

# High Level Design (HLD) Consumer Complaint Analysis (AIOPS PROJECT)

Revision Number: 1.0

Last date of revision: 19/02/2023

Shubham Kumar Chaturvedi

## Document Version Control

Date Issued	Version	Description	Author
19th FEB 2023	1.0	First Draft	Shubham Kumar Chaturvedi

## 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 Description.....	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.....	6
2.6 Data Requirements .....	7
2.7 Tools used.....	8
2.7.1 Hardware Requirements.....	8
2.8 Constraints .....	9
2.9 Assumptions.....	9
3 Design Details .....	10
3.1 Process Flow.....	10
3.1.1 Model Training and Evaluation .....	10
3.1.2 Deployment Process .....	11
3.2 Event log .....	11
3.3 Error Handling .....	11
3.4 Performance.....	12
3.5 Reusability.....	12
3.6 Application Compatibility .....	12
3.7 Resource Utilization .....	12
3.8 Deployment.....	12
4 Dashboards.....	13
4.1 KPIs (Key Performance Indicators).....	13
5 Conclusion .....	14

## Abstract

Complaint narratives are consumers' descriptions of their experiences in their own words. Consider what conclusions may be fairly drawn from reading consumers' descriptions of their experiences. We do not adopt their views or verify that their experiences are accurate or unbiased.

Complaints can give us insights into problems people are experiencing in the marketplace and help us regulate consumer financial products and services under existing federal consumer financial laws, enforce those laws judiciously, and educate and empower consumers to make informed financial decisions.

# 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

<i>Term</i>	<i>Description</i>
<i>Database</i>	Collection of all the information monitored by this system
<i>IDE</i>	Integrated Development Environment
<i>AWS</i>	Amazon Web Services

## 2 General Description

### 2.1 Product Perspective

Design and build a scalable machine learning pipeline to predict whether a given consumer complaint will be disputed or not.

### 2.2 Problem statement

A lack of complaints or a small number of complaints against a product, issue, or firm in the database does not necessarily imply that there is little or no consumer harm. Consumers may be harmed in ways that do not cause them to file a complaint with the Bureau or to blame the product or provider for the harm they have suffered, depending on the nature of the financial product and how consumers use it.

- Consider firm size and/or market share when analyzing complaint volume about a company or product. Companies with more consumers, for example, may receive more complaints than those with fewer customers.
- Consider the population of a state or ZIP code when looking at complaints.

### 2.3 PROPOSED SOLUTION

The main objective of the project is to predict if a person gives us insights into problems. People are experiencing the marketplace and help us regulate consumer financial products and services on dates,laws and educate and empower consumers to make informed financial decisions.

### 2.4 FURTHER IMPROVEMENTS

This project shall be delivered in two phases:

Phase 1: All the functionalities with PyPi packages.

Phase2: Integration of UI to all the functionalities.

### 2.5 Technical Requirements

<b>Front End</b>	HTML/CSS/JS/React
<b>Backend</b>	Python,PySpark,PySpark ML Airflow as Scheduler.

<b>Database</b>	MongoDB
<b>Deployment</b>	AWS

## 2.6 Data Requirements

### Dataset:

Dataset is available in JSON and CSV format you can use any format based on your own interest

- Download all complaint data | CSV:  
<https://files.consumerfinance.gov/ccdb/complaints.csv.zip>
- Download all complaint data | JSON  
<https://files.consumerfinance.gov/ccdb/complaints.json.zip>



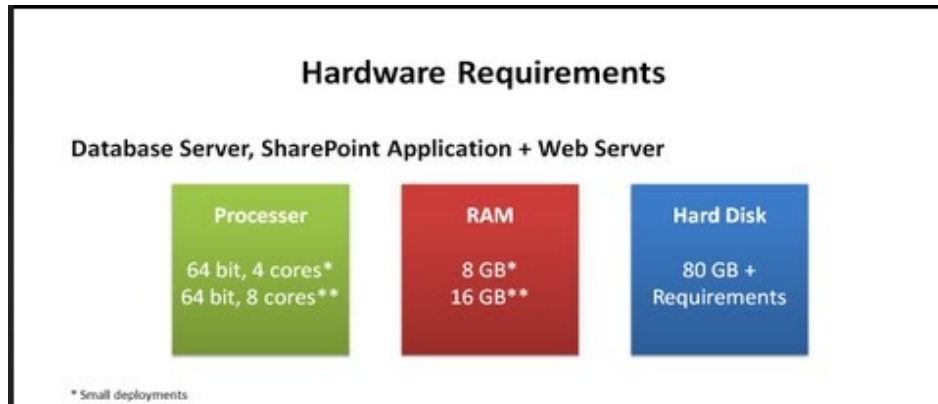
## 2.7 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, TensorFlow, Keras and Roboflow 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.

### 2.7.1 Hardware Requirements



## 2.8 Constraints

We will only be selecting a few of the issues.

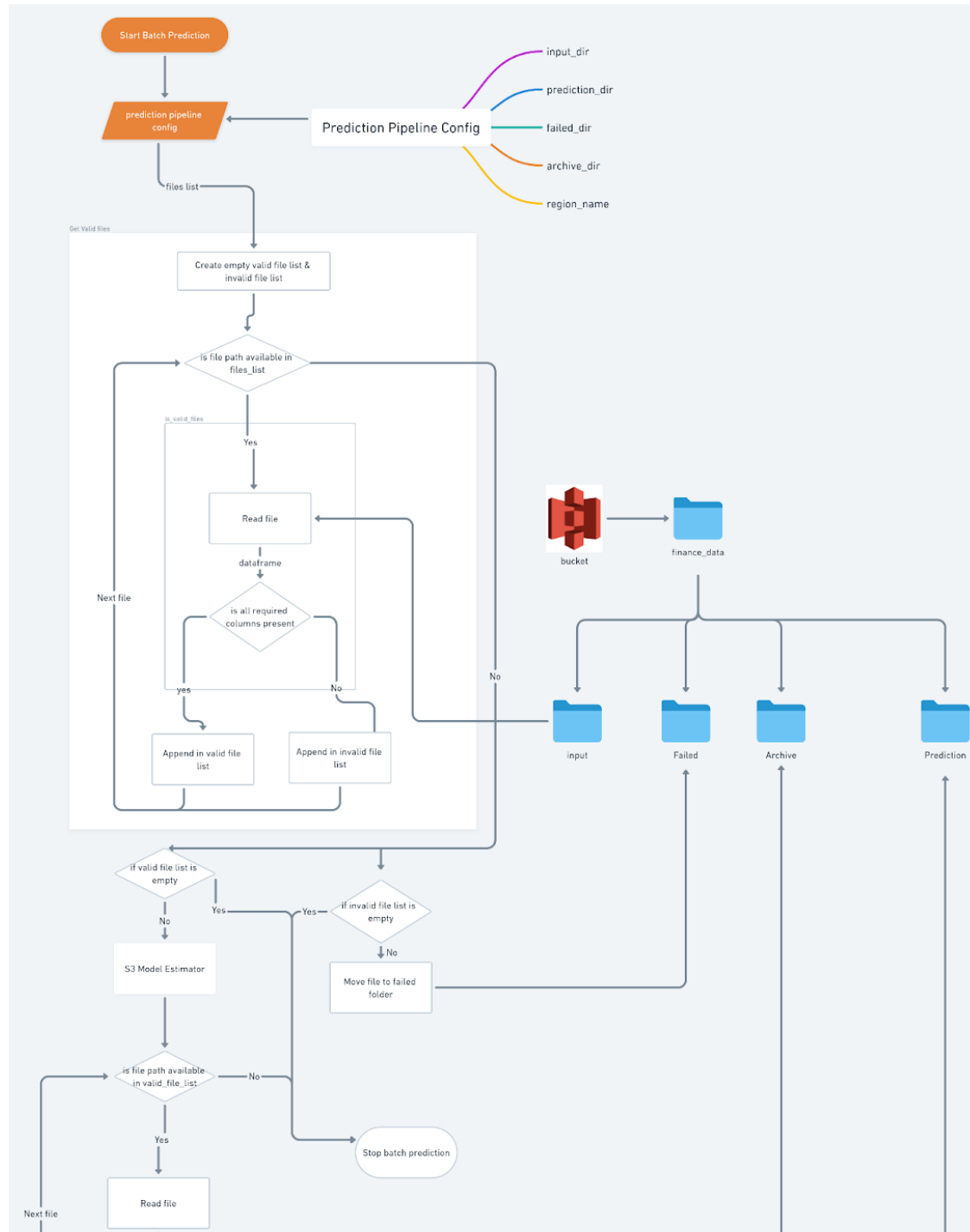
## 2.9 Assumptions

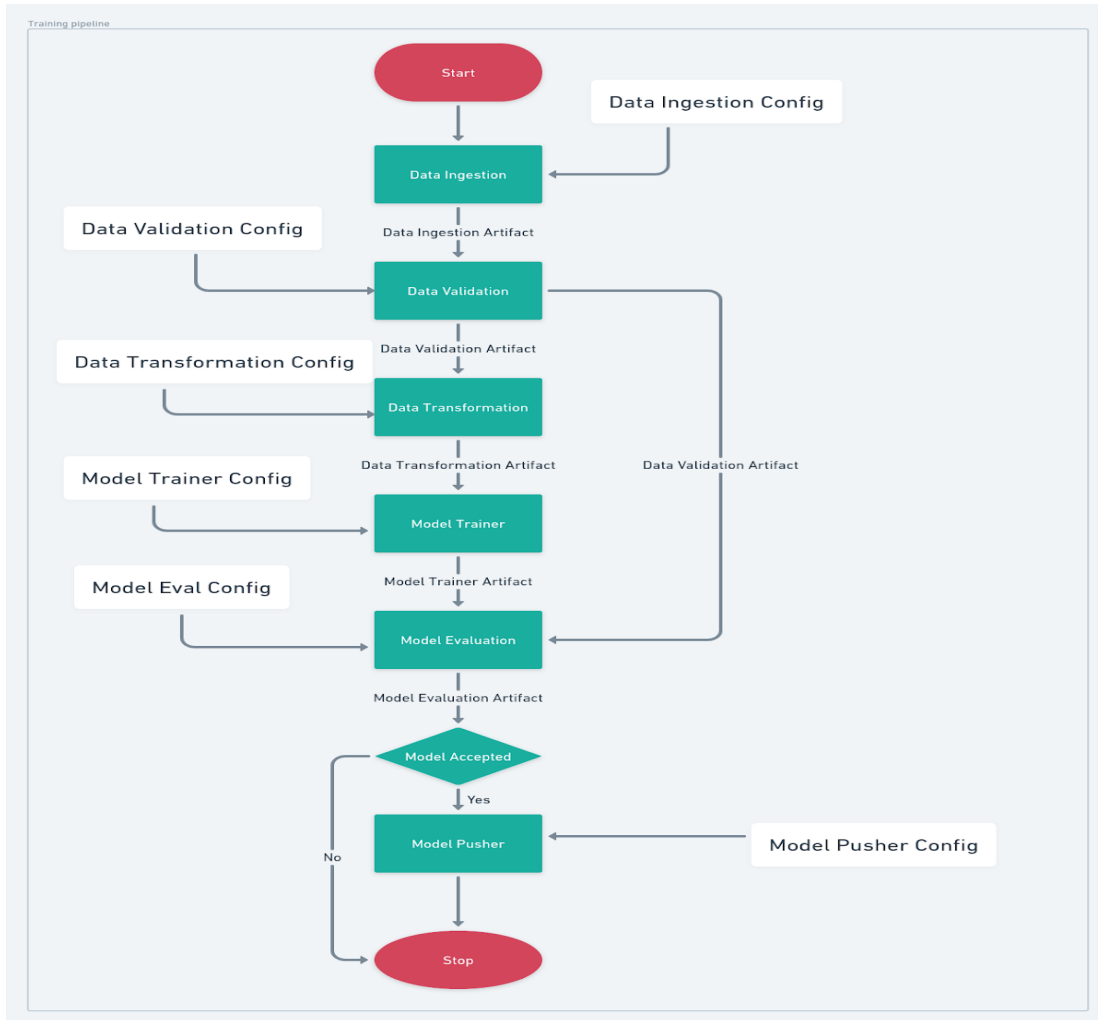
Based on the actual research paper, if we are using history of the complaints to predict the future then we might want to consider using LSTM. However, drawing a baseline in the form of some Machine Learning algorithm would be helpful. Why make a baseline model important? Well, to compare the performance of our actual model, let say LSTM in this case, is very important to ascertain that we are in the right direction as if performance of LSTM is not better than the baseline model then there is no point of using LSTM.

## 3 Design Details

### 3.1 Process Flow

For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram as shown below.





### 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.

## 4 Performance

- The system presents the set of inputs required from the user.
- The user gives required information.
- The system should be able to predict whether information/issue for the chosen dispute immediately based on the user information.

### 4.1 Reusability

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

### 4.2 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.

### 4.3 Resource Utilization

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

### 4.4 Deployment



## 5 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, dashboards will be included to display charts over time with progress on various indicators or factors.

### 5.1 KPIs (Key Performance Indicators)

An Consumer complaint analysis (CCA) contains information, such as:

- Incorrect information on your report.
- Problem with a credit reporting company's investigation into an existing problem.
- Improper use of your reportImproper use of your report.
- Company response to consumer.
- Company public response.
- The date the CFPB sent the complaint to the company.
- How did the consumer submit the complaint to the CFPB?

Only show complaints with narratives?

## 6 Conclusion

This software system will be an api solution and Web application. This system will be used to detect the useful complaints at earliest for better solution management, improved interventions, and more efficient care resource allocation using previous CCA records available. More specifically, Early detection of any preventable issue is important for better solution management. This system is designed to predict the legal and useful complaints from user complaints.



