Low Level Design (LLD)

FLIGHT FARE

PREDICTION

ABHISHEK MONARCH

EMMANUAL

**Contents**

**Abstract**

**INTRODUCTION**

**Why this LLD documentation?**

**1 Architecture 3**

**2 Architecture Design**

**2.1 Data Gathering from Main Source**

**2.2 Tools Used 6 2.3 Data Description**

**2.4 Import Data into Database**

**2.5 Export Data from Database**

**2.6 Data Pre-Processing**

**2.7 Modelling**

**2.8 UI Integration**

**2.9 Data From User**

**2.10 Data Validation**

**2.11 Rendering the Results**

**3 Deployment**

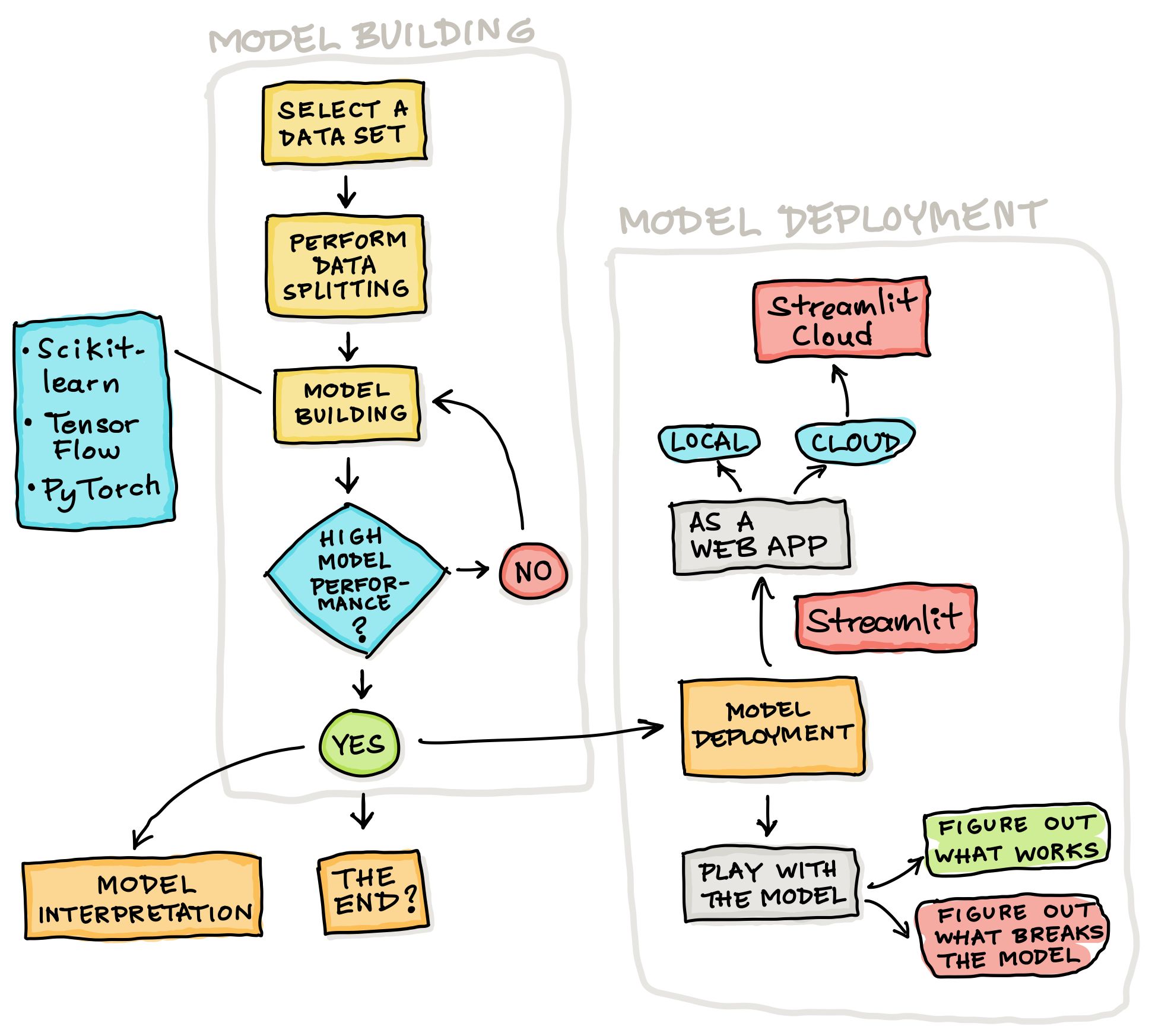
**Abstract**

The recent international things had a large impact on the aviation sector because of several reasons. This impact has 2 class folks, the primary is business perspective and therefore the second is that the customers perspective. As safety is that the major reason for such impact on the aviation sector, the governments round the world amended totally different rules to their various airlines firms. These restrictions had created the supply of the flights and their attendant capability less. Taking of these factors in thought the value of the flight tickets has accrued and vary from one place to the opposite. Booking a flight price tag has split into 2, one is that the on-line and therefore the alternative is that the offline bookings. each these have their various criteria for value of the price tag, one such example is that the server load and therefore the range of booking requests. during this machine learning implementation, we are going to see numerous factors that impact worth of the flight ticket price and predict the acceptable price of the ticket

**1 Introduction**

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

The main purpose of this LLD documentation is to feature the required details of the project and supply the outline of the machine learning model and also the written code. This additionally provides the careful description on however the complete project has been designed end-to-end.



**2. Architecture Design**

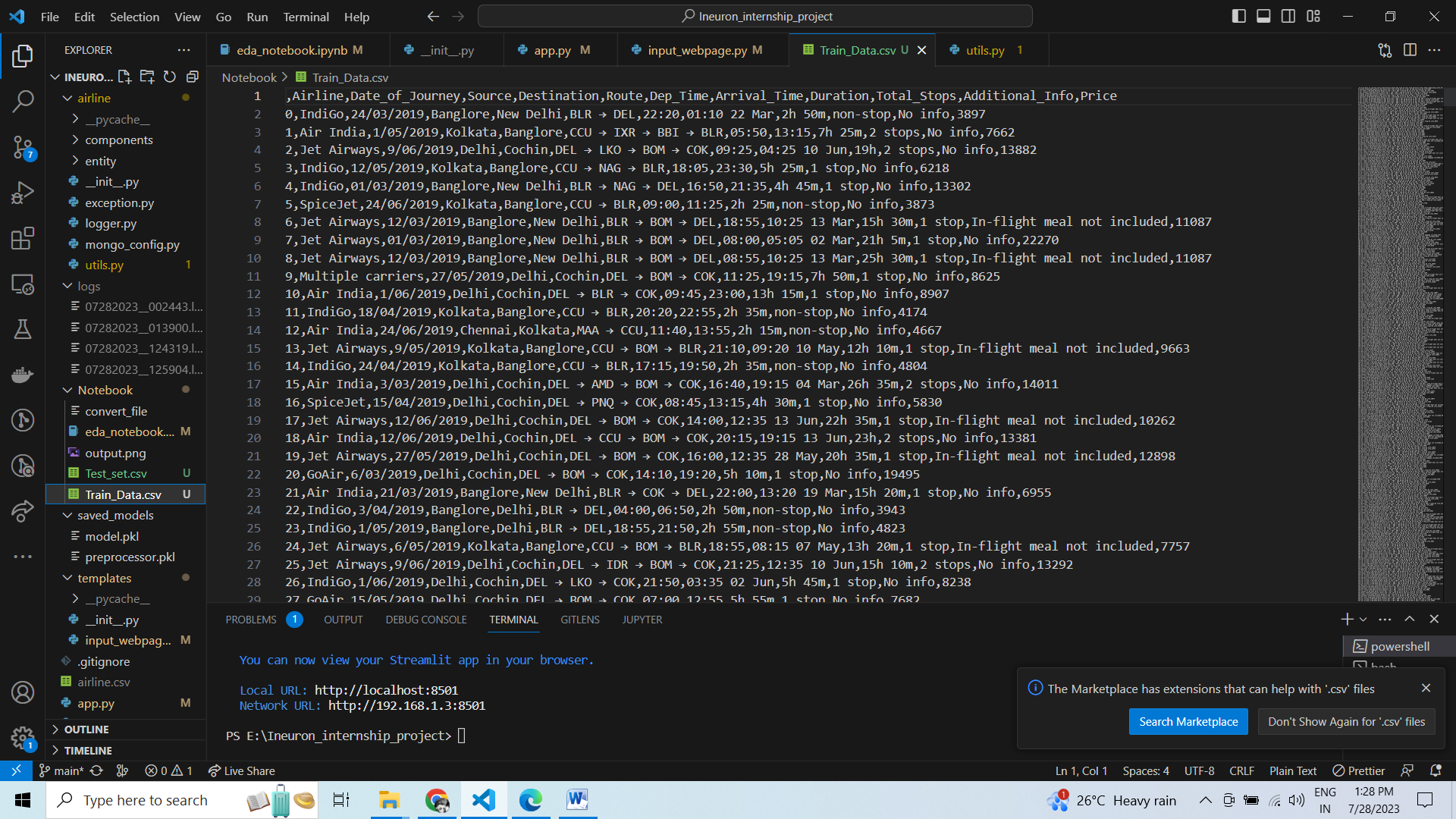
This project is to make associate interface for the user to grasp their approximate flight price ticket worth, additionally to the present, in would like of obtaining the important time project expertise we have a tendency to square measure mercantilism the gathered information into our own information then begin the project from the scratch.

**2.1. Data Gathering**

The data for the current project is being gathered from Kaggle dataset, the link to the data is: <https://www.kaggle.com/nikhilmittal/flight-fare-prediction-mh>

2.2. Tool Used • Python 3.9 is employed because the programming language and frame works like numpy, pandas, sklearn and alternative modules for building the model. • PyCharm is employed as IDE. • For visualizations seaborn and components of matplotlib are getting used • For information assortment prophetess info is getting used version management. • Streamlit is employed for deployment

**2.3 Data Description**

There are about 10k+ records of flight information such as airlines, data of journey, source, destination, departure time, arrival time, duration, total stops, additional information, and price. A glance of the dataset is shown below.

**2.4**

**Import 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 flightfare.

• Cqlsh command is written for making the info table with needed parameters.

• And finally, a cqlsh command is written for uploading the knowledgeset into data table by bulk insertion.

**2.5 Export Data into Database**

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

**2.6 Data Preprocessing**

Steps performed in pre-processing are:

• First the info sorts square measure being checked and located solely the value column is of sort number.

• Checked for null values as there square measure few null values, those rows square measure born. • Converted all the desired column into the date time format.

• Performed one-hot cryptography for the desired columns.

• Scaling is performed for needed information.

• And, the info is prepared for passing to the machine learning formula

**2.7 Modelling**

The pre-processed information is then envisioned and every one the specified insights ar being drawn. though from the drawn insights, the info is at random unfold however still modelling is performed with completely different machine learning algorithms to form positive we tend to cowl all the chances. and eventually, for sure random forest regression performed well and any hyperparameter calibration is finished to extend the model’s accuracy.

**2.8 UI Integration**

Streamlit Template 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 2.3 Data from User The data from the user is retrieved from the created HTML web page. 2.4 Data Validation The data provided by the user is then being processed by app.py file and validated. The validated data is then sent for the prediction. 2.11 Rendering Result The data sent for the prediction is then rendered to the web page.

**3. Deployment**

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

