ARCHITECTURE DESIGN (AD)

Restaurant Rating Prediction

REVISION NUMBER – 1.2

Last date of Revision: 25/03/2023

Authored by: Ankush choudhary

DOCUMENT VERSION CONTROL

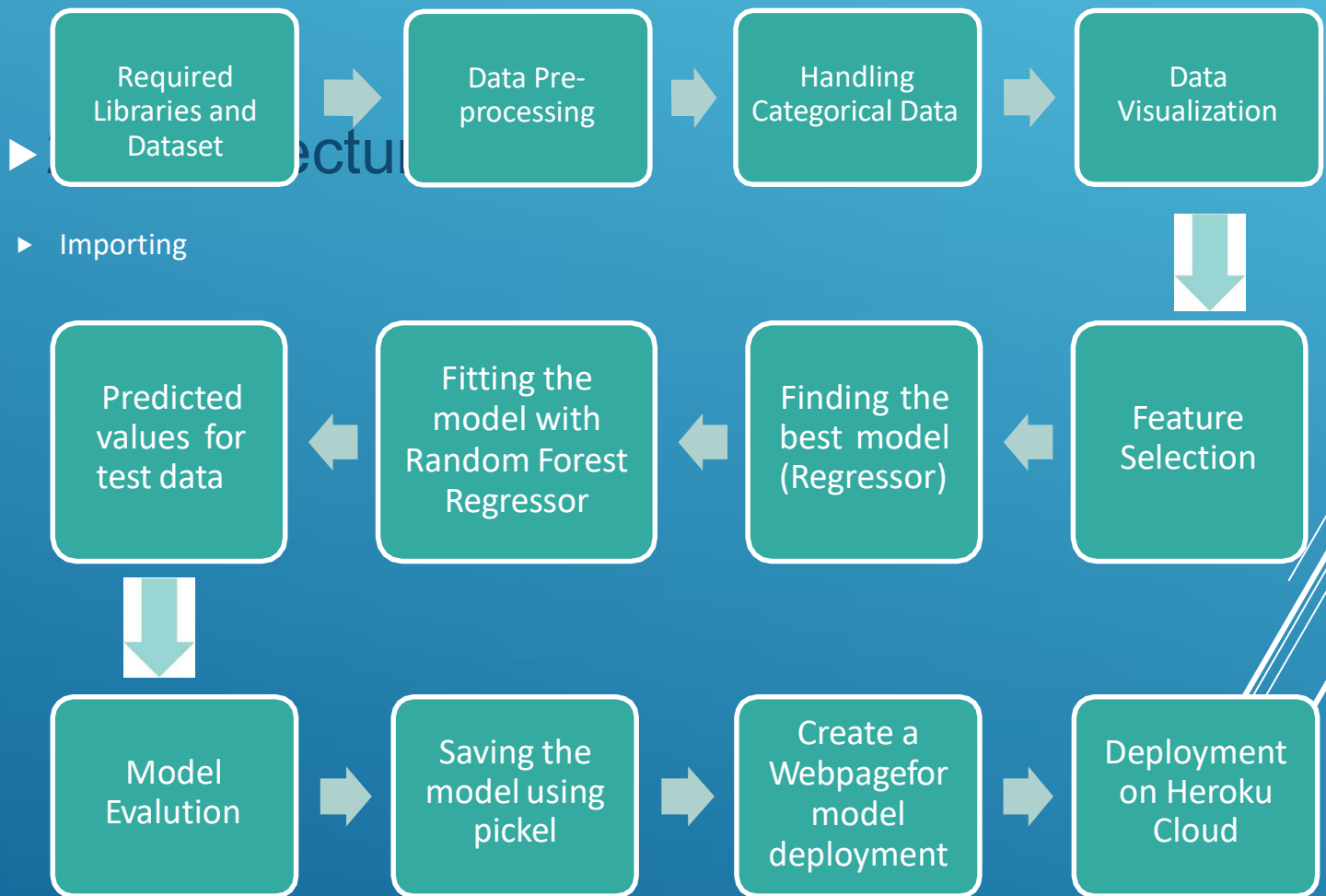| Date	Version	Description	Author
2/04/2023	1.0	Abstract Introduction Architecture	Ankush
3/04/2023	1.1	Architectural Design	Ankush

CONTENTS

ABSTRACT

The basic idea of analyzing the Zomato dataset is to get a fair idea about the factors affecting the establishment of different types of restaurants at different places in Bengaluru, aggregate rating of each restaurant, Bengaluru being one such city has more than 12,000 restaurants with restaurants serving dishes from all over the world. With each day new restaurants opening the industry hasn't been saturated yet and the demand is increasing day by day. Bengaluru being an IT capital of India, most of the people here are dependent mainly on the restaurant food as they don't have time to cook for themselves. With such an overwhelming demand for new restaurants, it has become important to study the ratings of restaurants.

1. INTRODUCTION

- ▸ 1.1 What this Architecture Design Document ?
  - ▸ The main objective of the Architecture design documentation is to provide the internal logic understanding of the flight fare prediction code. The Architecture design documentation is designed in such a way that the programmer can directly code after reading each module description in the documentation.

- ▸ Architecture

- ▸ Importing

Required Libraries and Dataset	→	Data Pre-processing	→	Handling Categorical Data	→	Data Visualization

⬇

Predicted values for test data	←	Fitting the model with Random Forest Regressor	←	Finding the best model (Regressor)	←	Feature Selection

⬇

Model Evalution	→	Saving the model using pickel	→	Create a Webpagefor model deployment	→	Deployment on Heroku Cloud

3. ARCHITECTURE DESIGN

1. Data Collection

The data for this project is collected from the Kaggle Dataset, the URL for the dataset is https://www.kaggle.com/datasets/himanshupoddar/zomato-bangalore-restaurants?resource=download

2. Data Description

The dataset contains 17 variables all of which were scrapped from the Zomato website. The dataset contains details of more than 50,000 restaurants in Bengaluru in each of its neighborhood. The total size of dataset is approximately 547 MB.

Variable	Type	#Unique Values	Description
url	object	51,717	contains the url of the restaurant in the zomato website
address	object	11,495	contains the address of the restaurant in Bengaluru
name	object	8,792	contains the name of the restaurant
online_order	category	2	whether online ordering is available in the restaurant or not
book_table	category	2	table book option available or not
rate	object	64	contains the overall rating of the restaurant out of 5
votes	int	2328	contains total number of rating for the restaurant as of the above mentioned date
phone	object	64	contains the phone number of the restaurant
location	category	93	contains the neighborhood in which the restaurant is located
rest_type	category	93	restaurant type
dish_liked	object	5271	dishes people liked in the restaurant
cuisines	object	2723	food styles, separated by comma
approx_cost(for two people)	float	70	contains the approximate cost for meal for two people
reviews_list	object	22513	list of tuples containing reviews for the restaurant, each tuple consists of two values, rating and review by the customer
menu_item	object	9098	contains list of menus available in the restaurant
listed_in(type)	category	7	type of meal
listed_in(city)	category	30	contains the neighborhood in which the restaurant is listed

3. **Importing data into Database**

Created associate API for the transfer of the info into the Cassandra info, steps performed are:

- Connection is created with the info.
- Created a info with name ZomatoInfo.
- cqlsh command is written for making the info table with needed parameters.
- And finally, a cqlsh command is written for uploading the Knowledge Set into data table by bulk insertion.

4. **Exporting Data from Database**

In the above created API, the download URL is also being created, which downloads the data into a csv file format.

5. **Data Preprocessing**

- Checked for info of the Dataset, to verify the correct datatype of the Columns.
- Checked for Null values, because the null values can affect the accuracy of the model.
- Converted all the illegal values into legal values.
- Performed Labeled encoding and One hot Encoding on the desired columns.
- Checking the distribution of the columns to interpret its importance.
- Now, the info is prepared to train a Machine Learning Model.

6. **Modelling Creation**

After preprocessing the data, we visualize our data to gain insights and then these insights are randomly spread and split into two parts, train and test data. After splitting the data, we use Random Forest Regressor to model our data to predict the Restaurant Rating

```
┌─────────────┐     ┌─────────────┐     ┌─────────────┐     ┌─────────────┐
│  Importing  │     │             │     │             │     │             │
│  Required   │ ──▶ │  Data Pre-  │ ──▶ │  Handling   │ ──▶ │    Data     │
│ Libraries and│     │ processing  │     │Categorical Data│  │Visualization│
│   Dataset   │     │             │     │             │     │             │
└─────────────┘     └─────────────┘     └─────────────┘     └─────────────┘
                                                                   │
                                                                   ▼
┌─────────────┐     ┌─────────────┐     ┌─────────────┐     ┌─────────────┐
│  Predicted  │     │ Fitting the │     │ Finding the │     │             │
│ values  for │ ◀── │ model with  │ ◀── │ best  model │ ◀── │   Feature   │
│  test data  │     │Random Forest│     │ (Regressor) │     │  Selection  │
│             │     │  Regressor  │     │             │     │             │
└─────────────┘     └─────────────┘     └─────────────┘     └─────────────┘
       │
       ▼
┌─────────────┐     ┌─────────────┐     ┌─────────────┐     ┌─────────────┐
│             │     │ Saving the  │     │  Create a   │     │ Deployment  │
│   Model     │ ──▶ │ model using │ ──▶ │ Webpagefor  │ ──▶ │ on Heroku   │
│  Evalution  │     │   pickel    │     │   model     │     │   Cloud     │
│             │     │             │     │ deployment  │     │             │
└─────────────┘     └─────────────┘     └─────────────┘     └─────────────┘
```
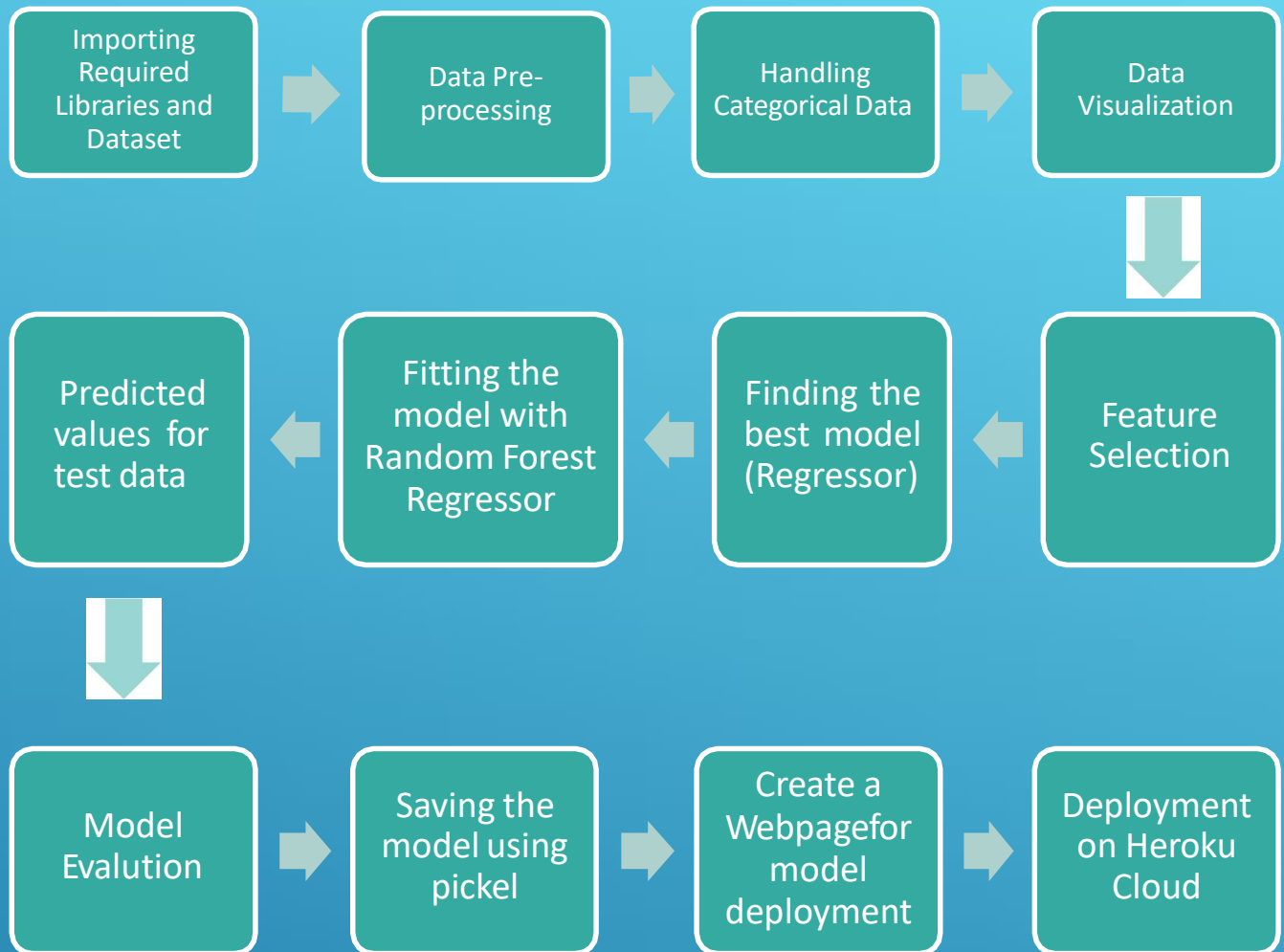
7. UI Integration

Both CSS and HTML files are being created and are being integrated with the created machine learning model. All the required files are then integrated to the app.py file and tested locally.

8. Data from User

The data from the user is retrieved from the created HTML web page.

9. Data Validation

The data provided by the user is then being processed by app.py file and validated. The validated data is then sent to the prepared model for the prediction.

10. Rendering the Results

The data sent for the prediction is then rendered to the web page.

11. Deployment

The tested model is then deployed to Heroku. So, users can access the project from any internet devices.