Architecture Document

Flight Fare Prediction

An airplane flying in the sky

Description automatically generated

Project Member: Rohit Atnoor

Mentor: Ineuron

Latest Version: 1.7

Last Data of Revision: 19/09/2023

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 15/07/2023 | 1.0 | Initial Project setup, data extraction setup. | Rohit Atnoor |
| 25/07/2023 | 1.1 | Setting up the data ingestion pipeline | Rohit Atnoor |
| 30/07/2023 | 1.2 | Creating of Data Transformation pipeline. | Rohit Atnoor |
| 03/08/2023 | 1.3 | Creation of prediction pipeline. | Rohit Atnoor |
| 08/08/2023 | 1.4 | Initial End to End prediction pipeline. | Rohit Atnoor |
| 12/08/2023 | 1.5 | Deploying of HTML Files. | Rohit Atnoor |
| 20/08/2023 | 1.6 | Data Addition | Rohit Atnoor |
| 10/09/2023 | 1.7 | Final End to End Deployable code. | Rohit Atnoor |

**Abstract**

In today's world, air travel has become an essential part of life for many people. However, the prices of flight tickets can fluctuate greatly depending on a variety of factors, such as the time of day, day of the week, time of year, and destination. This can make it difficult for travelers to find the best deals on flights.

This project aims to develop a machine learning model to predict flight prices based on different factors available in a provided dataset. The model will be trained on historical flight data, and it will be used to generate predictions for future flight prices.

The predicted flight prices can be used by travelers to find the best time to book their flights and to save money on their travel expenses. The model can also be used by airlines to set their prices and to maximize their profits.

This project has the potential to make a significant impact on the travel industry by helping travelers to save money and by helping airlines to optimize their pricing strategies

**Contents**

1 Introduction

1.1 Why this Low-Level Design Document?

1.2 Scope

1.3 Constraints

1.4 Risks

1.5 Out of Scope

2 Technical specifications

2.1 Dataset

2.2 Logging

2.3 Database

3 Technical Stack

4 User I/O workflow

5 Key Performance Indicators

**1 Introduction**

* 1. **Why this Low Level Design Architecture Document**

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

* 1. **Scope**

The Scope of the Project is to predict the flight price based on the user inputs.

the prices of flight tickets can fluctuate greatly depending on a variety of factors, such as the time of day, day of the week, time of year, and destination. This can make it difficult for travelers to find the best deals on flights

**1.3 Constraints**

We will only be selecting a few of the Flight Details and only few source and destination.

* 1. **Risk**

Document specific risks that have been identified or that should be considered.

* 1. **Out of Scope**

Delineate specific activities, capabilities, and items that are out of scope for the project

1. **Technical Specification**

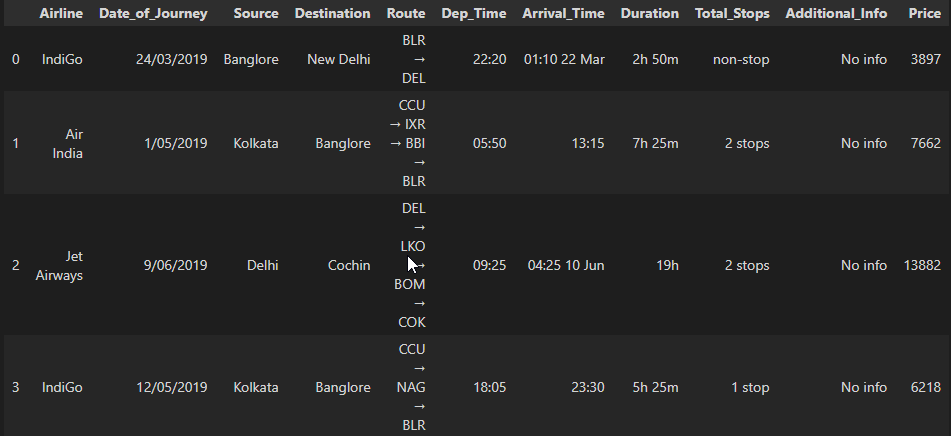
**2.1 Dataset**

1. Totaly there are 8 companys in this dataset.

2. Multiple carriers means a Connecting flight between source and destination.

3. Trujet this flight ceased all operations in February 2022, so we will delete all the records from this flight.

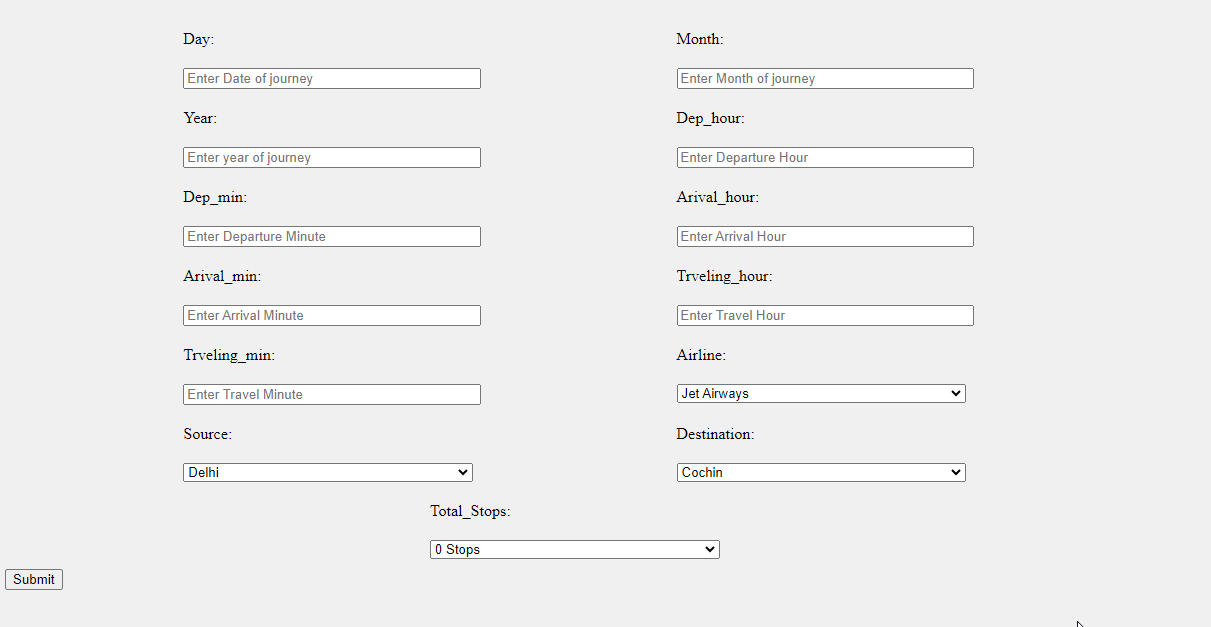
4. In India now there are 13 active Airlnes company.



Raw Data



Cleaned Data



Input Dataset Schema

**2.2 Logging**

The system is able to log every event so that the user will know what process is running internally.

1. The System identifies at what step logging required

2. The System is able to log each and every system flow and its failure.

3. In this we have used File logging method to store the log details.

4. System can handle so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

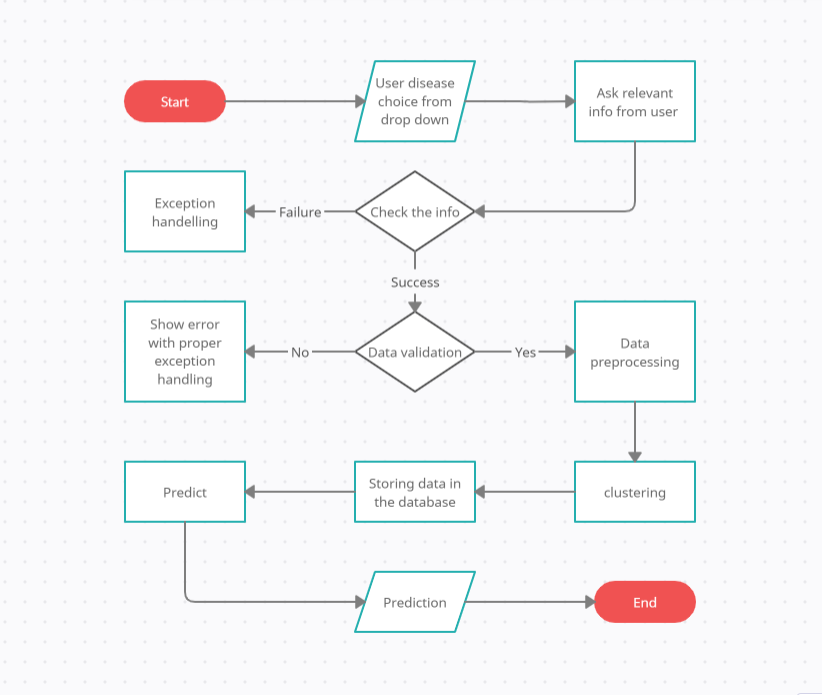
**2.3 Database**

System needs to store every request into the database and we need to store it in such a way that it is easy to retrain the model as well so we have used SQL database as this is a rational data set

1. **Technical Stack**

|  |  |
| --- | --- |
| **Front End** | HTML |
| **Backend** | Python Flask |
| **Database** | MySQL |
| **Deployment** | Azure Web Services, AWS |

1. **User I/O workflow**

****

Price Prediction UI

**3.7 KPIs (Key Performance Indicators)**

**Accuracy**: This KPI measures how close the predicted flight fare is to the actual flight fare. Accuracy can be measured using a variety of metrics, such as mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

**Precision**: This KPI measures how close the predicted flight fare is to the actual flight fare for flights that are actually booked. Precision is important because it tells us how well the model is predicting the fares of flights that are actually relevant to travelers.

**Recall**: This KPI measures how many of the flights that are actually booked are predicted to have a high fare. Recall is important because it tells us how well the model is identifying flights that are likely to be expensive.

**Timeliness**: This KPI measures how quickly the model can generate predictions. Timeliness is important because travelers often need to know the price of a flight quickly in order to make a booking decision.

**Cost**: This KPI measures the cost of developing and maintaining the flight fare prediction model. Cost is important because it needs to be balanced against the benefits of using the model, such as the savings that travelers can achieve.