**High level design**

**HLD**

**High Level Design (HLD)**

Finance (Adult Census Income Prediction)

**Contents :**

Document Version Control......................................................................................................02

Abstract……………………………………………………………………………………………………………………………..04

1 Introduction……………………………………………………………………………………………………………………05

1.1 Why this High-Level Design Document?.......................................................................05

1.2 Scope.............................................................................................................................05

1.3. Definitions……………………………………………………………………………………………………………….05

2 General Description............................................................................................................06

2.1 Problem Perspective………………………………………………………………………………..…………………06

2.2 Problem statement………………………………………………………………………………………………….…06

2.3 Proposed Solution……………………………………………………………………………………………………..06

2.5 Further Requirement…………………………………………………………………………………………………06

2.5 Technical Requirements…………………………………………………………………………………………….06

2.6 Data Requirements……………………………………………………………………………………………………07

2.7 Tools Used…………………………………………………………………………………………………………………08

2.7.1 Hardware Requirements…………………………………………………………………………..…………08

2.1 2 ROS (Robotic Operating System)………………………………………………………………………….09

2.8 Constraints………….……………………………………………………………………………..…………….………09

2.9 Assumptions…….………………………………………………………………………………………………...……09

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

3.3Error 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

Document Version Control :

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 22/10/2022 | 1 | Initial HLD -v1.0 | Rutuja Bankar |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Abstract :

Financing is the process of raising funds or capital for any kind of expenditure. It is the process of channelling various funds in the form of credit, loans, or invested capital to those economic entities that most need them or can put them to the most productive use.

Machine learning is a branch of artificial intelligence that uses statistical models to make predictions.

In finance, machine learning algorithms are used to detect fraud, automate trading activities, and provide financial advisory services to investors.

1. Introduction :
   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. 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. Definitions :

In finance, machine learning algorithms are used to detect fraud, automate trading activities, and provide financial advisory services to investors.

2 General Description:

2.1 Product Perspective:

Supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The type of algorithm data scientists chooses to use depends on what type of data they want to predict.

2.2 Problem Statement:

The Goal is to predict whether a person has an income of more than 50K a year or not.  
This is basically a binary classification problem where a person is classified into the  
>50K group or <=50K group.

2.3 Problem Solution :

Machine learning are many, including external (client-centric) applications such as product recommendation, customer service, and demand forecasts, and internally to help businesses improve products or speed up manual and time-consuming processes. Managing an increasing number of online customer interactions has become a pain point for most businesses. It is because they simply don’t have the customer support staff available to deal with the sheer number of inquiries, they receive daily.

* 1. FURTHER IMPROVEMENTS :
* Efficiency will skyrocket
* The employees will turn more to advisory positions
* Big firms will not be the only ones enjoying the benefits
* Online security will improve
* Chatbots will get better and more useful
* Trading and Cryptocurrency trading will improve
  1. Technical Requirements :

2.6 Data Requirements :

Machine learning algorithms consume and process large volumes of data to learn complex patterns about people, business processes, transactions, events, and so on. This intelligence is then incorporated into a predictive model. Comparisons to the model can reveal whether an entity is operating within acceptable parameters or is exhibiting an anomaly. Today, machine learning is used to solve well-bounded tasks such as classification and clustering

Machine learning has a voracious appetite for data during both development and production, making unique demands of an organization's infrastructure for data management.

**1.Large, diverse data sets :**

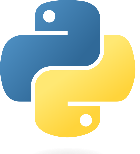
The development of a machine learning algorithm depends on large volumes of data, from which the learning process draws many entities, relationships, and clusters. To broaden and enrich the correlations made by the algorithm, machine learning needs data from diverse sources, in diverse formats, about diverse business processes. Savvy organizations are deploying tools for multiple types of analytics (not just machine learning), because each type tells them something unique and valuable. The data environment must provision large quantities of raw data for discovery-oriented analytics practices such as data exploration, data mining, statistics, and machine learning.

**2.Large, diverse infrastructure for data management:**

Infrastructure for training data for machine learning typically involves multiple data platforms, tools, and processing engines, ranging from traditional (relational and columnar databases) to modern (Hadoop, Spark, and cloud storage). Multiple technologies are required to cope with training data's extreme size, multiple data structures, and (in some cases) multiple latencies. Tools for machine learning are obviously important, but data management infrastructure is just as important.

* 1. Tools used :

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

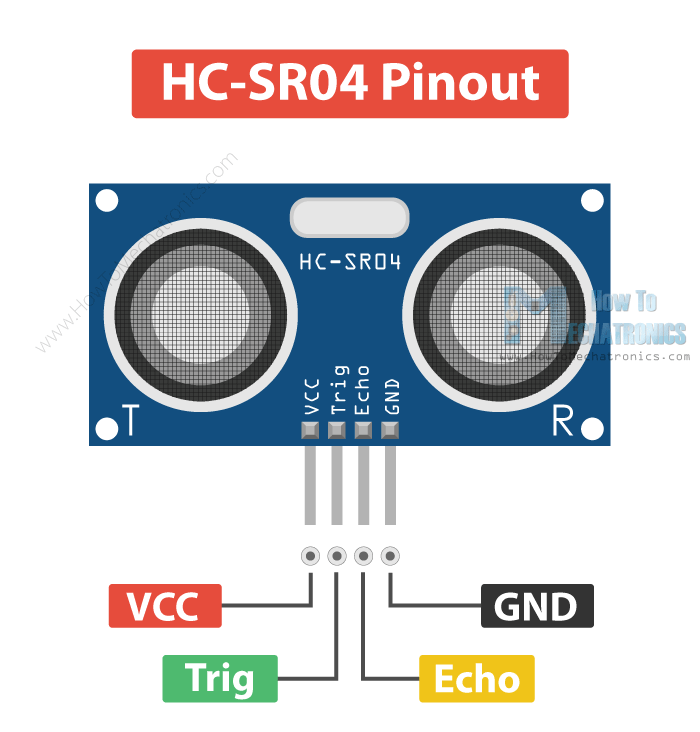




* PyCharm is used as IDE.
* For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
* AWS is used for deployment of the model.
* .Tableau/Power Bl is used for dashboard creation.
* MySQL/MongoDB is used to retrieve, insert, delete, and update the database.
* Front end development is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

2.7.1 Hardware Requirements

* USB Camera for object Detection
* LM35 temperature sensor
* MO Smoke Detector Sensor
* PC (check you are system supports: <https://7dfps.com/ros-system> requirement/)
* HC-SR04 Ultrasonic sensor
* Raspberry Pi

2.7.2 ROS (Robotic Operating System):

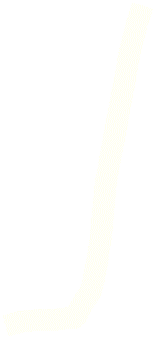
Robot Operating System is an open-source robotics middleware suite. Although ROS is not an operating system but a collection of software frameworks for robot software development, it provides services designed for a heterogeneous computer cluster such as hardware abstraction, low-level device control, implementation of commonly used functionality, message-passing between processes, and package management

* 1. Constraints:

There are two ways to apply physical constraints in ML: soft constraints, which are enforced by adding extra penalties to the loss function; and hard constraints, which refer to conditions that must be satisfied when generating the model.

* 1. Assumptions :

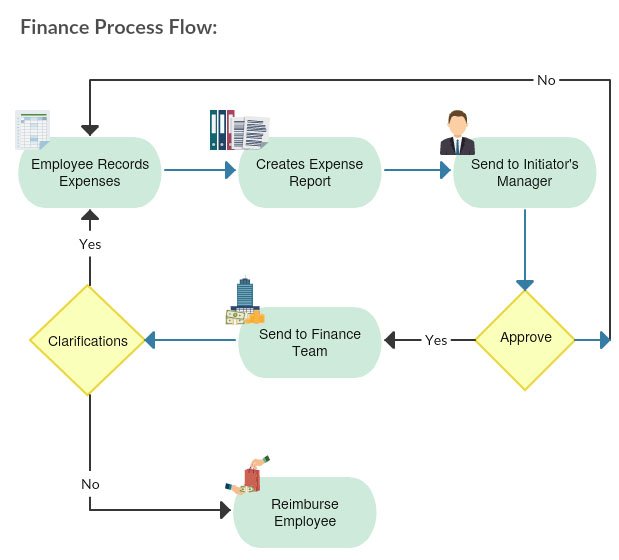
It assumes that there is minimal or no multicollinearity among the independent variables. It usually requires a large sample size to predict properly. It assumes the observations to be independent of each other

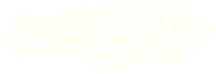


1. Design Details :

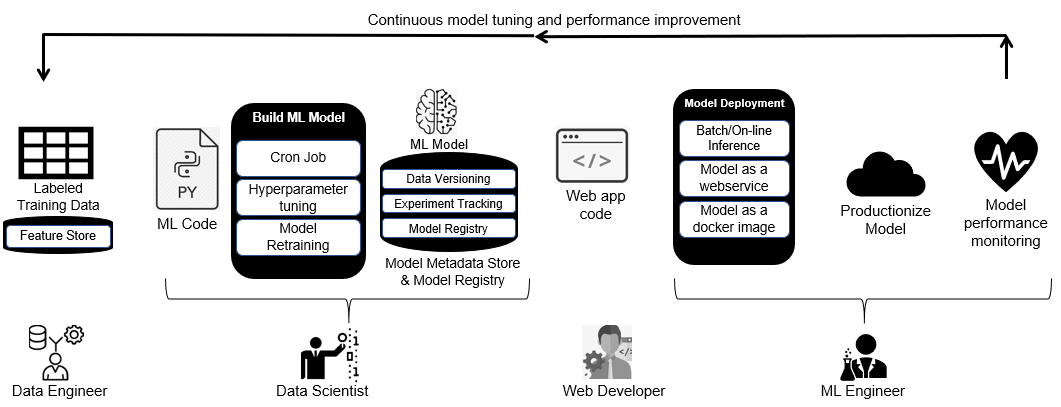
3.1 Process Flow







3.1.1 Model Training and Evaluation



3.1.2 Deployment Process



3.2 Event log

The main source of data for the discovery of the business processes that make up a company. 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 :

Financial performance is a subjective measure of how well a firm can use assets from its primary mode of business and generate [revenues](https://www.investopedia.com/ask/answers/111314/whats-difference-between-retained-earnings-and-revenue.asp). The term is also used as a general measure of a firm's overall financial health over a given period.

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 KPls 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 overtime with progress on various indicators or factors.

* 1. KPls (Key Performance Indicators)

### Financial Metrics

### **Liquidity Ratios**

### **Profitability Ratios**

### **Solvency Ratios**

### **Turnover Ratios**

### Customer Metrics

### **Number of New Ticket Requests**

### **Number of Resolved Tickets**

### **Average Resolution Time**

### **Average Response Time**

### **Top Customer Service Agent**

### **Type of Request**

### **Customer Satisfaction Rating**

### Process Performance Metrics

### **Production Efficiency**

### **Total Cycle Time**

### [Throughput](https://www.investopedia.com/terms/t/throughput.asp)

### **Error Rate**

### **Quality Rate**

### Marketing

### **Website Traffic**

### [Social Media](https://www.investopedia.com/terms/s/social-media.asp)**Traffic**

### **Conversation Rate on Call-To-Action Content**

### **Blog Articles Published Per Month**

### **Clickthrough Rates**

### IT

### **Total System Downtime**

### **Number of Tickets/Resolutions**

### **Number of Developed Features**

### **Count of Critical Bugs**

### **Back-up Frequency**

### Sales

### **Customer Lifetime Value (CLV)**

### [Customer Acquisition Cost](https://www.investopedia.com/terms/a/acquisition-cost.asp)**(CAC)**

### **Average Dollar Value for New Contracts**

### **Average Conversion Time**

### **Number of Engaged Leads**

### Staffing

### [**Absenteeism**](https://www.investopedia.com/terms/a/absenteeism.asp) **Rate**

### **Number of Overtime Hours Worked**

### **Employee Satisfaction**

### **Employee Turnover Rate**

### **Number of Applicants**

### 6.Conclusion:

In this paper, we studied the question of how researchers can leverage ML technology in finance. First, we established that different types of ML solve different problems than traditional linear regression with OLS. While the properties of OLS are beneficial for explanation problems, supervised ML is the superior method for prediction problems. As we illustrated with a real estate  
asset pricing prediction problem, ML-based price predictions can achieve dramatically lower pricing errors than OLS.