**Low Level Document (LLD)**

Flight Fare Prediction

**Dipali Bhanushali, Muaaz Patel, Soham Pawar, khushabu Patil.**

Pursuing Masters in Data Science & Business Analytics.

# Contents

[Abstract 1](#_TOC_250013)

1. [Introduction 2](#_TOC_250012)
   1. [What is Low-Level design document? 2](#_TOC_250011)
   2. [Scope 2](#_TOC_250010)
2. [Architecture 3](#_TOC_250009)
3. [Architecture Description 3](#_TOC_250008)
   1. [Data Collection 3](#_TOC_250007)
   2. [Data Pre-processing 4](#_TOC_250006)
   3. Feature Extraction and Selection 5
   4. [Train and take a look at knowledge Sets 6](#_TOC_250005)
   5. [SUMMARY OF VARIOUS PREDICTION 7](#_TOC_250004)
   6. [Data from User 7](#_TOC_250003)
   7. [Data preparing 7](#_TOC_250002)
   8. [Model called for the data 7](#_TOC_250001)
   9. [Predicted data 7](#_TOC_250000)

5. Unit Test Cases 8

# Abstract

According to a report, India’s civil aviation industry is on a high-growth trajectory. India aims become the third-largest aviation market by 2020 and the largest by 2030. Indian domestic air traffic is expected to cross 100 million passengers by FY2017, compared to 81 million passengers in 2015, as per Centre for Asia Pacific Aviation (CAPA).According to Google Trends, the search term - "Cheap Air Tickets" is most search in India. Moreover, as the middle-class of India is exposed to air travel, consumers hunting for cheap prices increases.

# Introduction

Flight ticket prices can be something hard to guess, today we might see a price, check out the price of the same flight tomorrow, and it will be a different story. To solve this problem, we have been provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities, using which we aim to build a model which predicts the prices of the flights using various input features

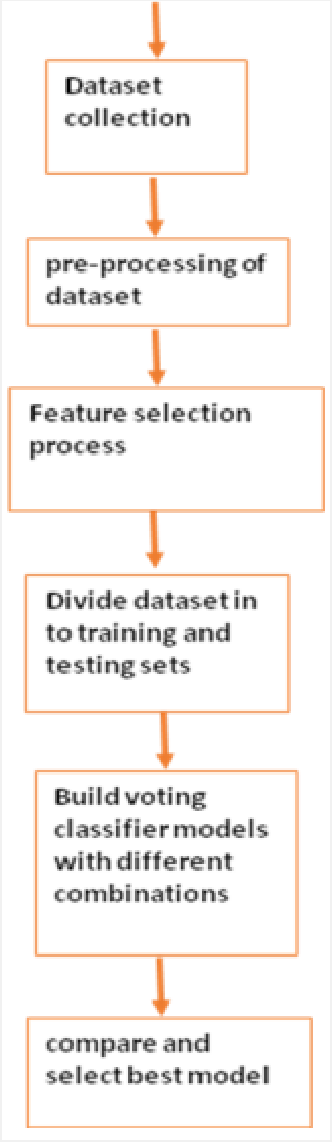
## What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Food Recommendation 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.

## Scope

Low-level design (LLD) is a component-level design process that follows a step-by step 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

# Architecture



# . Architecture Description

## Data Collection

The dataset for this article can be downloaded from this Kaggle link. Unzip the downloaded zip ﬁle and place the “advertising.csv” ﬁle in your local drive. This is the ﬁle that we are going to use to train our machine learning model

## Data Pre-processing

Date\_of\_Journey is an object data type, Therefore, we have to convert this data type into timestamp so as to use this column properly for prediction For this we require pandas to\_datetime to convert object data type to datetime dtype

## Feature Extraction and Selection

The data scientist's data has several features that may or may not be relevant to the topic of interest. Also, it may not be in a suitable format. The ﬁrst and foremost task to the data scientist is to extract the appropriate collection of attributes that preferably suits the learning algorithm. Before processing, it needs to be transformed to prevent relapse problems like overﬁtting and underﬁtting as presented.

## Train and take a look at knowledge Sets

Once the dataset is processed, we want to divide it into 2 components that's coaching and take a look at set. We'll Take and use the train\_test\_split operate for that and every one variable except ‘Clicked on Ad’ are the input values x for the cubic centimetre models. The variable ‘Clicked on Ad’ are keep in y, can represent the prediction variable and that we at random selected to portion thirty third of the whole knowledge for the coachin

## Data from User

Here given data will be undergone all the pre-processing techniques (3.3) which we done on the early available dataset.

## Model called for the data

The saved model will be called for the prediction on the given data.

## Predicted data

On the given data the loaded model will perform prediction.

## 5. Unit Test Cases

|  |  |  |
| --- | --- | --- |
| **Test Case Description** | **Pre-Requisite** | **Expected Result** |
| Verify whether the  Application URL is  accessible to the user | 1. Application URL  should be deﬁned | Application URL should be  accessible to the user |
| Verify whether the Application loads completely for the user when the URL  is accessed | 1. Application URL is accessible 2. Application is deployed | The Application should load completely for the user when the  URL is accessed |
| Verify whether user is able to see input  ﬁelds on logging in | 1. Application is accessible 2. User is able to see input ﬁelds. | User should be able to see input  ﬁelds on logging in |
| Verify whether user is able to edit all  input ﬁelds | 1. Application is accessible 2. User is able to see input ﬁelds. 3. User is able to edit input ﬁelds. | User should be able to edit all input  ﬁelds |
| Verify whether user gets Submit  button to submit the inputs | 1. Application is accessible 2. User is able to see input ﬁelds. 3. User is able to edit input ﬁelds. 4. User is able to see submit button. | User should get Submit button to  submit the inputs |
| Verify whether user is  presented with prediction results on clicking  submit | 1. Application is   accessible   1. User is able to see input ﬁelds. 2. User is able to edit input ﬁelds. 3. User is able to see submit button. | User should be presented  with  Predicted with results on clicking submit |