High-Level Design (HLD)

Flight Fare Prediction

Revision Number – 1.2

Last Date of Revision: 24 – 04 -2023

ANKIT

MUDIT ARORA

M.S.S.S.CHAITANYA

Document Version Control

Version	Date	Author	Description
0.1	19 – APRIL - 2023	M.S.S.S.CHAITANYA	Abstract Introduction Problem Statement
0.2	20 – APRIL - 2023	ANKIT	Design Flow
0.3	21 – APRIL - 2023	MUDIT ARORA	Performance Evaluation Conclusion

Contents

Abstract

The Aviation sector faced significant challenges due to the recent changes in the international market. These changes affected both the Business and Customer perspectives of the industry. One of the main reasons for this was the different regulations imposed by governments around the world on their Airline companies. As a result, the flight ticket prices varied widely across different locations. There were also two modes of booking a flight ticket: online and offline. Each mode had its own factors that influenced the price, such as the server load and the number of booking requests. In this machine learning implementation, we will explore various factors that affect the flight ticket price and predict the optimal price for the ticket.

1

1. Introduction

1.1 Why this HLD Document ?

The main purpose of this HLD document is to feature the required details of the project and supply the outline of the Model Creation, Evaluation and Deployment. This additionally provides a careful description on how the complete project has been designed end-to-end.

The HLD will :

- Present of the design aspects and define them in detail.
- Describe the user interface being implemented.
- Describe the hardware and software interfaces.
- Describe the performance requirements.
- Include design features and architectural design of the project.
- List and describe the non - functional attributes like :
  1. Security
  2. Reliability
  3. Maintainability
  4. Portability
  5. Reusability
  6. Resource
  7. Utilization

1.2 Scope

The HLD documentation presents the structure of the system, such as database design, architectural design, application flow and technology architecture. The HLD uses non-technical terms to technical terms that can be understandable to the administrator of the system.

1.3 Definitions

Term	Description
FFP	Flight Fare Prediction
Database	Collection of all the information used by the System.
Jupyter - Notebook	It is an interactive computational environment, in which you can combine code execution, rich text, mathematics, plots and rich media.
RAILWAY	RAILWAY is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud.

2. General Description

2.1 Problem Perspective

The flight fare prediction is a machine learning model that helps users to predict the price of the flight tickets and help them to understand the price of their journey.

2.2 Problem Statement

After amendment of the new rules, there are changes in the flight fare price from one location to another. The main goal of the system is to create a model to predict the price of their flight fare on the basis of bound input provided by users like date of journey, Source, Destination and many more.

2.3 Proposed Solution

To solve the problem, we have created a User interface for taking the input from the user to predict the flight fare price using our trained ML model after processing the input and at last the output (predicted value) from the model is communicated to the User.

2.4 Further Improvements

We also analyzed the data used for training the ML model by considering different occasions such as Weekday, Season or any Social reasons, considering different angles of business. If we use such information and predict the discounted flight fare price, it will bring some loss to the airline companies but users can benefit from that. If we develop these using the Business perspective of Airline, this technique isn't thought - about.

2.5 Technical Requirements

As technical requirements, we don't need any specialized hardware for virtualization of the application. The user should have a device that has the access to the web and the fundamental understanding of providing the input. And for the backend, we need a server to run all the required packages to process the input and predict the desired output (predicted flight fare price).

2.6 Data Requirements

The Data requirements totally supported the matter statement and also the dataset is accessible on the Kaggle within the file format of (.xlsx). Because the main theme of the project is to induce the expertise of real time issues, we have a tendency to transform the information into the prophetess database and commerce it into csv format.

2.7 Tools Used

- Python 3.9 is employed because of the programming language and frameworks like NumPy, Pandas, Scikit - learn and alternative modules for building the model.

- Jupyter - Notebook is employed as an IDE.

- For Data visualizations, seaborn and components of matplotlib are getting used.

- For information assortment prophetess info is getting used.

- Front end development is completed victimization HTML/CSS.

- Flask is employed for each information and backend readying.

- GitHub is employed for version management.

- RAILWAY is employed for deployment.

2.8 Constraints

The flight fare prediction answer should be user friendly, as automatic as attainable and also the user should not be needed to understand any of the operations.

2.9 Assumptions

The main objective of the project is to implement the utili cases for the new dataset that provides the user the ability to predict Flight Fare Price. Machine learning model is employed for process the user input for prediction. It additionally assumed that each one aspects of this project have the flexibility to figure along within the approach the designer is expecting.

3. Design Flow

3.1 Modelling Process

```
┌─────────────────────┐     ┌─────────────────────┐     ┌─────────────────────┐
│     Importing       │     │       Deep          │     │    Exploratory      │
│     Required        │ ──▶ │    Processing       │ ──▶ │   Data Analysis     │
│ Libraries and Data  │     │                     │     │                     │
└─────────────────────┘     └─────────────────────┘     └─────────────────────┘
                                                                   │
                                                                   ▼
┌─────────────────────┐     ┌─────────────────────┐     ┌─────────────────────┐
│  Finding the best   │     │      Feature        │     │     Handling        │
│       model         │ ◀── │     Selection       │ ◀── │  Categorical Data   │
│                     │     │                     │     │                     │
└─────────────────────┘     └─────────────────────┘     └─────────────────────┘
          │
          ▼
┌─────────────────────┐     ┌─────────────────────┐     ┌─────────────────────┐
│    Fitting the      │     │   Hyperparametr     │     │      Model          │
│       model         │ ──▶ │     Tuning          │ ──▶ │    Evaluation       │
│                     │     │                     │     │                     │
└─────────────────────┘     └─────────────────────┘     └─────────────────────┘
```
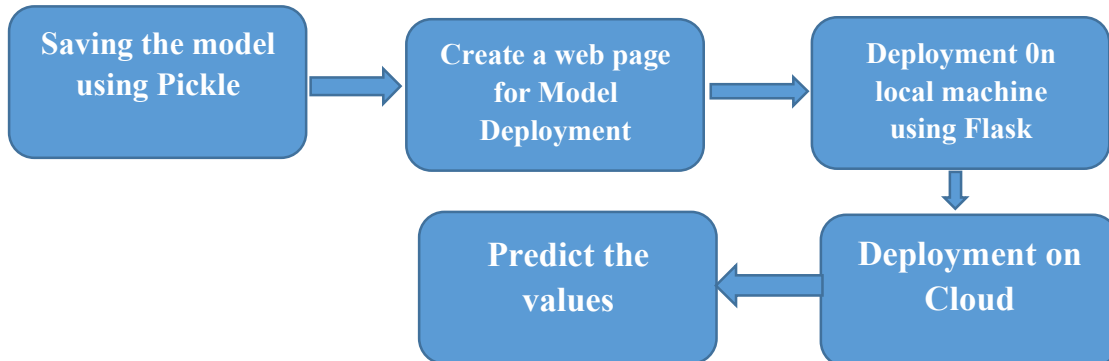
3.2 Deployment Process

```
┌─────────────────────┐     ┌─────────────────────┐     ┌─────────────────────┐
│  Saving the model   │     │   Create a web page │     │   Deployment 0n     │
│    using Pickle     │ ──▶ │   for Model         │ ──▶ │   local machine     │
│                     │     │   Deployment        │     │   using Flask       │
└─────────────────────┘     └─────────────────────┘     └─────────────────────┘
                                                                   │
                                                                   ▼
                            ┌─────────────────────┐     ┌─────────────────────┐
                            │    Predict the      │     │   Deployment on     │
                            │      values         │ ◀── │      Cloud          │
                            │                     │     │                     │
                            └─────────────────────┘     └─────────────────────┘
```
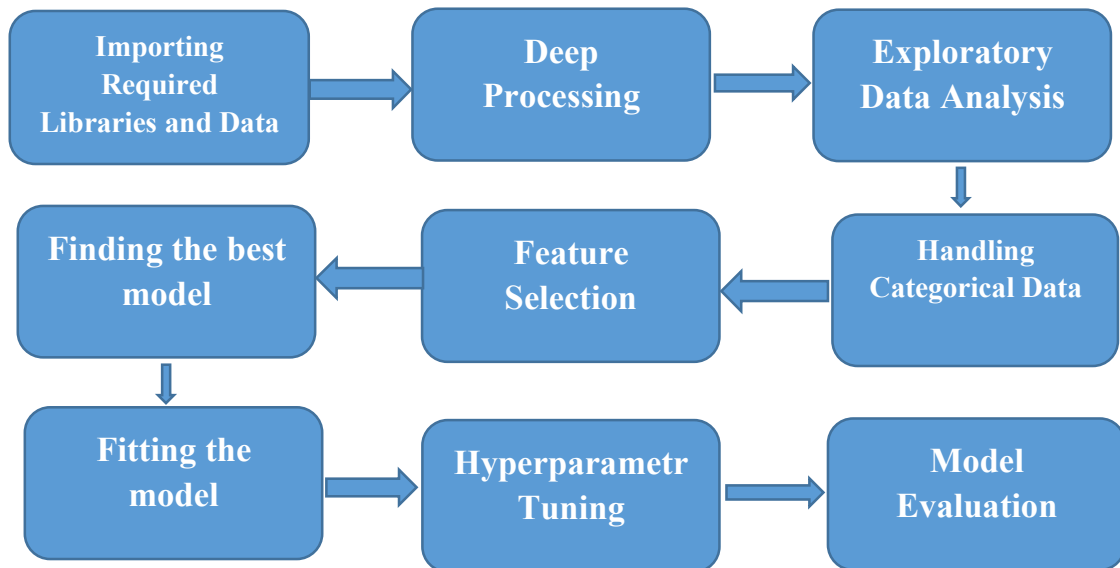
3.3 Logging

In logging, each time an error or an exception occurs, the event is logged into the system log file with reason and timestamp. This helps the developer to debug the system bugs and rectify the error.

3.4 Error Handling

Once the error occurs, the reason is logged into the log file with timestamp to rectify and handle it.

4. Performance Evaluation
4.1 Reusability

The code written and the components used should have the ability to be reused with no problems.

4.2 Application Compatibility

The different parts of the system are communicating or using Python as an interface between them. All the components have its own tasks to perform and it is the job of a Python to ensure proper transfer of data.

4.3 Resource Utilization

When a task is performed, it'll doubtless use all the process power offered till the process is finished.

4.4 Deployment

The model can be deployed using any cloud services such as Microsoft Azure, Amazon web services, Railway, Google cloud, etc.

5. Conclusion

The Flight Fare Prediction system will predict the price for helping the customers with the trained knowledge with set of rules. The user can use these system to recognize the approximate value of its flight fare for his or her journey.