The app should be compatible with different platforms and devices, such as web browsers, mobile phones, and tablets. It should accommodate various screen sizes and resolutions.
7. **Compliance:** The Credit Card Default Prediction app must comply with relevant regulations and laws related to credit card data handling and risk prediction.

## 2.9 ASSUMPTIONS

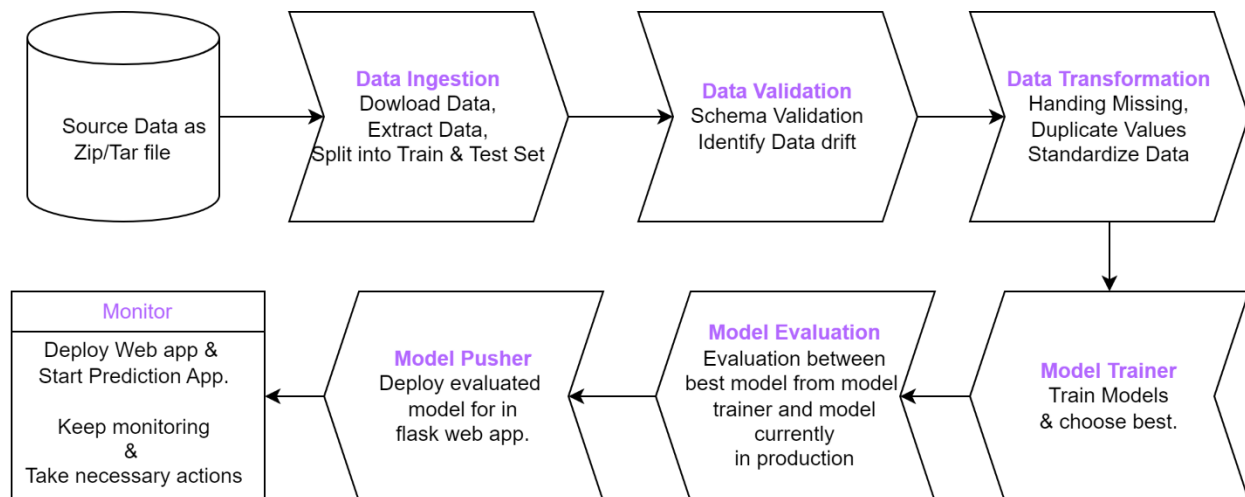
1. **Availability of Dataset:** It is assumed that a suitable and relevant dataset containing historical credit card transaction data, including information on defaults/non-defaults, is available for training and testing the Credit Card Default Prediction model.
2. **Data Quality:** The assumption is made that the dataset used for training the ML model is of high quality, with minimal missing or erroneous data. Adequate data preprocessing and cleaning will be performed to ensure the data's integrity.

## 3. Design Details

### 3.1 Process Flow

For predicting the credit card default , we will use machine learning model.  
Below is the process diagram as shown.

#### 3.1.1 Model Training And Evaluation



### 3.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. Developer can choose logging method. You can choose database logging/ File logging as well.
4. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

### 3.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage.

### 3.4 Performance

The Credit Card Default Prediction app's performance is crucial for accurate and reliable predictions. Key aspects include high prediction accuracy, precision, recall, model retraining with up-to-date data, scalability, real-time predictions, data privacy, and optional model explain ability. Monitoring and alerting ensure optimal performance

### 3.5 Reusability

The code written and the components used should have the ability to be reused with no problems.

### 3.6 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

### 3.7 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

### 3.8 Deployment



## 4. KPIs (Key Performance Indicators)

1. Default Prediction Accuracy
2. Reduction in Default Rates
3. Reduced Losses
4. Improved Portfolio Performance
5. Risk Assessment Efficiency
6. Increased Revenue
7. Customer Satisfaction
8. Operational Cost Reduction
9. Compliance and Regulatory Impact
10. Customer Acquisition
11. Optimized Credit Limits
12. Fraud Detection Improvement
13. Long-term Portfolio Health
14. Competitive Advantage
15. Timely Decision-Making

## 4. Conclusion

**The Credit Card Default Prediction app enables proactive credit risk management, reducing defaults and optimizing portfolio performance. With high accuracy and real-time predictions, it empowers stakeholders to make informed decisions, improve customer satisfaction, and ensure regulatory compliance for long-term success.**