Low Level Design (LLD)

BIG MART SALE PREDICTION APP

Revision Number: 1

Last date of revision: 08/02/2022

Afrin R Peshimam

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 8th Feb 2022 | 1.1 | Creation of document | Afrin R Peshimam |

**Contents**

[Document Version Control 2](#_Toc95207677)

[1 Introduction 4](#_Toc95207678)

[1.1 Why this Low-Level Design Document? 4](#_Toc95207679)

[1.2 Scope 4](#_Toc95207680)

[1.3 Constraints 4](#_Toc95207681)

[1.4 Risks 4](#_Toc95207682)

[1.5 Out of Scope 4](#_Toc95207683)

[2 Architecture Description 5](#_Toc95207684)

[2.1 Dataset 5](#_Toc95207685)

[2.1.1 Sales dataset overview 6](#_Toc95207686)

[2.2 CSV Training 6](#_Toc95207687)

[2.3 Data Pre processing 6](#_Toc95207688)

[2.4 Best Model 6](#_Toc95207689)

[2.5 Model Saving 6](#_Toc95207690)

[2.6 Cloud Setup 6](#_Toc95207691)

[2.7 Push app to cloud 6](#_Toc95207692)

[2.8 Data from client side for prediction purpose 7](#_Toc95207693)

[2.9 Predicting Sales 7](#_Toc95207694)

[3 Technology stack 7](#_Toc95207695)

**Abstract**

Build a predictive model and find out the sales of each product at a particular store

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Deep EHR System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

The main objective of the project is to predict the sales of each product in different Stores.

## Scope

This software system will be a Web application This system will be designed to detect sale of each product. This system will help in better store inventory management. It will lead to reduction in cost and increase in profit.

## Constraints

We will only be selecting a few stores.

## Risks

Document specific risks that have been identified or that should be considered.

## Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

# Technical specifications

## 2.1 Dataset

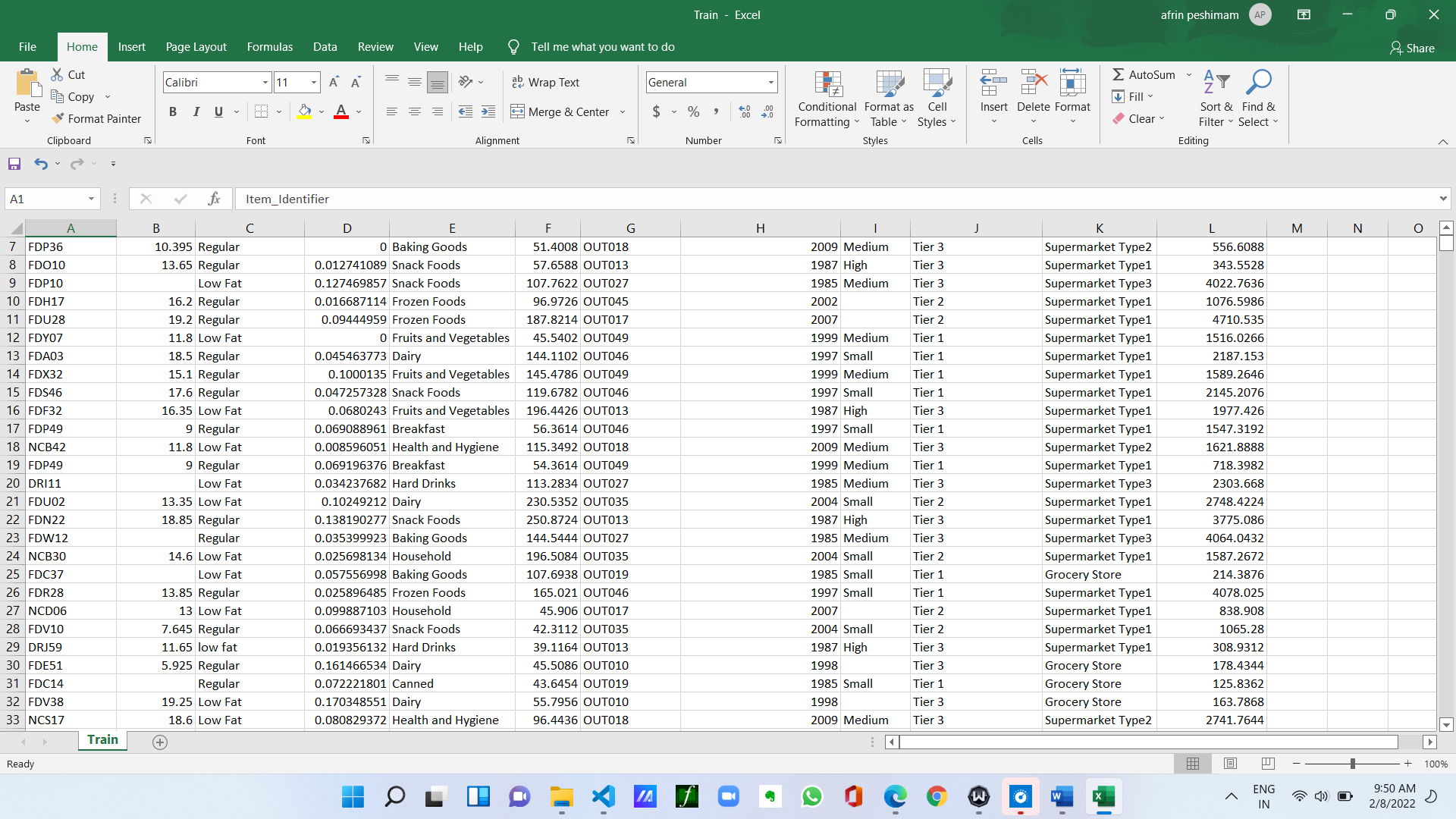
|  |  |
| --- | --- |
| **Data** | **Description** |
| Item Identifier | Unique product ID |
| Item Weight | Weight of product |
| Item Fat | Low Fat or Regular |
| Item Visibility | The % of total display area of all products |
| Item Type | The category to which the product belongs |
| Item MRP | Maximum Retail Price |
| Outlet Identifier | Unique Store ID |
| Outlet Establishment Year | Year store was created |
| Outlet Size | The size of store in term of ground area. |
| Outlet Location | Type of City store is located |
| Outlet Type | Whether outlet is grocery or supermarket |
| Item Outlet Sales | Sales of Product |

## 2.1.1 Sales dataset overview

Consists of 1 table.

There are a total of 8,523 records in the training set and 5,681 records in the test set.

* Sales



## 2.2 CSV Training

Here we will take all the data we want to train

## 2.3 Data Pre processing

We will be exploring our data set here and do EDA if required and perform data pre processing depending on the data set. We first explore our data set in Jupyter Notebook and decide what pre-processing and Validation we have to do such as imputation of null values, dropping some column, etc and then we have to write separate modules according to our analysis, so that we can implement that for training as well as prediction data

## 2.4 Best Model

Here we will train our model on the pre-processed data and select the best model.

## 2.5 Model Saving

Saved the best model

## 2.6 Cloud Setup

Here We will do cloud setup for model deployment. Here we also create our flask app and user interface and integrate our model with flask app and UI.

## 2.7 Push app to cloud

After doing cloud setup and checking app locally, we will push our app to cloud to start the application.

## 2.8 Data from client side for prediction purpose

Now our application on cloud is ready for doing prediction. The prediction data which we receive from client side and further will do same data cleansing process as we have done for training data and we will use our saved model for prediction on that data.

## 2.9 Predicting Sales

* The system displays the input fields for data.
* The User puts the values and click predict
* The system should be able to predict sales based on information provided.

# Technology stack

|  |  |
| --- | --- |
| **Front End** | HTML/CSS/BOOTSTRAP |
| **Backend** | Python Flask |
| **Deployment** | Heroku |

# PERFORMANCE

The XGBoost model gets R2 of 0.90 and rmse of 600