

**REVISION NUMBER – 1.0**

**LAST DATE OF REVISION – 26-MAY-23**

**FRAUD TRANSACTION DETECTION**

**HIGH LEVEL DESIGN**

|  |  |  |
| --- | --- | --- |
| **Date issued Version** | **Description** | **Author** |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 26/05/23 | 1.0 | First Version of Complete HLD Madhvendra |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contents**

1. **Introduction**

* Brief overview of the project and its objectives
* Importance of fraud detection in financial transactions
* System Architecture

1. **High-level system architecture diagram**
2. **Description of the system components and their interactions**
3. **Data Acquisition and Preprocessing**
4. **Data sources and formats**
5. **Data cleaning and transformation techniques**
6. **Data normalization and feature scaling**
7. **Feature Extraction**
8. **Feature selection and importance analysis**

* Techniques for extracting features such as transaction amount, location, time, etc.
* Feature engineering to create new features for the model
* Model Training
* Machine learning algorithms used such as Logistic Regression, Decision Trees, Random Forest, and Neural Networks

1. **Training set and test set creation**

* Hyperparameter tuning and optimization
* Model Evaluation and Fine-tuning
* Metrics used to evaluate model performance such as accuracy, precision, recall, and F1-score
* Confusion matrix analysis to evaluate false positives and false negatives
* Techniques for fine-tuning the model to improve its performance
* Real-time Fraud Detection and Alert Generation

1. **Model deployment and integration with real-time transaction data**

* Alert generation techniques and thresholds for suspicious activity
* Escalation procedures for fraud alerts
* Documentation and Testing

1. **User manual for the system**

* Instructions for interpreting the alerts generated by the system
* System testing procedures
* Conclusion
* Summary of the key aspects of the project and its outcomes
* Potential for future enhancements and improvements
* References

**Project Title: Fraud Detection System**

Project Overview: The aim of this project is to develop a system that can detect fraudulent transactions in real-time using machine learning techniques. The system will take input data from transaction records and analyze them to identify suspicious patterns of activity. The system will be built using Python programming language and various machine learning libraries and frameworks.

**Key Features and Functionality:**

1. Data Acquisition: The system will obtain transaction data from various sources such as payment gateways, bank databases, etc.
2. Data Preprocessing: The system will clean and transform the raw transaction data to prepare it for analysis.
3. Feature Extraction: The system will extract important features such as transaction amount, location, time, etc. that are likely to be indicative of fraudulent activity.
4. Model Training: The system will use various machine learning algorithms such as Logistic Regression, Decision Trees, Random Forest, and Neural Networks to build a predictive model that can classify transactions as either legitimate or fraudulent.
5. Model Evaluation: The system will evaluate the performance of the model using various metrics such as accuracy, precision, recall, and F1-score.
6. Real-time Fraud Detection: The system will deploy the trained model to detect fraudulent transactions in real-time by comparing the transaction details with the patterns identified during the training phase.
7. Alert Generation: The system will generate alerts for suspicious transactions that are flagged as potentially fraudulent. These alerts can be sent to a designated security team or to the account holders to take necessary actions.

**Project Deliverables**:

1. A detailed report on the system architecture, design, and implementation.
2. Source code for the data preprocessing, feature extraction, and model training modules.
3. A trained model file that can be deployed for real-time fraud detection.
4. Documentation on how to use the system and interpret the alerts generated by the system.

**Project Timeline:** The project timeline will depend on the complexity of the system and the availability of resources. A tentative timeline could be as follows:

1. Data Acquisition and Preprocessing: 2 weeks
2. Feature Extraction and Model Training: 4 weeks
3. Model Evaluation and Fine-tuning: 2 weeks
4. Real-time Fraud Detection and Alert Generation: 4 weeks
5. Documentation and Testing: 2 weeks