Low Level Design (LLD) Stores Sales Prediction

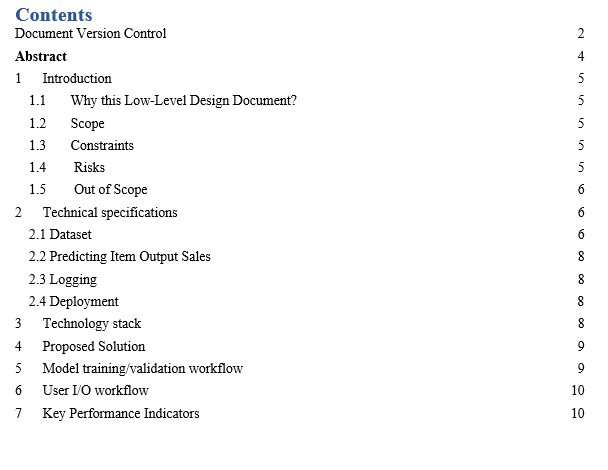
Revision Number: 1.0

Last date of revision: 03/09/2021

Ashok Kumar S M C Shravan

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| **3rd September2021** | 1.0 | Complete Architecture | M C Shravan, Ashok Kumar S |
|  |  |  |  |



# Abstract

The value for a particular product or item keeps on changing from time to time. Any business cannot improve its financial performance without estimating the demand of the customer and future sales of products or items accurately. In this project, we are trying to predict the sales of a stores and Big Marts using different machine learning techniques and trying to determine the best algorithm suited to our particular problem statement.

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Stores Sales Prediction. 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 the given product in stores or Big Marts. Our datasets consists of information, such as:

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.

## Scope

This software system will be a Web application. This system will be designed to predict the sales outlets for different stores and Big Marts using Maachine Learning Techniques

## Constraints

Variation in accuracy and limited to the given dataset.

## 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.

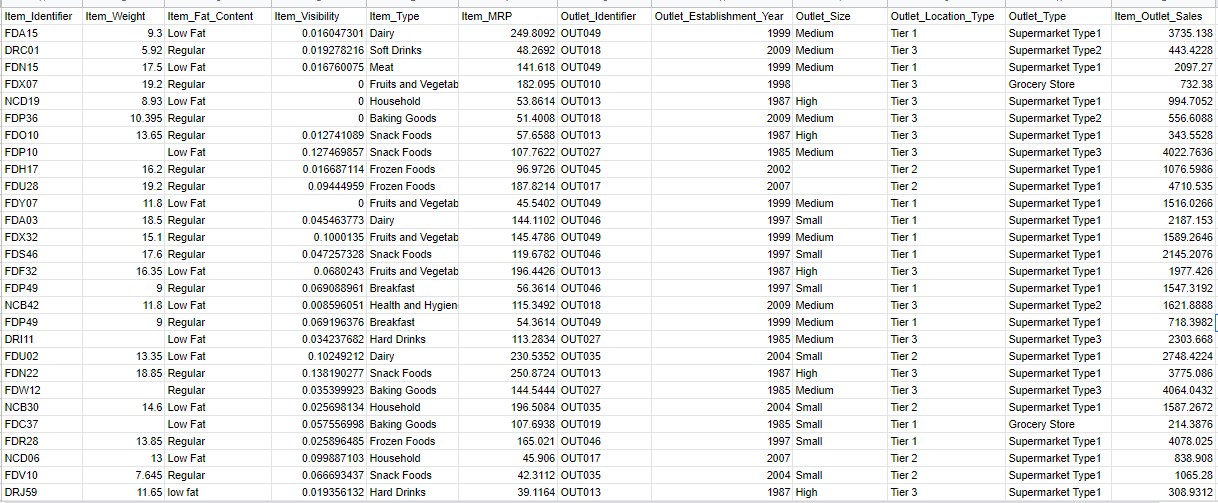
# Technical specifications

## Dataset

* + 1. **Dataset overview**

Consists of 12 different tables.

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 test data set.



* + 1. **Input schema**

|  |  |  |
| --- | --- | --- |
| **Feature name** | **Datatype** | **Null/Required** |
| Item Weight | float | Required |
| Item Fat Content | object | Required |
| Item Visibility | float | Required |
| Item Type | object | Required |
| Item MRP | float | Required |
| Outlet Identifier | object | Required |
| Outlet Establishment Year | int | Required |
| Outlet Size | object | Required |
| Outlet Location Type | object | Required |
| Outlet Type | object | Required |
| Item Outlet Sales | float | Required |

## Predicting Item Outlet Sales

* The system displays the item outlet sales based on entered input
* The User selects the product type.
* The system helps to provide the set of inputs required from the user.
* The user gives required information.
* The system should be able to predict the item outlet sales.

## Logging

We should be able to log every activity done by the user.

* The System identifies at what step logging required
* The System should be able to log each and every system flow.
* Developers can choose logging methods. You can choose database logging/ File logging as well.
* System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

## Deployment

1. Heroku



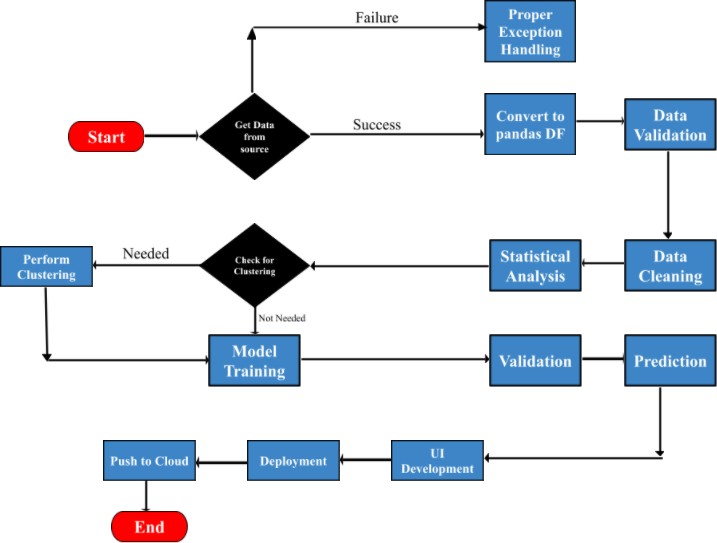
# Technology stack

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

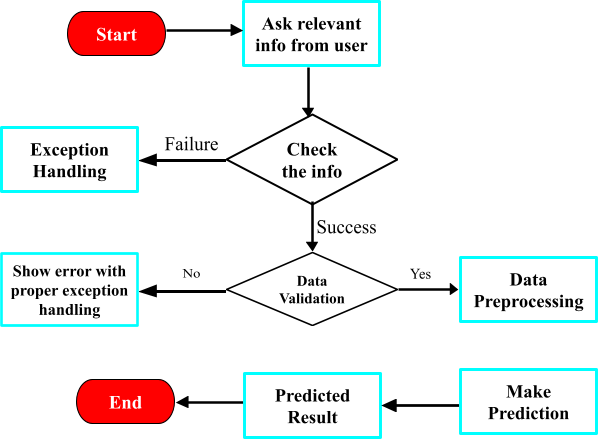
1. **Proposed Solution**

The solution proposed here is a Store Sales Prediction System which will help to predict the sales of the products in the stores based on the provided datasets and helps to estimate future sales by implementing the classic machine learning tasks like Data Exploration, Data Cleaning, Feature Engineering, Model Building and Model Testing and convenient machine learning algorithms.

# Model training/validation workflow



1. **User I/O workflow**



# Key performance indicators (KPI)

* The output will be provided in milli seconds
* Provides a good accuracy.
* Easy to access.
* Sales prediction can be done efficiently.

# Latency

# The output for the given input will be provided in milli seconds after clicking on the submit button.