Low Level Design (LLD)

Stores Sales Prediction

Revision Number: 1.6

Last date of revision: 23 Sep 2021

SOMAY

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| **23 SEP 2021** | 1.1 | First Draft | SOMAY |
| **23 SEP 2021** | 1.2 | Added Workflow chart | SOMAY |
| **23 SEP 2021** | 1.3 | Added Exception Scenarios Overall, Constraints | SOMAY |
| **23 SEP 2021** | 1.4 | Added user I/O flowchart | SOMAY |
| **23 SEP 2021** | 1.5 | Added dataset overview and updated user I/O flowchart. | SOMAY |
| **23 SEP 2021** | 1.6 | Restructure and reformat LLD | SOMAY |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contents**

[Document Version Control 2](#_bookmark0)

1. Abstract 4
2. Introduction 4
   1. Why this uses Low Level Document 4
   2. Scopes 5
   3. Constraint 5
   4. Risks 5
   5. Out of scopes 5
3. Technical 5
   1. Datasets 5
   2. Sales Data Overview 6
   3. Predicting Sales 6
   4. Deployment 7
4. Technology Stack 7
5. Proposed Solution 8
6. Model Training 8
7. User Input workflow 9
8. Exceptional Scenario 9
9. Test Cases 10
10. Performance 10

# Abstract

The purpose of this research is to construct a **sales prediction** model for retail **stores** using the deep learning approach, which has gained significant attention in the rapidly developing field of machine learning in recent years. Using such a model for analysis, an approach

to **store** management could be formulated.

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Stores Sales Prediction 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 stores sales month wise. Stores Sales is the part of Economy

* + - Contain a item weight, item fat content, item mrp, item sales outlet year
    - Allow access to evidence-based tools that providers can use to make decisions about a sale of the store.
    - Automate and streamline provider workflow

A Stores Sales Prediction contains item details, such as:

* + - Item sales
    - Sales profit
    - Item quantity
    - Item Location
    - Item Value

This project shall be delivered in two phases: Phase 1: All the functionalities with PyPi packages. Phase2: Integration of UI to all the functionalities.

## Scope

This software system will be a Web application This system will be designed to predict the Stores Sales and whether it is a compensated, primary or secondary item for better customer management, improved interventions, and more efficient Store sales resource allocation. More specifically, Early predicting of any preventable sales is important for better stores management. This system is designed to predict the Sales Prediction from store information such as item value, item location, item fat content, procedures.

## Constraints

The Stores Sales Prediction application must be user friendly, as automated as possibleand users should not be required to know any of the workings.

## 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. Sales dataset overview

The Sales dataset consists of a table with 5681 records and 11 features. Features are distributed as 7 continuous features and 4 categorical features.

## Input schema

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature name** | **Datatype** | **Size** | **Null/Required** |
| Item\_weight | int | 3 | Required |
|  |  |  |  |
|  |  |  |  |

* 1. Predicting Sale
* The system presents the set of inputs required from the user.
* The user gives required information.
* The system then predicts that the user is having sales is high or not.

Also, it tells whether a user is having compensated, primary or secondary dataset resources.

* 1. Deployment



# Technology stack

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

|  |  |
| --- | --- |
| **Deployment** | Heroku/AWS |

1. **Proposed Solution**

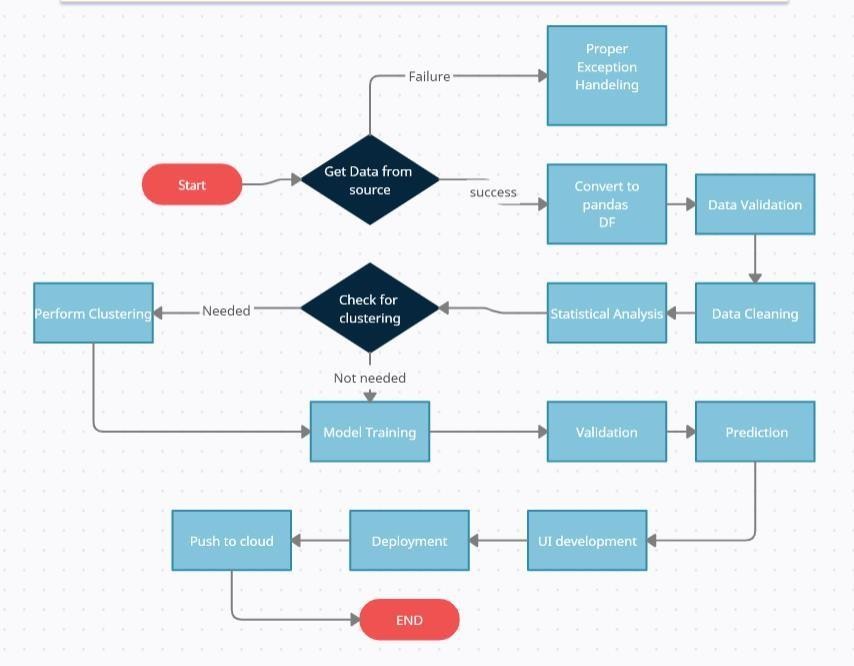
### The proposed solution for this project is Machine learning algorithms can be implemented to predict the stores sales. Considering various features like item type, outlet size, item mrp, item year as inputs from the web app, the implemented classification model will predict the output as store sales

Here, we have used Random Forest Classifier to predict whether the stores sales is high or not.

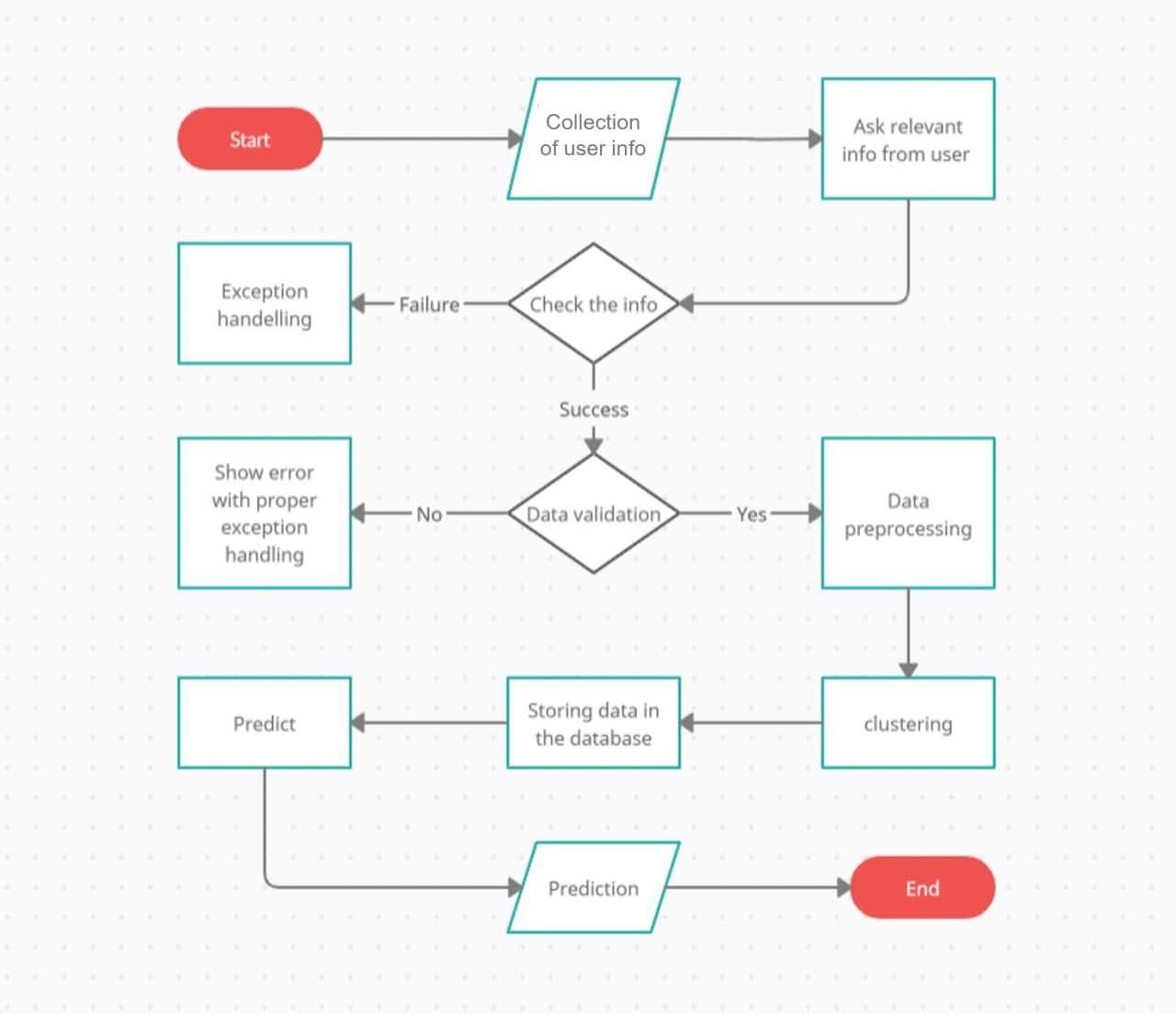
### However, drawing a baseline model is important since it tells us how well other models have performed compared to base model. Here, the base model for Store sales dataset isLinear regression

1. Baseline Model: Linear Regression
2. Actual Model: Random Forest

# Model training/validation Workflow



1. **User I/O workflow**



# Exceptional scenario

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Exception** | **Mitigation** | **Module** |
| **15 Sep 2021** | 1.1 | First Draft | Somay |
| **15 Sep 2021** | 1.2 | Added Workflow chart | Somay |

1. **Test cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Steps to perform test case** | **Module** | **Pass/Fail** |
| 1. | Predict the sales using random forest | The proposed solution for this project is Machine learning algorithms can be implemented to predict the stores sales. Considering various features like item type, outlet size, item mrp, item year as inputs from the web app, the implemented classification model will predict the output as store sales Here, we have used Random Forest Classifier to predict whether the stores sales is high or not. | pass |

# Performance

### We can observe that the accuracy of the predicted output was seen at 80% using Random Forest classifier. Other classification models such as logistic regression and decision tree have given good accuracy above 80