HIGH LEVEL DESIGN (HLD)

FRAUD TRANSACTION DETECTION (FTD)

Revision Number: 2.0

Last date of revision: 24-03-2023

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| 10/01/2023 | 1 | Initial HLD – V1.0 | SK mahammadrafi |
| 07/03/2023 | 2 | Create a Model | Sk mahammadrafi |
| 20/03/2023 | 3 | Deployed In AWS | SK mahammadrafi |

**Contents**

[Document Version Control 2](#_heading=h.gjdgxs)

[Abstract 4](#_heading=h.30j0zll)

1. [Introduction 5](#_heading=h.1fob9te)
   1. [Why this High-Level Design Document? 5](#_heading=h.3znysh7)
   2. [Scope 5](#_heading=h.2et92p0)
   3. [Definitions 5](#_heading=h.tyjcwt)
2. [General Description 6](#_heading=h.3dy6vkm)
   1. [Product Perspective 6](#_heading=h.1t3h5sf)
   2. [Problem statement 6](#_heading=h.4d34og8)
   3. [PROPOSED SOLUTION 6](#_heading=h.2s8eyo1)
   4. [FURTHER IMPROVEMENTS 6](#_heading=h.17dp8vu)
   5. [Technical Requirements 6](#_heading=h.3rdcrjn)
   6. [Data Requirements 7](#_heading=h.26in1rg)
   7. [Tools used 8](#_heading=h.lnxbz9)
      1. [Hardware Requirements 8](#_heading=h.35nkun2)
      2. [ROS(Robotic Operating System) 9](#_heading=h.1ksv4uv)
   8. [Constraints 9](#_heading=h.44sinio)
   9. [Assumptions 9](#_heading=h.2jxsxqh)
3. [Design Details 10](#_heading=h.z337ya)
   1. [Process Flow 10](#_heading=h.3j2qqm3)
      1. [Model Training and Evaluation 10](#_heading=h.1y810tw)
      2. [Deployment Process 11](#_heading=h.4i7ojhp)
   2. [Event log 11](#_heading=h.2xcytpi)
   3. [Error Handling 11](#_heading=h.1ci93xb)
   4. [Performance 12](#_heading=h.3whwml4)
   5. [Reusability 12](#_heading=h.2bn6wsx)
   6. [Application Compatibility 12](#_heading=h.qsh70q)
   7. [Resource Utilization 12](#_heading=h.3as4poj)
   8. [Deployment 12](#_heading=h.1pxezwc)
4. [Deployment in AWS 14](#_heading=h.147n2zr)

# Abstract

Fraud detection is a set of activities undertaken to prevent money or property from

being obtained through false pretenses. Fraud detection is applied to many industries

such as banking or insurance. In banking, fraud may include forging checks or using

stolen credit cards. Other forms of fraud may involve exaggerating losses or causing an

accident with the sole intent for the pay-out. With an unlimited and rising number of

ways someone can commit fraud, detection can be difficult to accomplish. Fraud

detection is a critical issue for retailers determined to prevent losses and preserve

customer trust.

With the increase in digitalization, there is also increase in the fraudulent activities

happening in various domains, mainly in the retail domain. These are detrimental to the

ecosystem of online transactions. Machine learning provides an intelligent option in

dealing with this challenge.

Dataset:

# Introduction

## 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

## 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.

## Definitions

|  |  |
| --- | --- |
| *Term* | *Description* |
| *FTD* | Fraud Transaction Detection |
| *Database* | Collected data from randomly generated fraud transaction data from below link  https://fraud-detection-handbook.github.io/fraud-detection-handbook/Chapter\_3\_GettingStarted/SimulatedDataset.html |
| *IDE* | Integrated Development Environment |
| *AWS* | Amazon Web Services |

# General Description

## Product Perspective

## The Fraud Transaction Detection solution system is a supervised machine learning , classification model which will help us to detect whether the transaction is Correct or Fraud.

#### Problem statement

To create an supervised classification model using fraud transaction detection to implement the following use cases.

* + - Whether the completed transaction is fraud or not.
    - Digitalization is one of the major advancements we have in this time. The global market

is at the fingertip of each and every individual through Online purchase to detect fraud transaction.

#### PROPOSED SOLUTION

The solution proposed here is an FTD(Fraud Transaction Detection) based Surveillance (Fraud Transaction Detection) can be implemented to perform above mention use cases .

Digitalization is one of the major advancements we have in this time. The global market is at the fingertip of each and every individual through Online purchase. Both for the consumers and sellers, online market tends to give more in terms of profit as well as exposure to a larger community.

Fraud detection is applied to many industries such as banking or insurance. In banking, fraud may include forging checks or using stolen credit cards. Other forms of fraud may involve exaggerating losses or causing an accident with the sole intent for the pay-out. With an unlimited and rising number of ways someone can commit fraud, detection can be difficult to accomplish.

#### FURTHER IMPROVEMENTS

These days all works and everything is are digitalized in every domain, Digitalization is one of the major advancements we have in this time. The global market is at the fingertip of each and every individual through Online purchase. Both for the consumers and sellers, online market tends to give more in terms of profit as well as exposure to a larger community.

## Data Requirements

## In a days Every thing is digitalized , There is a lot of ways to store data in any sector , Some of them are confidential of customer transactions Based on their policy, So here Randomly generating data like transactions with time stamp , customer id , transaction amount …etc.

## In this project collected data from a website where data is generated and stored in pickle files.

## Down loaded the pickle files form the link and un pickle using python program and stores in git hub.

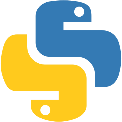
## So here we need to write a python Program to un pickle and converted them as a data frame and stored in a Git hub.

## 1.Link of Fraud Transaction Detection Dataset.

## 2.A python program for un pickle the data and convert as a data frame and upload to git.

## Tools used

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



* + - PyCharm is used as IDE.
    - For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
    - AWS is used for deployment of the model.
    - Tableau/Power BI is used for dashboard creation.
    - MySQL/MongoDB is used to retrieve, insert, delete, and update the database.
    - Front end development is done using HTML/CSS
    - Python Django is used for backend development.
    - GitHub is used as version control system.
    - Airflow and Schedulers.

## Constraints

The Fraud Transaction detection solution system must be user friendly, as automated as possible and users should not be required to know any of the workings.

## Assumptions

## Digitalization is one of the major advancements we have in this time. The global market is at the fingertip of each and every individual through Online purchase. Both for the consumers and sellers, online market tends to give more in terms of profit as well as exposure to a larger community

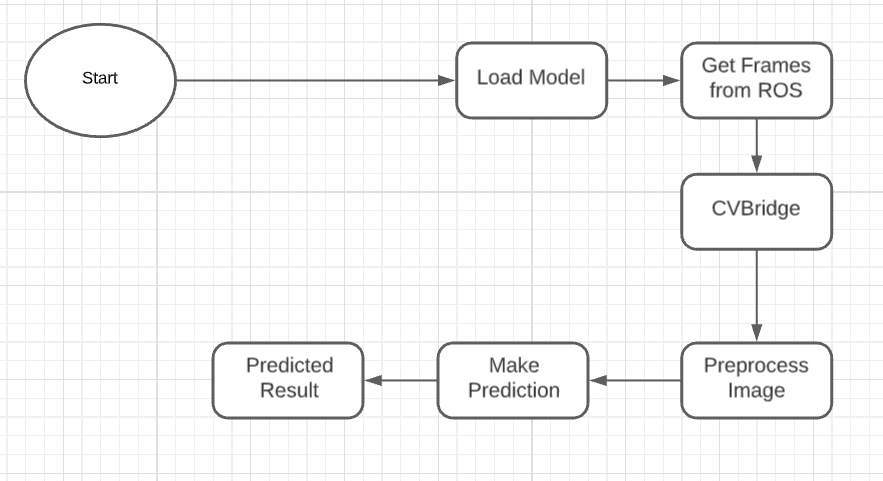
# Design Details

## Process Flow

Below is the process flow diagram is as shown below.

### Model Training and Evaluation

### Deployment Process



After testing code run successfully deployed into AWS .

## 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.

## 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.

# Performance

The FTD based surveillance solution is used for detection of fraud transactions in the society whenever FTD detects any Fraud detections it can be used in global markets , online purchase to improve transactions maintain healthy relation ships between customers and retailers. Reusability

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

## 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.

## Resource Utilization

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

* 1. **Deployment**



# Dashboards

Dashboards will be implemented to display and indicate certain KPIs and relevant indicators for the unveiled problems that if not addressed in time could cause catastrophes of unimaginable impact.



As and when, the system starts to capture the historical/periodic data for a user, the dashboards will be included to display charts over time with progress on various indicators or factors.

## DEPLOYEMENT IN AWS

1. 1.After deploying in aws we can send a file which contains information about the transactions and detecting them they are correct or fraud.
2. 2.After completing in AWS the deployment machine will give a link for accessing the model. using these link we can predict transactions. In any machine.

# Conclusion

Digitalization is one of the major advancements we have in this time. The global market

* + - is at the fingertip of each and every individual through Online purchase to detect fraud transaction. To predict whether the completed transaction is fraud or not.



