Architecture Design (AD)

#### REVISION NUMBER – 1.0

##### Last date of Revision: 07/08/2022 Authored by: Samir Saiyed



Date Version Description Author

06/08/2022 1.0 Abstract Introduction Architecture

Samir Saiyed

07/08/2022 1.1 Architectural Design Samir Saiyed



Document Version Control 2

[Abstract 4](#_TOC_250013)

1. [Introduction 5](#_TOC_250012)
   1. Why this Architecture Design Document ? 5
2. [Architecture 5](#_TOC_250011)
3. [Architecture Design 5](#_TOC_250010)
   1. [Data Collection 6](#_TOC_250009)
   2. [Data Description 6](#_TOC_250008)
   3. [Importing Data into Database 7](#_TOC_250007)
   4. [Exporting Data from Database 7](#_TOC_250006)
   5. [Data Preprocessing 7](#_TOC_250005)
   6. Modeling Process 7
   7. [UI Integration 8](#_TOC_250004)
   8. [Data from User 8](#_TOC_250003)
   9. [Data Validation 8](#_TOC_250002)
   10. [Rendering the Results 9](#_TOC_250001)
   11. [Deployment 9](#_TOC_250000)

# 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. **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.

**2. Architecture**

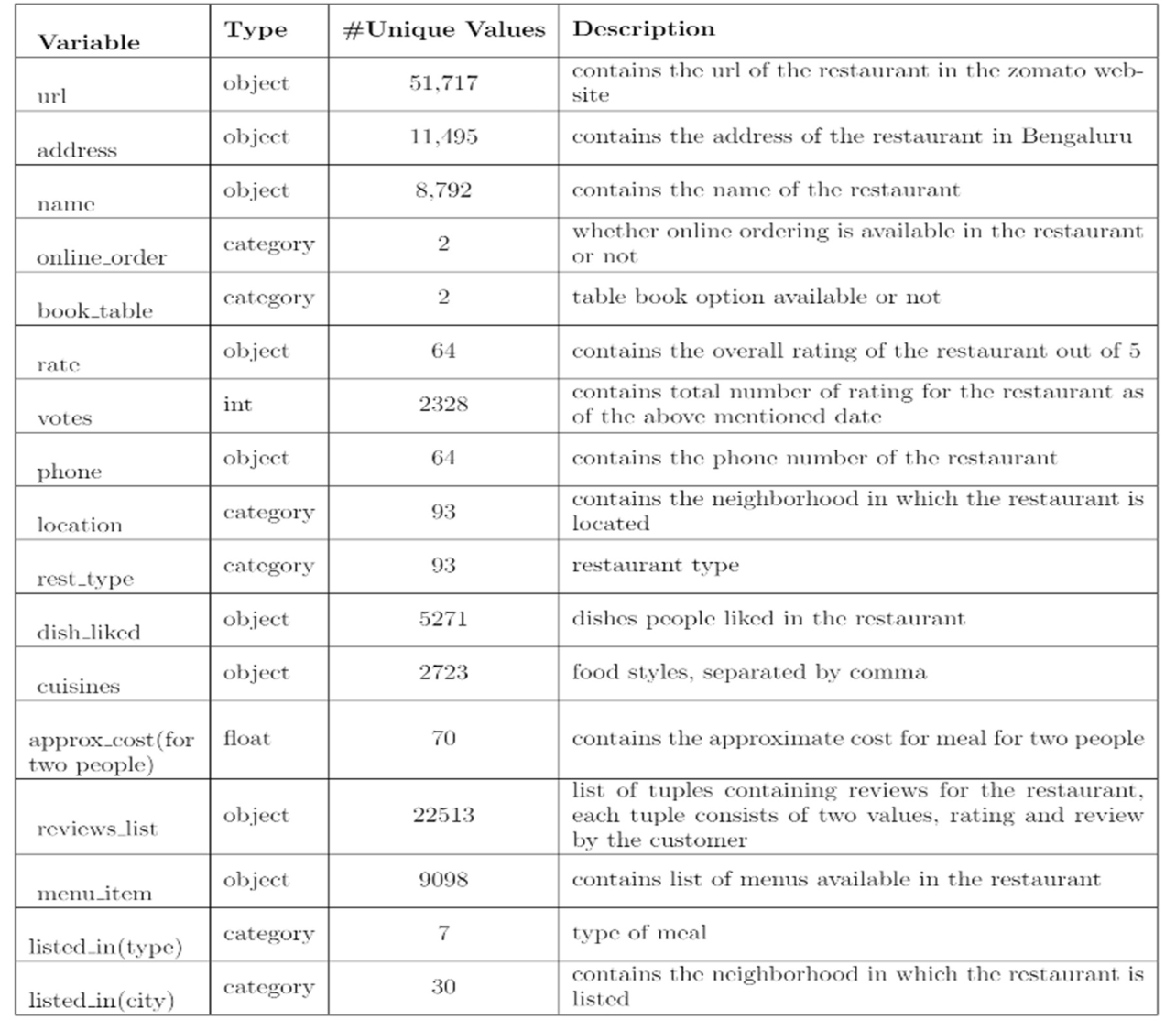
****

1. **Architecture Design**

## 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-](http://www.kaggle.com/datasets/himanshupoddar/zomato-) bangalore-restaurants?resource=download

## 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.

## 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.

## 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.

## 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.
  1. **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



## 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.

## Data from User

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

## 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.

## Rendering the Results

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

## Deployment

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