Architecture Design

SALES PREDICTION TOOL

Written by	SACHIN SHARMA MUNENDRA KUMAR KUSHWAHA
Version	1.0
College Name	D.S. Degree college Aligarh

Index

Content	Page No
Abstract	4
1. Introduction	4

1.1 What is Architecture Design?	4
1.2 Scope	4
1.3 Constraints	4
2. Technical Specification	5
2.1 Dataset	5
2.2 Logging	7
2.3 New Feature Generation	7
2.4 Deployment	7
3. Technology Stack	8
4. Proposed Solution	8
5. Architecture	8
5.1 Architecture Description	9
6. Model Training/Validation Workflow	11
7. User Input/Output Workflow	12

Abstract

Machine Learning is a class of methods that allows software to improve its accuracy in predicting events without having to be explicitly coded. Machine learning is based on the creation of models and algorithms that can take in data and apply statistical analysis to predict an output while updating outputs as new data becomes available. These models can be used in a variety of situations and trained to match management's expectations so that

precise procedures can be taken to meet the organization's goals. The case of Big Mart, a one-stop-shopping-center, has been explored in this paper in order to predict the sales of various types of things and to comprehend the effects of various circumstances on the items' sales. Taking numerous features of a dataset into consideration Results with high degrees of accuracy are obtained using the dataset collected for Big Mart and the approach used to build a predictive model, and these observations can be used to make decisions to boost sales.

1. INTRODUCTION

1.1 What is the definition of architecture design?

The purpose of Architecture Design (AD), also known as a low-level architecture document, is to provide the internal design of the actual programme code for the 'Bike Share Prediction System.' Class diagrams with methods and relationships between classes are described by AD in the programme specification. It explains the modules in such a way that the programmer can code the programme directly from the document.

1.2 Scope

Architecture Design (AD) is a component-level design approach that is refined step by step. This method can be used to create data structures, software, architecture, source code, and, eventually, performance algorithms. Overall, data organisation can be determined at the requirement analysis phase and then refined throughout the data design phase. As well as the entire workflow.

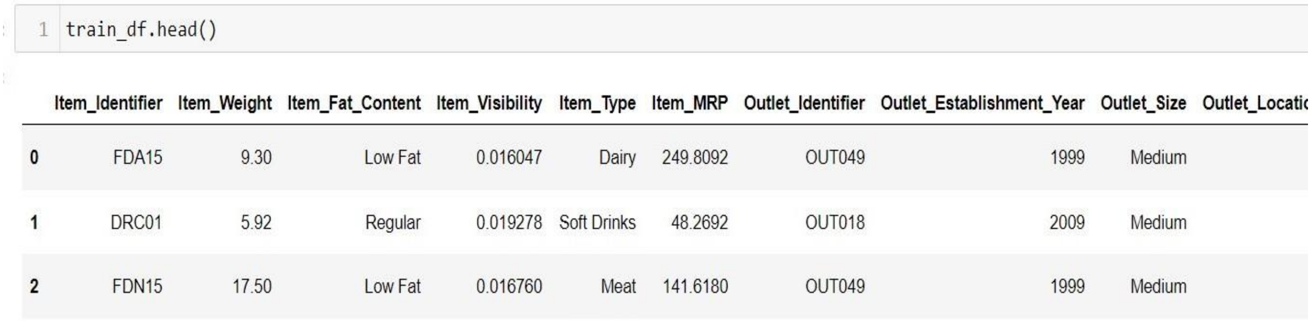
1.3 Constraints

We only predict the expected casual and registered customers based on the weather condition and date information.

2. Technical Specification

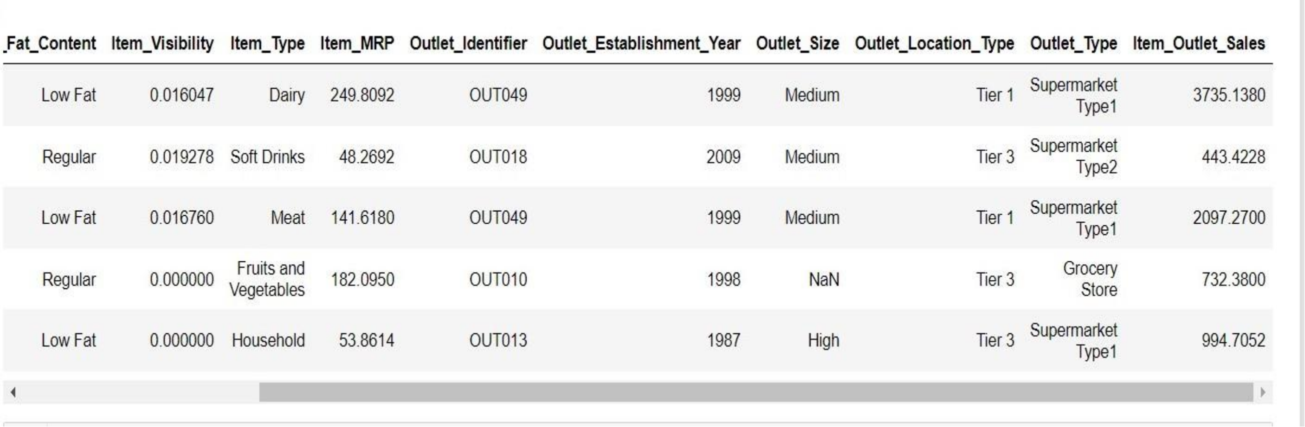
2.1 Dataset

Big Mart's data scientists collected sales data of their 10 stores situated at different locations with each store having 1559 different products as per data collection. Using all the observations it is inferred what role certain properties of an item play and how they affect their sales.Thedatasetlookslike as follow:



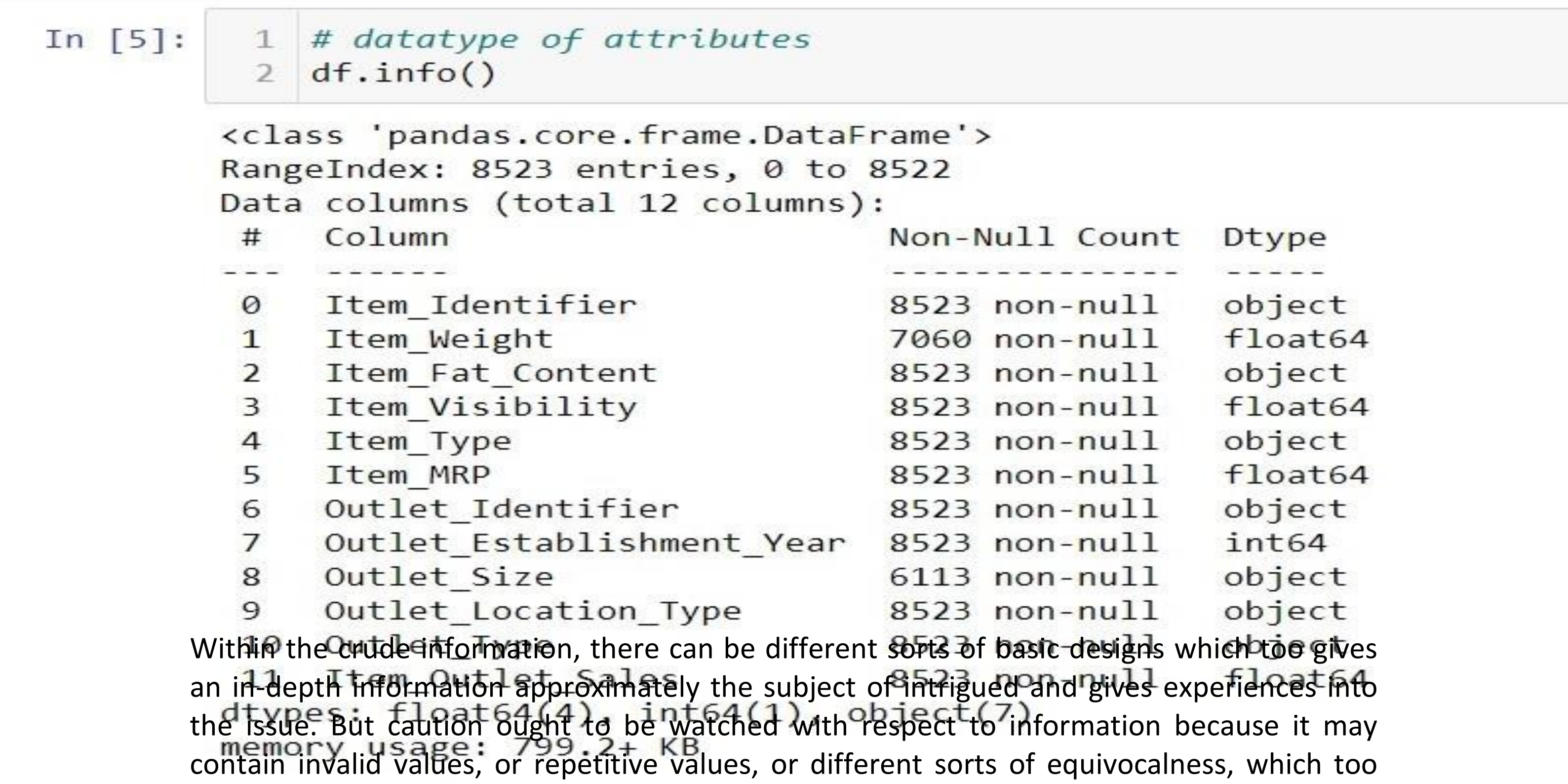
```
train_df.head()
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Locatic
0	FDA15	9.30	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	
1	DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	
2	FDN15	17.50	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	



Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Outlet_Sales
Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	3735.1380
Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	443.4228
Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	2097.2700
Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	Tier 3	Grocery Store	732.3800
Low Fat	0.000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	994.7052

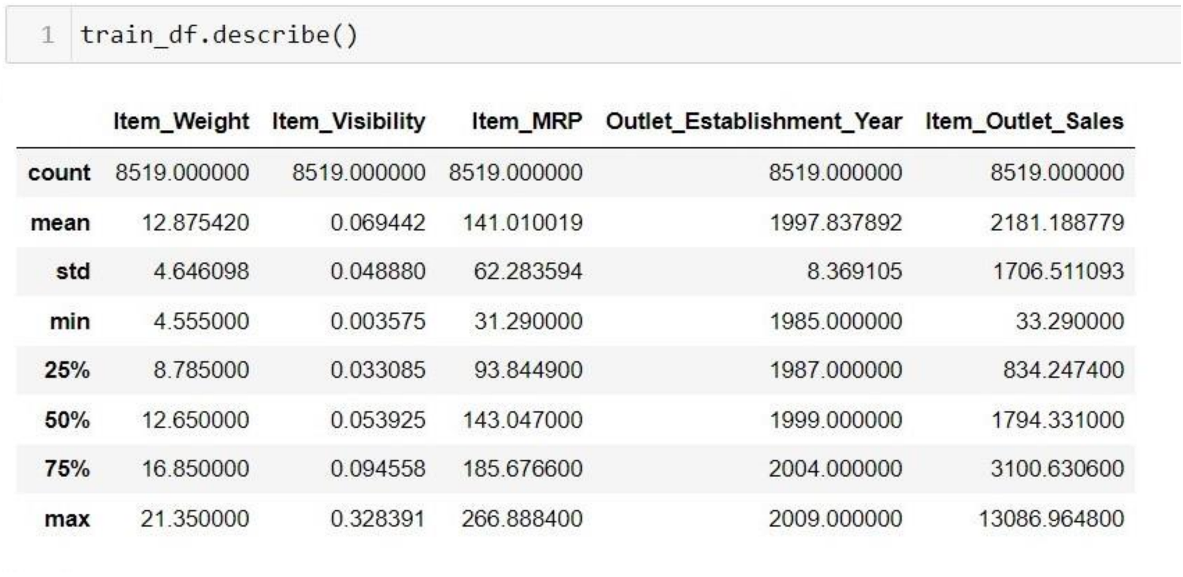
The data set consists of various data types from integer to floating to object as shown in Fig.



```
In [5]:  # datatype of attributes
          df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8523 entries, 0 to 8522
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Item_Identifier            8523 non-null   object
 1   Item_Weight                7060 non-null   float64
 2   Item_Fat_Content           8523 non-null   object
 3   Item_Visibility            8523 non-null   float64
 4   Item_Type                  8523 non-null   object
 5   Item_MRP                   8523 non-null   float64
 6   Outlet_Identifier          8523 non-null   object
 7   Outlet_Establishment_Year  8523 non-null   int64
 8   Outlet_Size                6113 non-null   object
 9   Outlet_Location_Type       8523 non-null   object
 10  Outlet_Type                8523 non-null   object
 11  Item_Outlet_Sales          8523 non-null   float64
dtypes: float64(4), int64(1), object(7)
memory usage: 799.2+ KB
```

Within the crude information, there can be different sorts of basic designs which too gives an in-depth information approximately the subject of intrigued and gives experiences into the issue. But caution ought to be watched with respect to information because it may contain invalid values, or repetitive values, or different sorts of equivocalness, which too requests pre-processing of information. The dataset ought to hence be investigated as much as possible. Various variables critical by factual implies like cruel, standard deviation, middle, check of values and greatest esteem, etc. are appeared underneath for numerical attributes.



```
train_df.describe()
```

	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
count	8519.000000	8519.000000	8519.000000	8519.000000	8519.000000
mean	12.875420	0.069442	141.010019	1997.837892	2181.188779
std	4.646098	0.048880	62.283594	8.369105	1706.511093
min	4.555000	0.003575	31.290000	1985.000000	33.290000
25%	8.785000	0.033085	93.844900	1987.000000	834.247400
50%	12.650000	0.053925	143.047000	1999.000000	1794.331000
75%	16.850000	0.094558	185.676600	2004.000000	3100.630600
max	21.350000	0.328391	266.888400	2009.000000	13086.964800

Preprocessing of this dataset incorporates doing examination on the autonomous factors like checking for invalid values in each column and after that supplanting or filling them with backed fitting information sorts so that analysis and demonstrate fitting isn't prevented from their way to exactness. Appeared over are a few of the representations gotten by utilizing Pandas devices which tell approximately variable tally for numerical columns and show values for categorical columns. Maximum and least values in numerical columns, at the side their percentile values for middle, play an critical figure in choosing which esteem to be chosen at need for assist investigation assignments and examination. Information sorts of distinctive columns are utilized assist in name handling and a one-hot encoding plot amid the modelbuilding.

2.2 Logging

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

- The system identifies at which step logging require.
- The system should be able to log each and every system flow.
- Developers can choose logging methods. Also can choose database logging.
- The system should be not be hung even after using so much logging. Logging just because we can easily debug issuing so logging is mandatory to do.

2.3 DataBase

The system needs to store every request into the database and we need to store it in such a way that it is easy to retain and look into the records.

The system should capture every data that any user gave and the prediction that has been made by that input.

2.4 Deployment

For the hosting of the project, we will use heroku

HEROKU

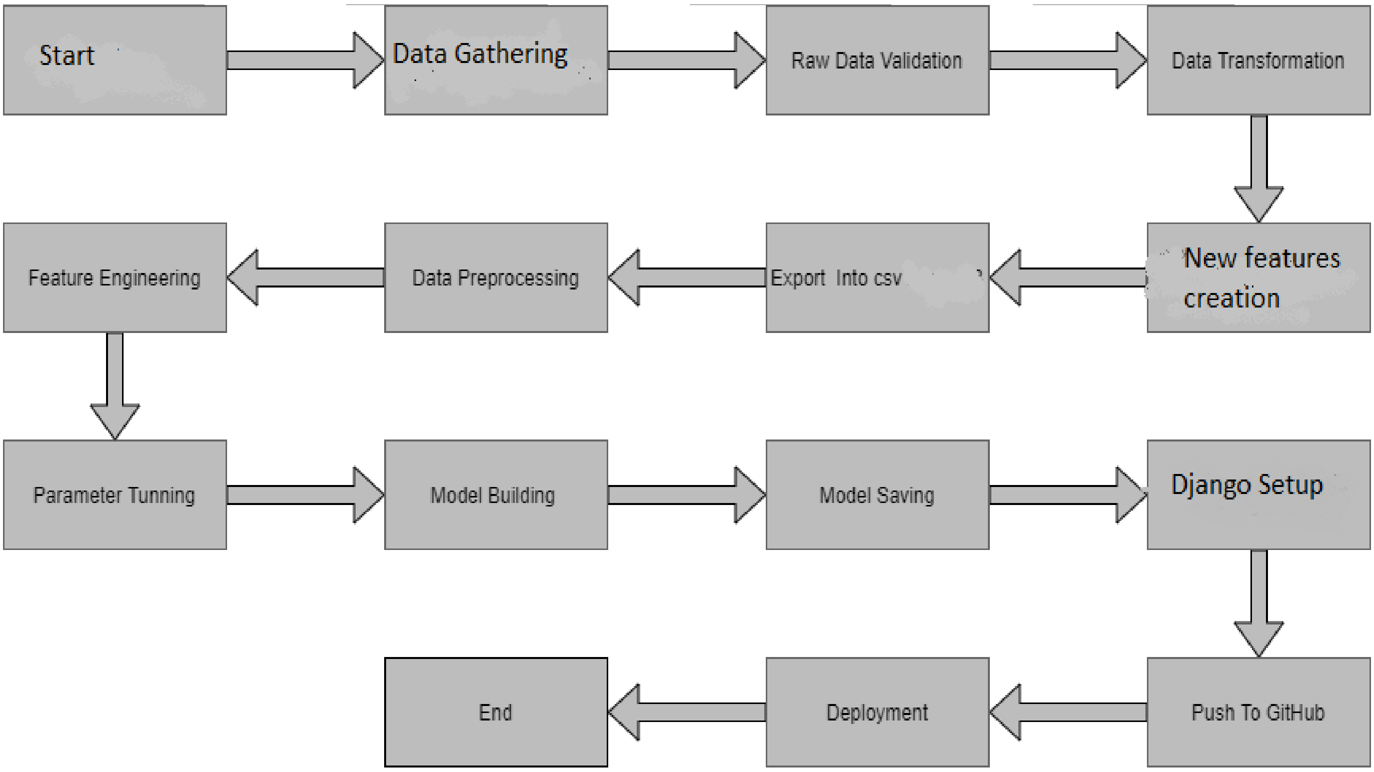
3. Technology Stack

Front End	HTML/JavaScript
Backend	Python/ Django
Deployment	Heroku

4. Proposed Solution

We are going utilize performed EDA to discover the imperative connection between distinctive traits and will use a machine-learning calculation to foresee the longer term deals request. The client will be filled the specified include as input and will get comes about through the internet application. The framework will get highlights and it'll be passed into the backend where the highlights will be approved and preprocessed and after that it'll be passed to a hyperparameter tuned machine learning show to anticipate the ultimate result.

5. Architecture



5.1 Data Gathering

Data source: https://www.kaggle.com/brijbhushannanda1979/bigmart-sales-data

Train and Test data are stored in .csv format.

5.2 Raw Data Validation

After information is stacked, different sorts of approval are required some time recently we continue assist with any operation. Validations like checking for zero standard deviation for all the columns, checking for total lost values in any columns, etc. These are required since The traits which contain these are of no utilize. It'll not play part in contributing to the deals of an thing from particular outlets. Like on the off chance that any quality is having zero standard deviation, it implies that's all the values are the same, its cruel is zero. This demonstrates that either the deal is expanding or diminish that quality will stay the same. So also, in the event that any quality is having full missing values, at that point there's no utilize in taking that property into an account for operation. It's pointless expanding the chances of dimensionality curse.

5.3 Data Transformation

Before sending the data into the database, data transformation is required so that data are converted into such form with which it can easily insert into the database. Here, the 'Item Weight' and "Outlet Type' attributes contain the missing values. So they are filled in both the train set as well as the test set with supported appropriate data types.

5.4New Feature Generation

We can derive new item cateogory from item type and create mrp categories as mrp bin

5.5 Data Preprocessing

In information preprocessing all the forms required some time recently sending the information for demonstrate building are performed. Like, here the 'Item Visibility' traits are having a few values break even with to 0, which isn't fitting since on the off chance that an item is display within the advertise, at that point how its perceivability can be 0. So, it has been supplanted with the normal esteem of the thing perceivability of the particular 'Item Identifier' category. Unused qualities were included named ''Outlet years", where the given foundation year is subtracted from the current year. A modern "Item Type" attribute was included which fair takes the primary two characters of the Thing Identifier which shows the sorts of the things. At that point mapping of "Fat content" is done based on 'Low', 'Reg' and 'Non-edible'.

5.6 Feature Engineering

After preprocessing it was found that some of the attributes are not important to the item sales for the particular outlet. So those attributes are removed. Even one hot encoding is also performed to convert the categorical features into numerical features.

5.7 Parameter Tuning

Parameters are tuned using Randomized searchCV. Four algorithms are used in this problem, Linear Regression, Gradient boost, Random Forest, and XGBoost regressor. The parameters of all these 4 algorithms are tunned and passed into the model.

5.8 Model Building

After doing all kinds of preprocessing operations mention above and performing scaling and hyperparameter tuning, the data set is passed into all four models, Linear Regression, Gradient boost, Random Forest, and XGBoost regressor. It was found that Gradient boost performs best with the smallest RMSE value i.e. 587.0 and the highest R2 score equals 0.55. So 'Gradient boost' performed well in this problem.

5.9 Model Saving

Model is saved using pickle library in `.pkl` format.

5.10Django Setup for Data Extraction

After saving the model, the API building process started using Django.Web application creation was created here. Whatever the data user will enter and then that data will be extracted by the model to predict the prediction of sales, this is performed in this stage.

5.12 GitHub

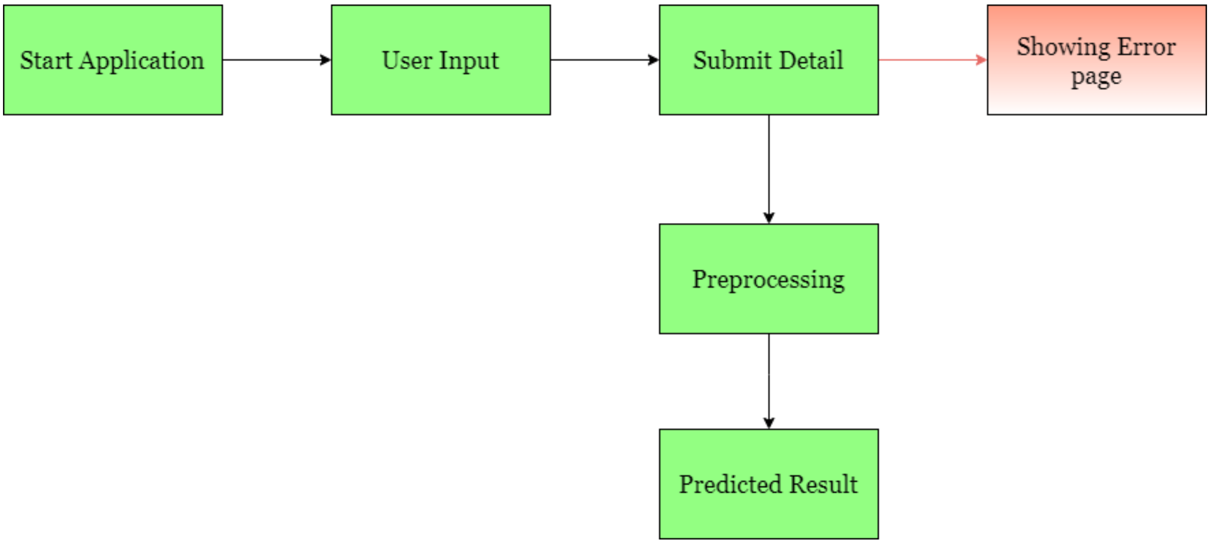
The whole project directory will be pushed into the GitHub repository.

5.13 Deployment

The cloud environment was set up and the project was deployed from GitHub into the Heroku cloud platform.

App link- **https://salepredict.herokuapp.com/**

7. User Input / Output Workflow.


Start Application
User Input
Submit Detail
Showing Error page
Preprocessing
Predicted Result