RESTAURANT RATING

PREDICTION(LLD)

Revision Number: 1.0

Last date of revision: 08/07/2022

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 08/07/2022 | 1 | Initial LLD — V1.0 | Priya Ganesan |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Document Version Control 2](#_Toc108189613)

[1. Introduction 4](#_Toc108189614)

[1.1 Why this Low-Level Design Document? 4](#_Toc108189615)

[1.2 Scope 4](#_Toc108189616)

[2. Architecture 5](#_Toc108189617)

[2.1 Architecture details 5](#_Toc108189618)

[3. Class diagram 6](#_Toc108189619)

[3.1 Training Class diagram 6](#_Toc108189620)

[3.2 Prediction Class Diagram 7](#_Toc108189621)

[4. Data Description 8](#_Toc108189622)

[4.1 Column details 8](#_Toc108189623)

# Introduction

## 1.1 Why this Low-Level Design Document?

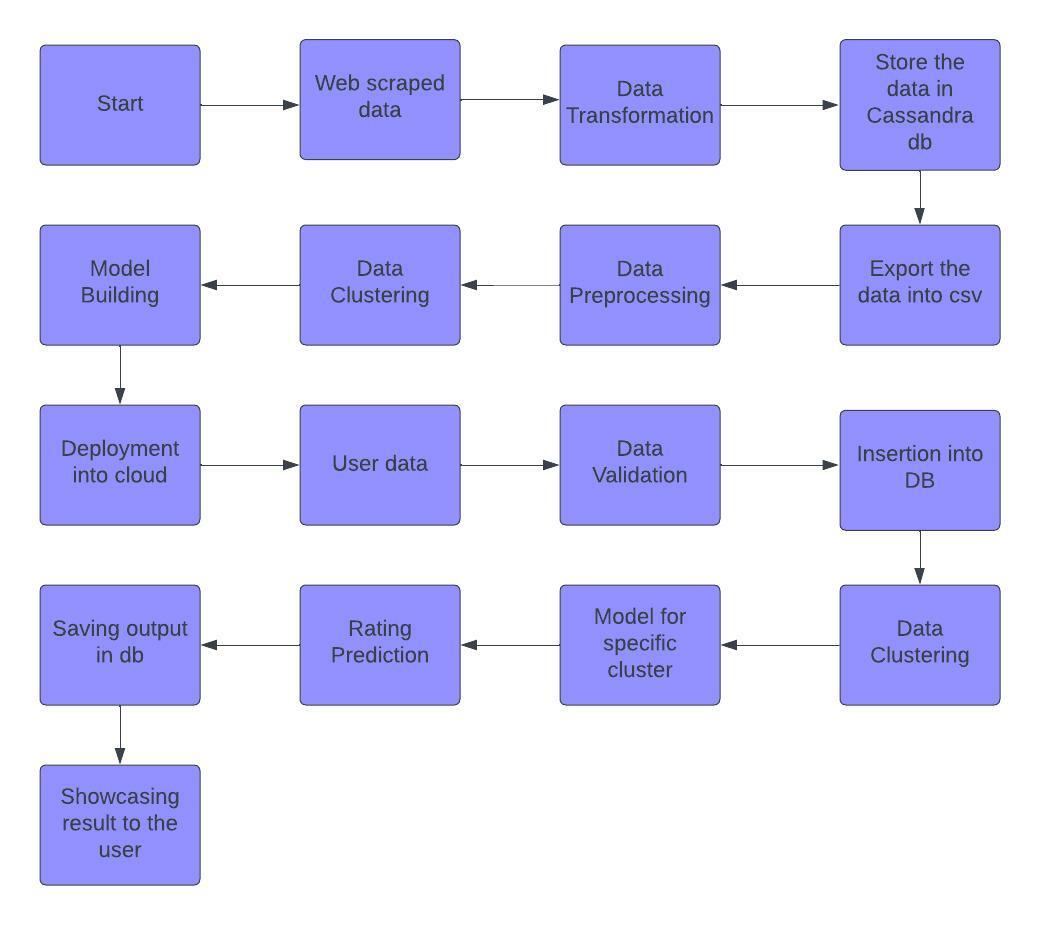
The goal of low level design document (LLDD) is to give the internal logical design of Restaurant Rating Prediction Project. Low level design is created based on the high level design. LLD document describes the class diagrams with the methods and relations between the classes.

## 1.2 Scope

This document contains contain class diagram, low level architecture and the data description for the project. When a new member joins the team, the member can understand the project very easily using this Low Level design document.

# 

# Architecture

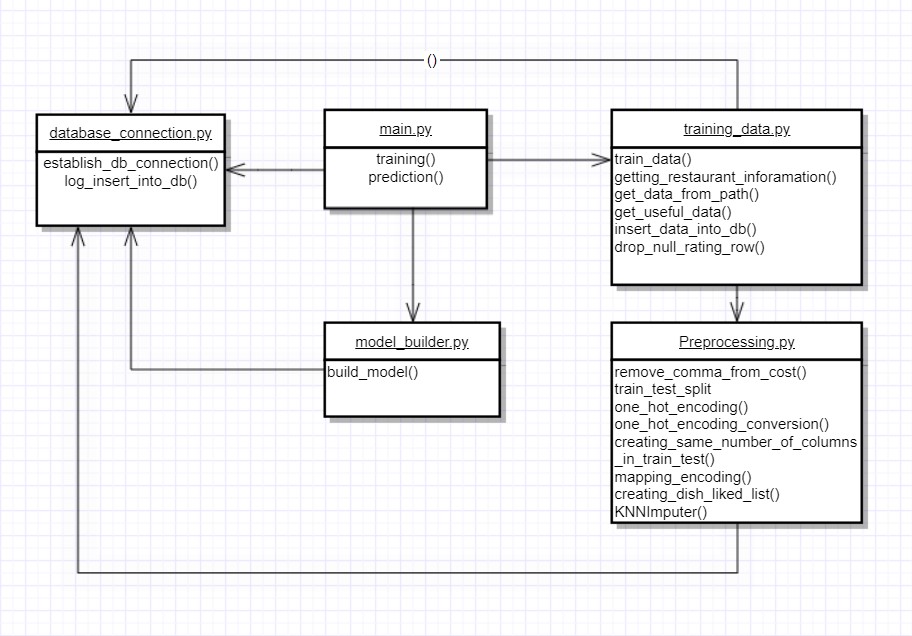


## 2.1 Architecture details

* The web scrapping is done in the Zomato website and the data is restaurant data.
* Data transformation like replace missing values with na, checking if all the data is valid data is done and it is stored in Cassandra db
* From Cassandra db, import of data is done in csv format.
* Preprocessing steps like finding null values, imputing the null values, normalizing the data is done
* Then different clusters are created for the model so that we can apply different models for different clusters.
* Model is build on different clusters of the data. Different models like xgboost, random forest, decision tree is built and the better accuracy model is selected for each cluster
* Then deployment in the cloud takes place
* The user can give input from the ui and based on the input prediction is made
* The rating prediction is showed to the user and user can decide with those insights.

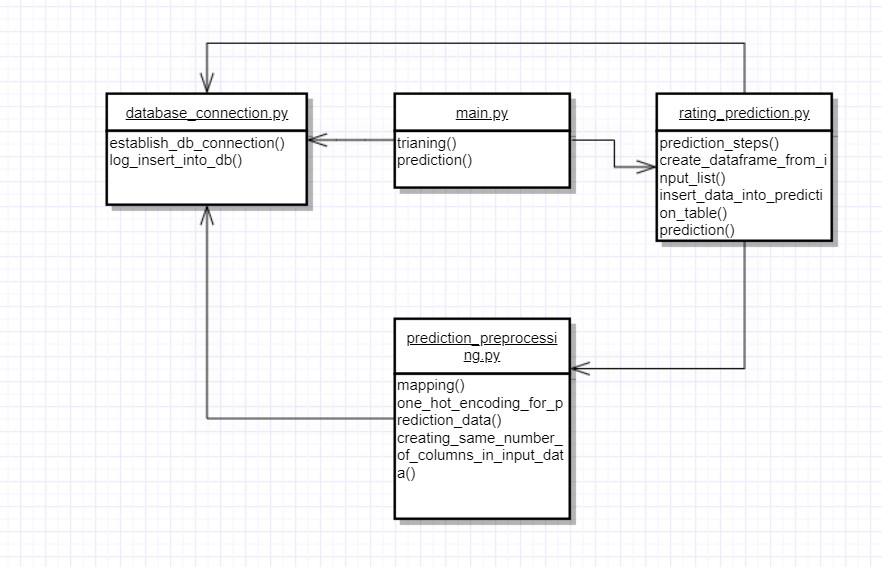
# Class diagram

## 3.1 Training Class diagram



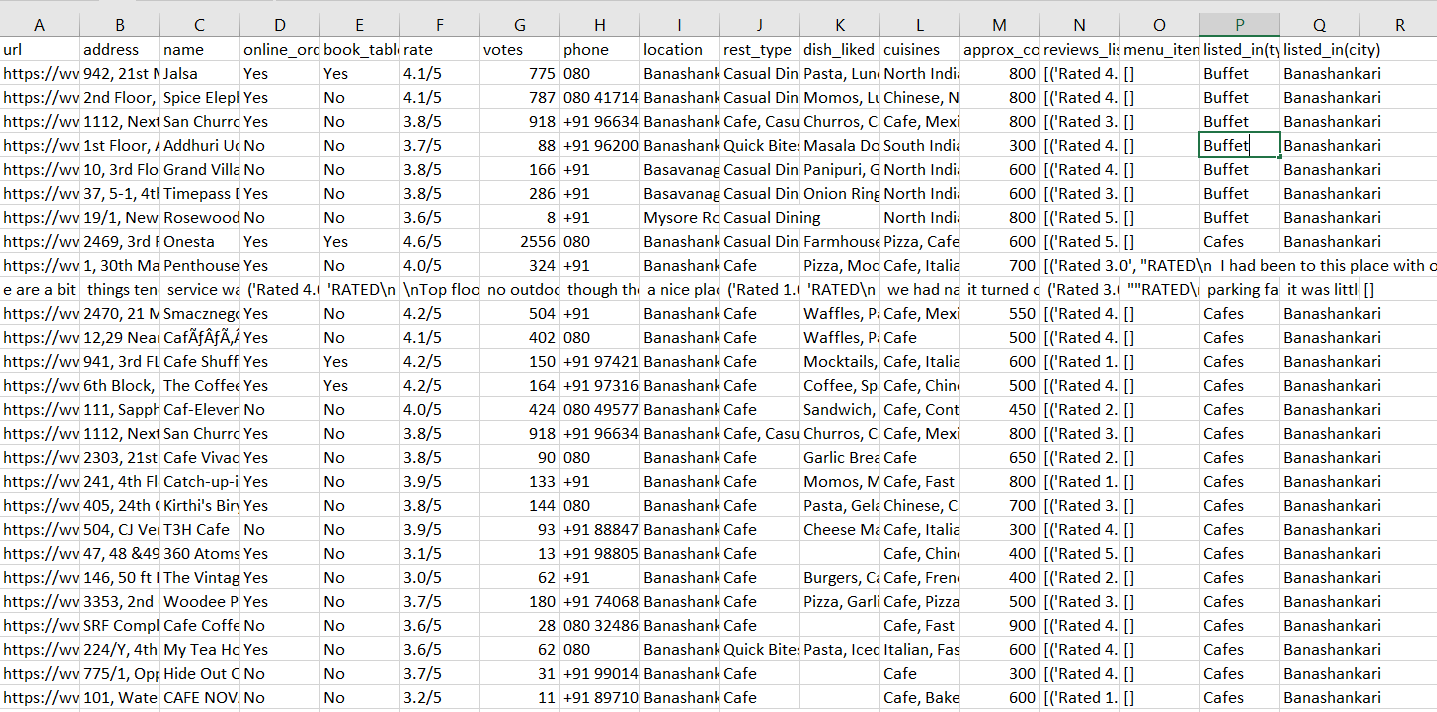
## 

## 3.2 Prediction Class Diagram



# Data Description

This data is collected from Bengaluru restaurants. The data looks like this :



Data consists of 51717 rows, 17 columns. Description of the data as follows :

## 4.1 Column details

1. url
2. address
3. name
4. online\_order : This column explains if online\_order accepted in the restaurant or not
5. book\_table : This column explains if table booking option available in the restaurant or not
6. Rate : This is the rating of that particular restaurant. This acts as a label for the training and while prediction we need to predict the rating with other features.
7. Votes : This explains total number of votes for the restaurant
8. Phone : This column includes phone number of the restaurant
9. Location : The locality of the restaurant is available in this column
10. Rest\_type : This column has the information about the restaurant type. There are different types of restaurants like Café, fine dining etc
11. Dish\_liked : This column give the details about the dish liked in the restaurant
12. Cuisines : This column contains information about various cuisines available in the restaurant
13. Approx\_cost(for two people) : This column explains the cost for 2 people in the restaurant
14. Reviews\_list : List of reviews available for the restaurant
15. Menu\_item : Menu of the restaurant
16. Listed\_in(type) : Type of the restaurant whether it is Buffet, Café, Delivery or dine out kind of restaurant in explained in this column
17. Listed\_in(city) : City in which restaurant is located