1. Choose a project from an internship portal and try to write a HLD and LLD based on the sample given in your portal for a respective project .

**Low Level Design (LLD)**

**FLIGHT FARE PREDICTION**

Document Version: 0.3

Document Version Control:

|  |  |
| --- | --- |
| Version | Description |
| 0.1 | Introduction Architecture, Document Content, Version Control and Unit Test cases to be  added |
| 0.2 | Data Pre-processing,  Model Building |
| 0.3 | Deployment |

**Contents**

1. [Introduction 3](#_TOC_250008)
   1. [What is Low-Level design document ? 3](#_TOC_250007)
   2. [Scope 3](#_TOC_250006)
2. [Architecture 3](#_TOC_250005)
3. [Architecture Description 4](#_TOC_250004)
   1. [Data Gathering 4](#_TOC_250003)
   2. [Data Description 4](#_TOC_250002)
   3. [Tool Used 5](#_TOC_250001)
   4. [Data Pre-processing 5](#_TOC_250000)
   5. Model Building 5
   6. Data from User 6
   7. Data Validation 6
   8. Rendering Result 6
   9. Deployment 6
4. Unit Test Cases 6

# Introduction

## What is 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.

* 1. **Scope**

Low-level design (LLD) is a component-level design process that follows a step-bystep [refinement](https://en.wikipedia.org/wiki/Refinement_(computing)) 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 refined during data design work

# Architecture

Import python libraries and read data

Data Processing

EDA

Handling categorical data

Saving Data in Pickle File

Hyper Parameter Tuning

Model Building

Feature Selection

Create a webpage Using Streamlit for Deployment

Deployment on Heroku Cloud

Input Value and Predict Final Result

End

# Architecture Description

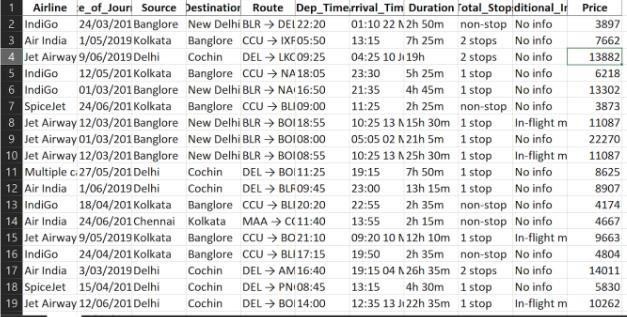
This project is to make associate interface for the user to grasp their approximate flight price ticket worth, additionally to the present, in would like of obtaining the important time project expertise we have a tendency to square measure mercantilism the gathered information into our own information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

## 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>

## 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.



## 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.
* Jupyter Notebook is used for EDA and for model building.

## Data Pre-processing

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

## Model Building

The pre-processed information is then envisioned and every one the specified insights are 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.

## Data from User

The data from the user is retrieved from the created HTML web page.

## 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.

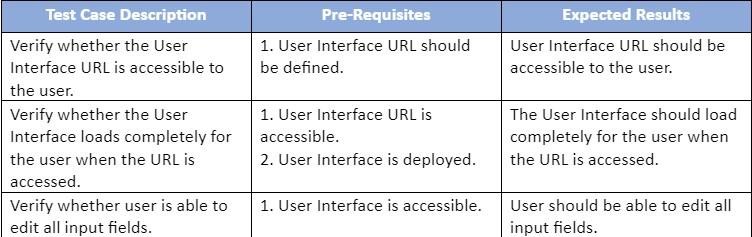
## Rendering Result

The data sent for the prediction is then rendered to the web page.

## Deployment

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

## Unit Tests



**High Level Design (HLD)**

**FLIGHT FARE PREDICTION**

Document Version: 0.3

Document Version Control:

|  |  |
| --- | --- |
| Version | Description |
| 0.1 | Introduction  Problem Statement |
| 0.2 | Design Flow |
| 0.3 | Performance Evaluation |

**Contents**

1. [Introduction 3](#_TOC_250016)
   1. [What is High-Level design document ? 3](#_TOC_250015)
   2. [Scope 3](#_TOC_250014)
2. [Description 3](#_TOC_250013)
   1. [Problem Perspective 3](#_TOC_250012)
   2. [Problem Statement 3](#_TOC_250011)
   3. [Purposed Solution 3](#_TOC_250010)
   4. [Solution Improvements 4](#_TOC_250009)
   5. [Technical Requirements 4](#_TOC_250008)
   6. [Data Requirements 4](#_TOC_250007)
   7. Constraints 4
   8. Assumptions 5
3. [Design Flow 5](#_TOC_250006)
   1. Modelling Process 5
   2. Deployment Process 5
   3. Logging 5
   4. Error Handling 5
4. [Performance Evaluation 6](#_TOC_250005)
   1. [Reusability 6](#_TOC_250004)
   2. [Application Compatibility 6](#_TOC_250003)
   3. [Resource Utilization 6](#_TOC_250002)
   4. [Deployment 6](#_TOC_250001)
5. [Conclusion 6](#_TOC_250000)

# Introduction

## What is High-Level design document?

The main purpose of this HLD 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.

## Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

# Description

## Problem Perspective

The flight fare prediction may be a machine learning model that helps America to predict the price of the flight price tag and helps the users to understand the price of their journey.

## Problem Statement

The most goal of the project is to form a programme that predicts the price of the flight price tag by taking bound input from the user like date of journey, aboard location and destination etc.

## Purposed Solution

Projected to require the desired input of user from the created interface and method all the provided information to satisfy the wants of the machine learning model and at last show the output oral communication so and then quantity is that the expected value.

## Solution Improvements

we will even predict the price of price tag considering whether or not is it a weekday, season or alternative social reasons. however, considering from the angle of business, if we have a tendency to method such information and predict the price of the discounted price tag it'll bring some loss to the airlines company. therefore, this technique isn't thought-about.

## Technical Requirements

There are not any hardware needs needed for victimization this application, the user should have AN interactive device that has access to the web and should have the fundamental understanding of providing the input. And for the backend half the server should run all the package that's needed for the process the provided information and to show the results.

## Data Requirements

The info demand is totally supported the matter statement. and also, the information set is accessible on the Kaggle within the type of standout sheet(.xlsx). because the main theme of the project is to induce the expertise of real time issues, we have a tendency to ar once more mercantilism {the information into the prophetess data base and commerce it into csv format.

* 1. **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.
* Front end development is completed by Streamlit.
* GitHub is employed for version management.
* Cloud platform is employed for deployment.
  1. **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 operating.

* 1. **Assumptions**

The most objective of the project is to implement the utilization cases as for the new dataset that user provides through the programme. Machine learning model is employed for process the on top of computer file. it's additionally assumed that each one aspects of this project have the flexibility to figure along within the approach the designer is expecting.

# Design Flow

* 1. **Modelling and Deployment Process**

Import python libraries and read data

Data Processing

EDA

Handling categorical data

Saving Data in Pickle File

Hyper Parameter Tuning

Model Building

Feature Selection

Create a webpage Using Streamlit for Deployment

Deployment on Heroku Cloud

Input Value and Predict Final Result

End

* 1. **Logging**

Each step is being logged within the system that runs internally, that shows the date time and therefore the processed that has been performed, work is completed in several layers as information, DEBUG, ERROR, WARNINGS. this provides US the perceive of the logged info.

* 1. **Error Handling**

Once ever a slip is occurred, the reason are logged in its several log file, in order that the developer will rectify the error.

# Performance Evaluation

## Reusability

Elements of the code written is accustomed different applications and therefore the rest is changed and be reused.

## Application Compatibility

The various parts for this project are exploitation python as associate interface between them. every element can have its own tasks to perform, and it's the work of the python to make sure correct transfer of data.

## Resource Utilization

Once any task is performed, it'll doubtless use all the process power offered till that perform is finished.

## Deployment

The model is being deployed on Cloud platform.

# Conclusion

The flight fare prediction will predict the worth supported the trained knowledge set within the rule. therefore, the user will recognize the approximate value for his or her journey.