**1 LOW LEVEL DESIGN** 

Low Level Design

**House Price Prediction**

| **Written By** | Kiran Kumari |
| --- | --- |
| **Document Version** | 0.3 |
| **Last Revised Date** |  |

**Airport Data Analysis**

**2 LOW LEVEL DESIGN** 

**DOCUMENT CONTROL**

**Change Record:**

| **VERSION** | **DATE** | **AUTHOR** | **COMMENTS** |
| --- | --- | --- | --- |
| 0.1 | 26th-jan-2023 | Kiran Kumari | Introduction and architecture defined |
| 0.2 | 26th-jan-2023 | Kiran Kumari | Architecture & Architecture description appended and updated. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Airport Data Analysis**

**3 LOW LEVEL DESIGN**

**1. Introduction**

**1.1 What is Low-Level design document?**

The goal of the LDD or Low-level design document (LLDD) is to give the internal logic design of the actual program code for the House Price Prediction dashboard. LDD describes the class diagrams with the methods and relations between classes and programs 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-by-step refinement process. The 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.

**Airport Data Analysis**

**5 LOW LEVEL DESIGN** 

**2. Architecture**

****

**Tableau Server Architecture**

Tableau has a highly scalable, n-tier client-server architecture that serves mobile clients, web clients and desktop-installed software. Tableau Server architecture supports fast and flexible deployments.

The following diagram shows Tableau Server’s architecture:

**6 LOW LEVEL DESIGN** 

Tableau Server is internally managed by the multiple server processes.

**1. Gateway/Load Balancer**

It acts as an Entry gate to the Tableu Server and also balances the load to the Server if multiple Processes are configured.

**2) Application Server:-**

Application Server processes (wgserver.exe) handle browsing and permissions for the Tableau Server web and mobile interfaces. When a user opens a view in a client device, that user starts a session on Tableau Server. This means that an Application Server thread starts and checks the permissions for that user and that view.

**3) Repository:-**

Tableau Server Repository is a PostgreSQL database that stores server data. This data includes information about Tableau Server users, groups and group assignments, permissions, projects, data sources, and extract metadata and refresh information.

**4) VIZQL Server:-**

**Airport Data Analysis**

**7 LOW LEVEL DESIGN** 

Once a view is opened, the client sends a request to the VizQL process (vizqlserver.exe). The VizQL process then sends queries directly to the data source, returning a result set that is rendered as images and presented to the user. Each VizQL Server has its own cache that can be shared across multiple users

**5) Data Engine:-**

It Stores data extracts and answers queries.

**6) Backgrounder:-**

The backgrounder Executes server tasks which includes refreshes scheduled extracts, tasks initiated from tabcmd and manages other background tasks.

**7) Data Server:-**

Data Server Manages connections to Tableau Server data sources

It also maintains metadata from Tableau Desktop, such as calculations, definitions, and groups.

**3. Architecture Description**

**3.1. Data Description**

The Dataset contains house price of cities that fall under the categories A,B and C based on the availability of parking, rainfall, its built-up area etc

1. Dist\_Taxi: Distance to nearest taxi stand from the property (in metres).

2. Dist\_Market: Distance to nearest grocery market from the property (in metres). 3. Dist\_Hospital: Distance to nearest hospital from the property (in metres). 4. Carpet: Carpet area of the property in square feet (in square ft.)

5. Built-up: Built-up area of the property in square feet (in square ft.)

6. Parking: Type of car parking available with the property

7. City\_Category: Categorization of the city based on the size

**HOUSE PRICE PREDICTION** 7

**8 LOW LEVEL DESIGN** 

8. Rainfall: Annual rainfall in the area where property is located (in cm)

9. House\_Price: Price at which the property was sold (in Dollars)

**3.2. Web Scrapping**

Web scraping is a technique to automatically extract content and data from websites using bots. It is also known as web data extraction or web harvesting. Web scrapping is made simple now days, many tools are used for web scrapping. Some of python libraries used for web scrapping are Beautiful Soup, Scrapy, Selenium, etc.

**3.3. Data Transformation**

In the Transformation Process, we will convert our original datasets with other necessary attributes format. And will merge it with the Scrapped dataset.

**3.4. Data Insertion into Database**

a. Database Creation and connection - Create a database with name passed. If the database is already created, open the connection to the database.

b. Table creation in the database.

c. Insertion of files in the table

**Airport Data Analysis**

**9 LOW LEVEL DESIGN** 

**Airport Data Analysis**

**10 LOW LEVEL DESIGN** 

If you want to run a specific SQL command every-time a new connection is established, you can use the Initial SQL option. This will open a dialogue box, where you can specify your desired SQL query.

**Step 2: Configuring Data Source**

The data source page loads up after configuring the Tableau connector and successfully signing in. This is how the page looks like:

Select the data source name option and give a unique name to the database you are using. It’s considered a good practice to have a unique name as it makes it much easier for users to identify the database from which data is being fetched.

To select the desired schema, you can use the schema drop-down list from the column on the left. You can also perform a text-based search to find the desired option. Now similarly find and select the desired table and drag it onto the canvas.

**11 LOW LEVEL DESIGN**

**3.5. Export Data from Database**

Data Export from Database - The data in a stored database is exported as a CSV file to be used for Data Pre-processing.

**3.6 Deployment.**

Once you’ve completed your dashboard, follow these steps:**- Server, Tableau Public, Save to Tableau Public As**

You may be prompted to log into your Tableau Public profile first if this is your first time publishing.

**Airport Data Analysis**

**12 LOW LEVEL DESIGN**

**4. Unit Test Cases**

| **TEST CASE DESCRIPTION** | **EXPECTED RESULTS** |
| --- | --- |
| Rainfall parameter slicer | When clicked on the slicer, a dropdown should occur which has various parameters of the rainfall. |
| House Price Parameter | When clicked on the slicer, a dropdown should occur which describes the parameters of the House Prices. |
| Relation Between Rainfall and Average Housing Price | Here a time series graph is shown of Rainfall VS Average House Price data. |
| Rainfall and Average House Price across the cities | Various city category is shown and a visualization is created which shows the City Category and Avg. House Price and relation. |
| Relation between Rainfall and Built-up Parameters across the Cities | The visual should show a bubble diagram of relation between various built-up parameters across various cities. |
| Min, Max & Avg. Housing Price Comparison by categories | This is an important visual in bar-graph which shows the category of Max Housing Price, Mini Housing price and Avg. housing price across Built-up parameters and City categories. |

**Airport Data Analysis**