

ARCHITECTURE

***Restaurant Rating PREDICTION***

***January 23, 2023***

***Written by Sofana Benoutiq & Othmane Zoubairi***

CONTENTS

**DOCUMENT 1**

**ABSTRACT**

1. **INTRODUCTION**

**1.1 WHY THIS LOW-LEVEL DESIGN DOCUMENT?**

**1.2 SCOPE** 4 **2) TECHNICAL SPECIFICATIONS** 5 **2.1 DATASET OVERVIEW** 3

**2.2 PREDICTING Travel Package Purchase PROBABILITY** 3

**2.3 LOGGING** 4

**3) TECHNOLOGY STACK** 4

**4) PROPOSED SOLUTION** 4

**5) WORKFLOW** 5

1 | P a g e

**DOCUMENT VERSION CONTROL**

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Versions | Description | Author |
| 22/12/2022 | 1.0.1 | First Draft | Othmane Zoubairi  &  Sofana Benoutiq |
| 15/01/2023 | 1.0.2 | Added Technical  specifications | Othmane Zoubairi  &  Sofana Benoutiq |
| 23/01/2023 | 1.0.3 | Added Technology  stack, Proposed  solution and  Workflow | Othmane Zoubairi  &  Sofana Benoutiq |
|  |  |  |  |

2 | P a g e

**ABSTRACT**

**The basic idea of analyzing the dataset is to get a fair idea about the factors affecting the establishment of different types of the restaurant at different places and Cities, aggregate rating of each restaurant, With each day new restaurants opening the industry hasn't been saturated yet and the demand is increasing day by day. Inspite of increasing demand it, however, has become difficult for new restaurants to compete with established restaurants.**

**Most of the people here are dependent mainly on the restaurant food as they don’t have time to cook for themselves. So that it is important to predict witch Restaurant has good Rating or not.**

**1) Introduction**

**1.1 Why this Low-Level Design Document?**

**The goal of LLD or a Low-level design document is to give an  internal logical design of the actual program code for the  Concrete Compressive Strength Prediction System. LLD  describes the class diagrams with the methods and relations  between classes and program specs. It describes the modules  so that the programmer can directly code the program from the  document.**

**1.2 Scope**

**Low-level design (LLD) is a component level design process  that follows a step-by-step refinement process. This process  can be used for designing data structures, required software  architecture, source code and ultimately, performance  algorithms. Overall, the data organization may be defined during  requirement analysis and then defined during data design work.**

3 | P a g e

**2) Technical Specifications**

**2.1 Dataset Overview**

**For training and testing the model, I used the public set  available in Ineuron internship :**

URL : **https://raw.githubusercontent.com/mrinalmayank7/datascience/main/zom.csv**

**Dataset Attributes:**

1. Restaurant Id: Unique id of every restaurant across various cities of the world
2. Restaurant Name: Name of the restaurant
3. Country Code: Country in which restaurant is located
4. City: City in which restaurant is located
5. Address: Address of the restaurant
6. Locality: Location in the city
7. Locality Verbose: Detailed description of the locality
8. Longitude: Longitude coordinate of the restaurant’s location
9. Latitude: Latitude coordinate of the restaurant’s location
10. Cuisines: Cuisines offered by the restaurant
11. Average Cost for two: Cost for two people in different currencies (local currency)
12. Currency: Currency of the country
13. Has Table booking: yes/no
14. Has Online delivery: yes/ no
15. Is delivering: yes/ no
16. Switch to order menu: yes/no
17. Price range: range of price of food
18. Aggregate Rating: Average rating out of 5
19. Rating color: depending upon the average rating color
20. Rating text: text on the basis of rating of rating
21. Votes: Number of ratings given

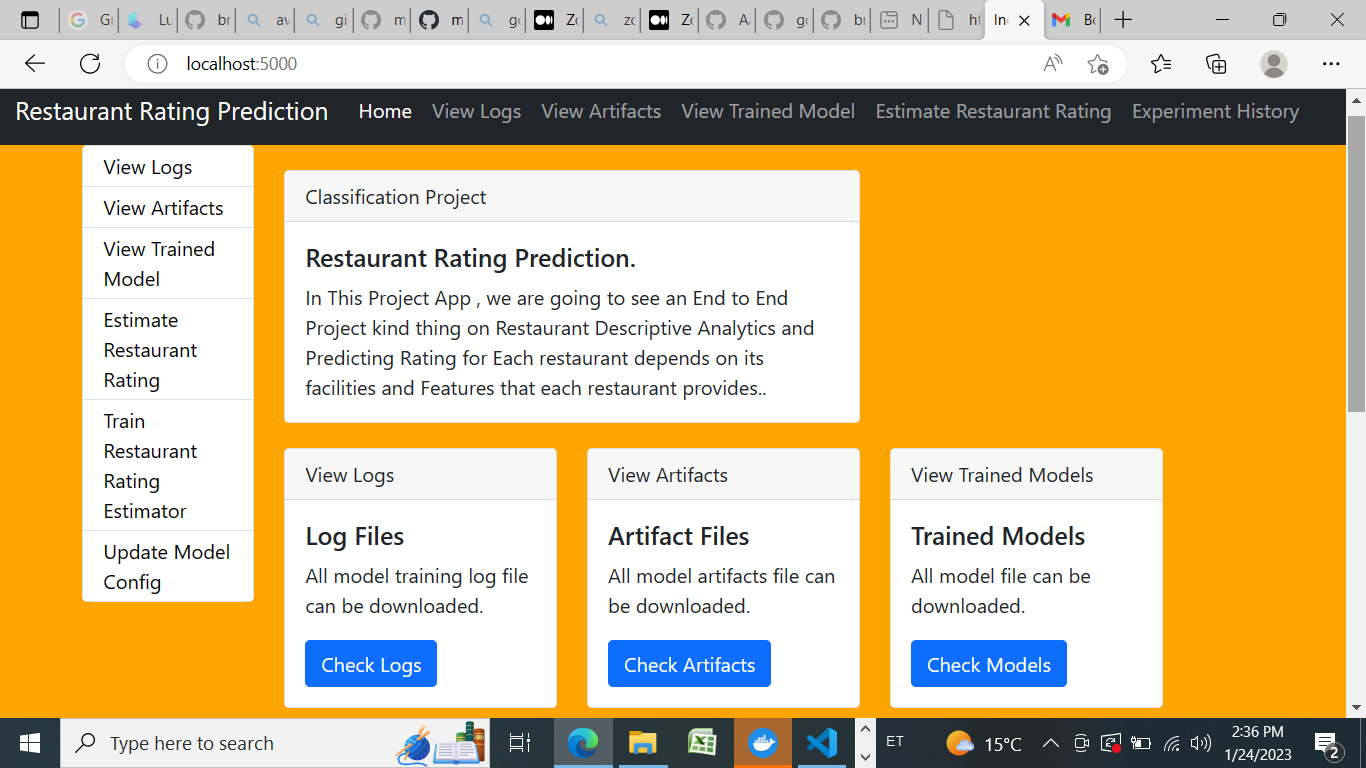
**2.2 Predicting Restaurant Rating**

**The web application must be loaded properly for the users  without any technical glitches like server timeouts.**

**• It must display the input fields and the “Predict” button to  the users who accessed the application and allow the user  to enter the values with respect to the attributes of the  customer.**

**• The user gives the required information.**

**• Then the application should be able to predict the  probability of Package Purchase based on the information given by the  user about the customer**.



4 | P a g e

**2.3 Logging**

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

**• The system should be able to log every step in the program**

**flow and in every Time .**

**• System should not be hung even after using so**

**many loggings.**

**• Logging makes debugging much easier, like we can  directly**

**go to that specific line of code, having bugs.**

**• In this project, logs will be written in the files**

**• “restaurant\_rating\_logs.” and the “deployment\_logs.” respectively.**

****

**Technology stack**

|  |  |
| --- | --- |
| Front-end | HTML with CSS styling |
| Back-end | Python version 3.7,  Flask version 2.0.1 |
| Deployment | Heroku, gunicorn version  20.1.0 |
|  |  |

**3)-Proposed solution**

**The solution proposed here is a web application, which takes  the details of the customer and those details will be taken by a  machine learning model in the backend, which will then predict the probability of Restaurant Rating and display it on the front-end page of  the user.**

**4)- Workflow**

**- START DATA -🡪 INGESTION DATA**

**- Data validation-🡪MODEL Transformation**

**- Model Trainer 🡪-Model EVALUATION**

**-BEST MODEL🡪- MODEL Pusher**

**-DEPLOYMENT🡪- APPLICATION START**

**-PREDICT RESTAURANT RATING - END**