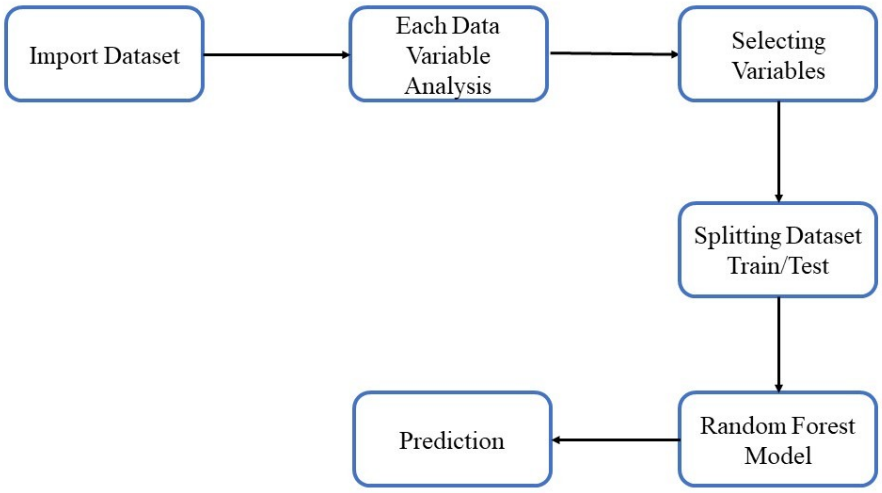
1. Introduction

1.1. What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the
actual program code for flight fare estimation 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-bystep
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 refined during data design work.

2. Architecture

3. Architecture Description

3.1. Data Collection

We have 14k Dataset row columnar data includes the flight service,flight fare,number of
stops,total number of duration,departure-arrival date and all.These is given in the comma

seprated value format (.csv).These data is collected from the Kaggle which contains both the test
data and train data.

3.2. Data Cleaning

In the Cleaning process, We have cleaned up all the data because data is present in very bad
format which was cannot reconigzed by machine.So data engineering is done very first.

3.3. Data Pre-processing

Data Pre-processing steps we could use are Null value handling, One hot encoding, columns in a
integer format by implementing proper techniques to manage the columnar data.

3.4. Exploratory Data Analysis

In eda we have seen various insights from the data so we have selected which column is most
important and dropped some of the columns by observing their sperman rank co-relation and
plotting their heatmap from seaborn library also we done outlier removal and null value managed in
a efficient manner and also implemented one hot encoding their.

Model Creation:

After cleaning the data and completing the feature engineering. we have done splitted data in the train
data and test data and implemented various regression algorithm like Linear, DecisionTree, SVR,
RandomForest, K-NN, AdaBoost, GradientBoost Regression and also calculated their accuracies on test
data and train data.

Model Dump:

After comparing all accuracies and checked all roc, auc curve we have choosen hyperparameterized
random forest regression as our best model by their results so we have dumped these model in a pickle
file format with the help of joblib python module.

User Interface :

In Frontend creation we have made a user interactive page where user can enter their input values to
our application. In these frontend page we have made a form which has beautiful styling with css and
bootstrap. These html user input data is transferred in json format to backend. Made these html fully
in a decoupled format.

3.10. Data from User

Here we will collect users requirement to catch their flight like a number of stops, departure date,
destination , Source, Number of hours , Day etc.

3.11. Model Call/.pkl file loaded

Based on the User input will be throwing to the backend in the dictionary format so
our we are loading our pickle file in the backend and predicting some total number
of hours as a output and sending to our html page.

3.12. Deployment