

Architecture Design

Flight Fare Prediction - System



Content:

Document Control.....	1
1. Introduction	
1.1 What is Architecture Design Document.....	2
1.2 What is Scope.....	2
1.3 Risks.....	2
1.4 Out of Scope.....	2
2. Proposed Solution.....	3
3. Technical Specification.....	3
3.1 Dataset.....	3
3.2 Logging.....	3
3.3 Database.....	3
3.4 Technology Stack.....	3
4. Model Training.....	4
5. User I/O workflow.....	5

Document Control

Date	Version	Description	Author
07/01/2022	1.0	Introduction, Architecture, Deployment	Abhishek Saha
14/01/2022	1.1	Final Revision	Abhishek Saha

1 Introduction

1.1 What is Architecture Design Document?

Any software needs the architectural design to represent the design of the software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectures. Each style will describe a system category that consists of:

- A set of components (Eg: A database, computational modules) that will perform a function required by the system.
- The set of connectors will help in coordination, communication, and cooperation between the components.
- Conditions that how components can be integrated to form the system.
- Semantic models help the designer to understand the overall properties of the system.

1.2 What is Scope?

Architecture Design Document (ADD) is an architectural design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the design principles may be defined during requirement analysis and then refined during architectural design work.

1.3 Risks:

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

1.4 Out of Scope:

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

2. Proposed Solution:

The Solution proposed here is a Flight Fare Prediction based system that can be implemented to perform above mentioned use case. In this case we have dataset which contain several information like Date-of-journey, Source, Destination, Route, Number-of-stoppage, Travel duration, Additional information, Fare depends on this various factor we trying to build a Machine Learning Model. After training the Model we are saving it for future prediction, so that depend on predict value Passenger make their tour plan and also save money and time.

3. Technical Specification:

3.1 Dataset:

Dataset	Finalize	Source
Flight Fare Prediction	Yes	https://github.com/abhisheksaharaja/Flight_Price_Prediction_ML_Project/tree/main/Training_Batch_Files

3.2 Logging:

We should be able to log every activity done by entire project workflow.

- The system should be able to maintain a log on each and every system flow.
- System should not be hung even after so many loggings. Logging just because we can easily debug issue so logging is mandatory to do.

3.3 Database:

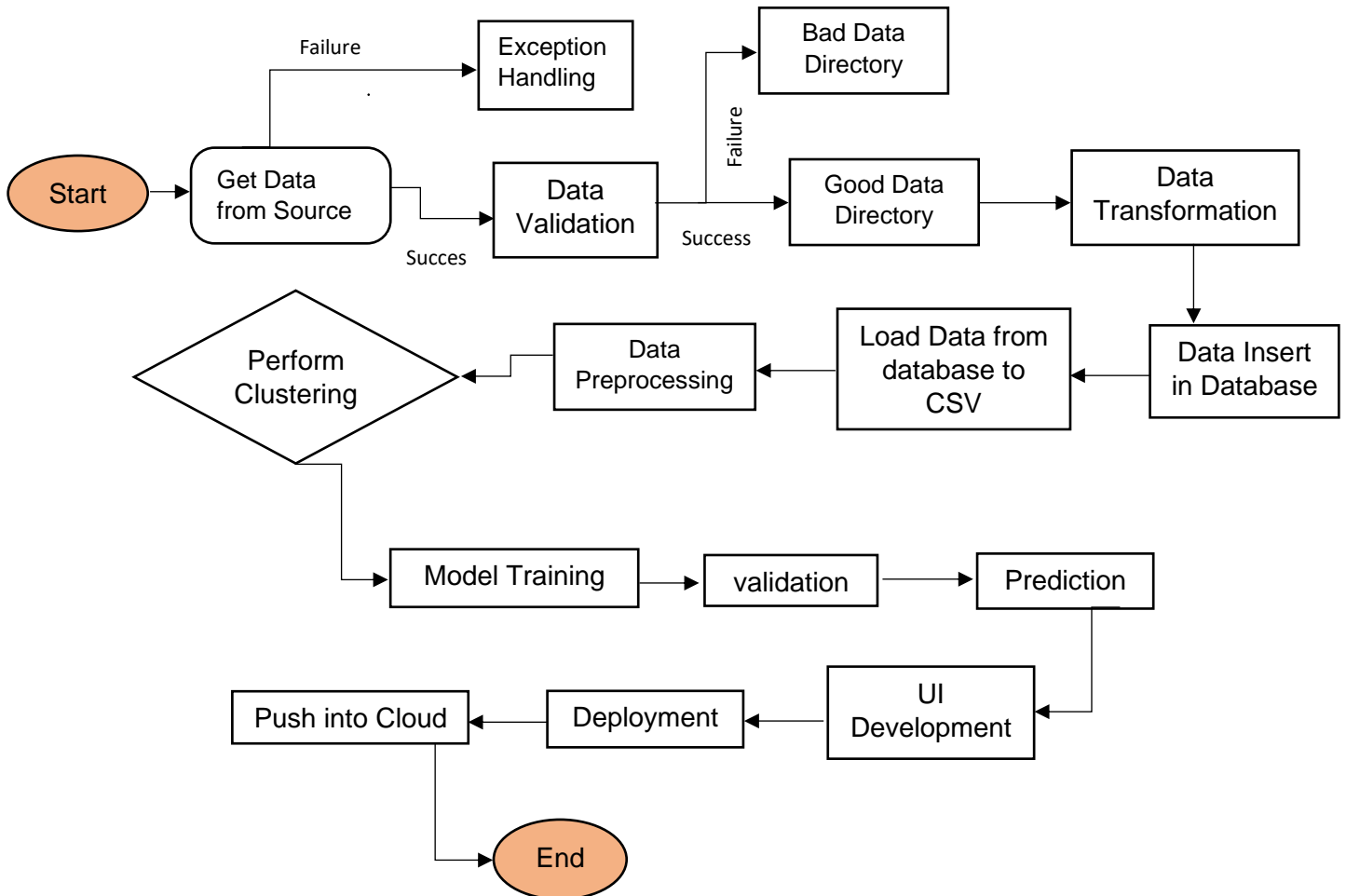
Systems need to store every request into the database and we need to store it in such a way that is easily to retrain the Model.

The system store each and every data given by the user.

3.4 Technology Stack:

Front End	HTML, CSS, JS
Web Framework	Flask
Database	SQLite
Deployment	Heroku

4. Model Training/ Validation workflow:



5. User I/O workflow:

