

# High Level Design (HLD)

## Store Sales Prediction

Revision Number: 2.0

Last date of revision: 30-07-2022

Revision Number 1.0

Last Date of revision : 30-07-2022

# Document Version Control

Date Issued	Version	Description	Author
25-05-2022	1	Initial HLD -V1.0	Siddhartha Borgohain
25-07-2022	2	Update the data	Siddhartha Borgohain

# Content

Document Version Control

Abstract

1	Introduction.....	5
1.1	Why this High-Level Design Document?.....	5
1.2	Scope.....	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.....	7
2.8	Constraints.....	8
3	Design Details.....	9
3.1	Process Flow.....	9
3.1.1	Model Training and Evaluation.....	10
3.1.2	Deployment Process.....	11
4	Performance.....	12
4.1	Reusability.....	12
4.2	Application Compatibility.....	12
4.3	Resource Utilization.....	12
4.4	Deployment.....	12
5	Conclusion.....	13

# ABSTRACT

This is the age of the internet where the amount of data being generated is so huge that man alone is not able to process the data. Many machine learning techniques hence have been discovered for this purpose. We are trying to predict the sales of a store using different machine learning techniques and trying to determine the best algorithm suited to our particular problem statement. We have implemented normal regression techniques as well as boosting techniques in our approach and have found that the boosting algorithms have better results than the regular regression algorithms.

# 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 performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like :
  - Security
  - Relatability
  - 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.

## 2 General Description

### 2.1 Product Perspective

Store Sales Prediction is a solution which is able to predict the sales of the different stores of Big Mart according to the provided dataset.

### 2.2 Problem Statement

Nowadays, Shopping malls and Big Marts keep track of individual item sales data in order to forecast future client demand and adjust inventory management. In a data warehouse, these data stores hold a significant amount of consumer information and particular item details. By mining the data store from the data warehouse, more anomalies and common patterns can be discovered.

### 2.3 Proposed Solution

We will perform EDA to find the important relation between different attributes and will use a machine-learning algorithm to predict the future sales demand. The client will be filled the required feature as input and will get results through the web application. The system will get features and it will be passed into the backend where the features will be validated and preprocessed and then it will be passed to a hyperparameter tuned machine learning model to predict the final outcome.

### 2.4 Data Requirements

The data required for the building of the project is already available on the dashboard. The Store Sales Prediction data recorded many product descriptions along with past sales quantity. For building the ml model we will use the dataset that is given. The data is consist of 8523 rows and various information about products like product id, product category, store id, store location, e.t.c

## 2.4 Further Improvements

We can save the User's History of Prediction and identify the item's future demand. It will automatically be assigned for production.

## 2.5 Technical Requirements

Any Device with Internet Access and runs in any browser.

## 2.6 Data requirements

Data requirements completely depend on our problem statement.

We have train (8523) and test (5681) data set, train data set has both input and output variable(s). We need to predict the sales for the test data set.

- Item\_Identifier: Unique product ID
- Item\_Weight: Weight of product
- Item\_Fat\_Content: Whether the product is low fat or not
- Item\_Visibility: The % of total display area of all products in a store allocated to the particular product
- Item\_Type: The category to which the product belongs
- Item\_MRP: Maximum Retail Price (list price) of the product
- Outlet\_Identifier: Unique store ID
- Outlet\_Establishment\_Year: The year in which store was established
- Outlet\_Size: The size of the store in terms of ground area covered
- Outlet\_Location\_Type: The type of city in which the store is located
- Outlet\_Type: Whether the outlet is just a grocery store or some sort of supermarket
- Item\_Outlet\_Sales: Sales of the product in the particular store. This is the outcome variable to be predicted.

## 2.7 Tools Used

Python Programming language and frameworks such as Numpy, Pandas, Scikit-learn, Google cloud are used to build the whole model.





- VS code and Google colab is used as IDE.
- For visualization of the plots Autoviz, Matplotlib, Seaborn are used.
- Flask is used for deployment of the model.
- Front end Development is done using HTML/CSS, Bootstrap 4.
- Python is used for backend development.
- Github is used as a version control system.

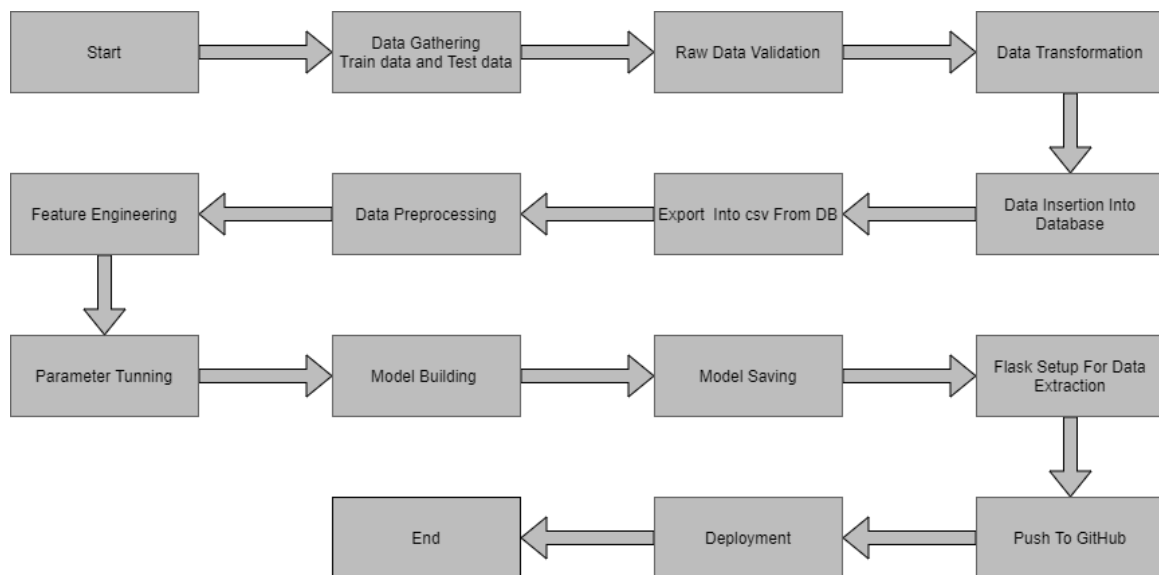
## 2.8 Constraints

The Store Sales Prediction system must be user friendly, as automated as possible and users should not be required to know any of the workings.

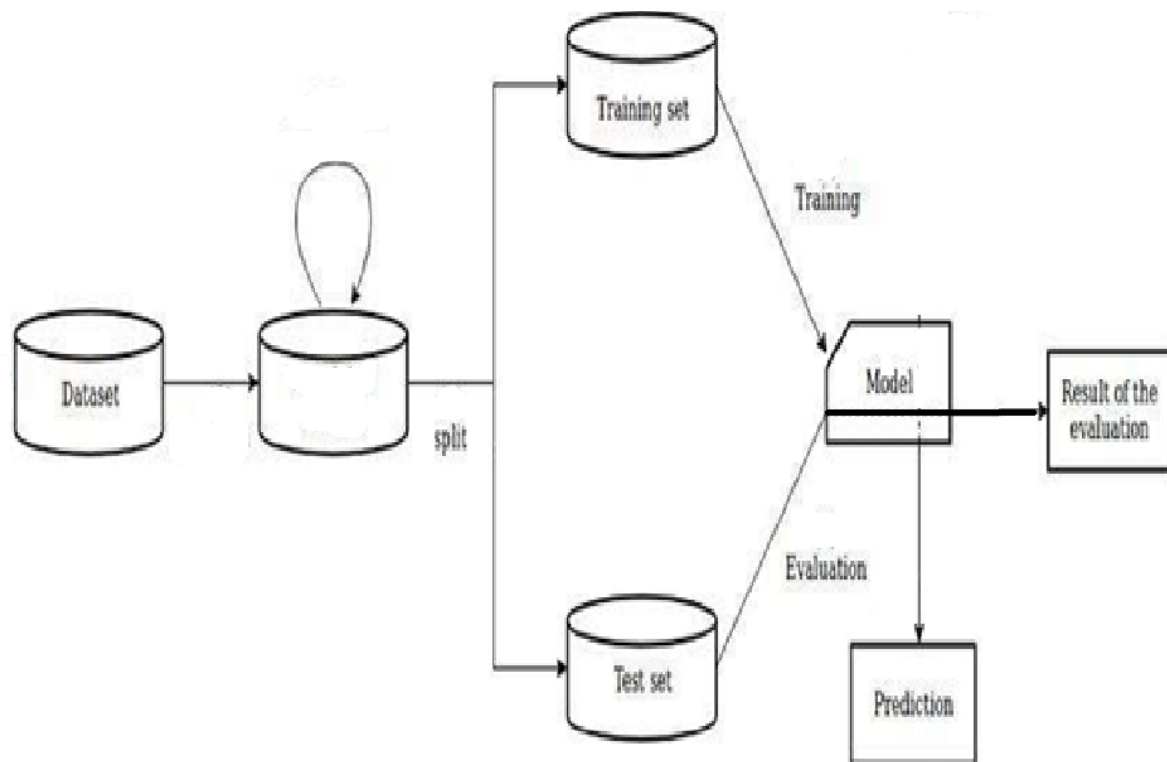
## 3 Design Details

### 3.1 Process Flow

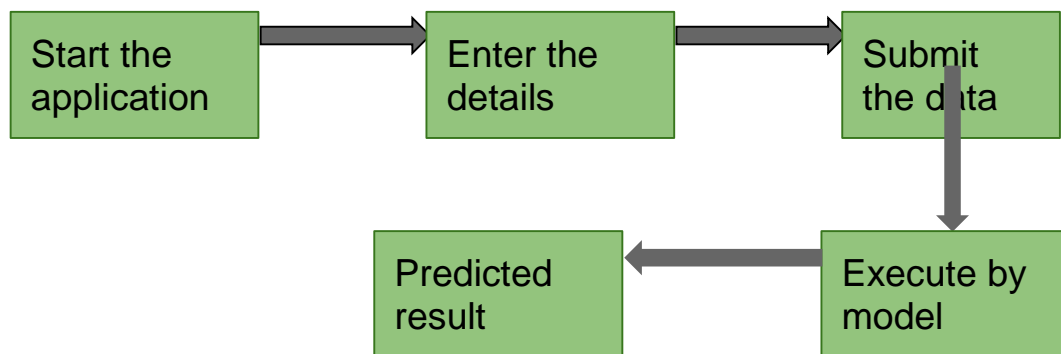
For Identifying the different types of anomalies, we will use a machine learning model. Below is the process flow diagram as shown below.  
Proposed Methodology



### 3.1.1 Model Training and Evaluation



### 3.1.2 Deployment Process



## 4 Performance

The Performance of the model depends on the dataset. We have done in depth preprocessing of the dataset for greater accuracy and much closer prediction with less error.

### 4.1 Reusability

The code and the module are created during the time of building the project should maintain all coding guidelines and full project code is written in a Modular fashion. Our system should have the flexibility to work properly from any location. And it should handle any improper input value from the user and should give a meaningful error message so the user can correct his/her mistake and enter valid input to get the result. And the system should be reusable in every manner with different types of inputs values that are all are it has been trained.

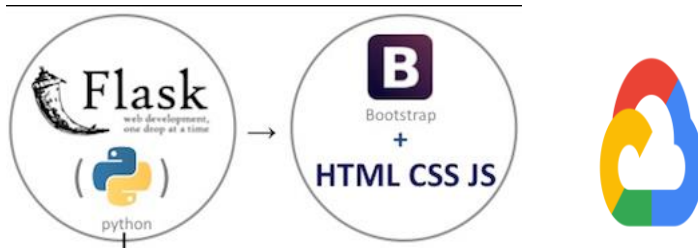
### 4.2 Application Compatibility

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

### 4.4Deployment



## 5 Conclusion

Model is able to predict the sales of the different stores of Big Mart according to the provided dataset.