iNeuron

Airbnb Data Analysis

Architecture Document

Divyansh Verma
Data Analyst Intern
iNeuron AI

Table of Contents

1. Introduction

1.1 What is an Architecture Design Document?

An Architecture Design Document provides a comprehensive overview of the architecture of a software system. It details the system's components, their interactions, and deployment strategies to ensure efficient and effective data analysis and visualization.

1.2 Scope

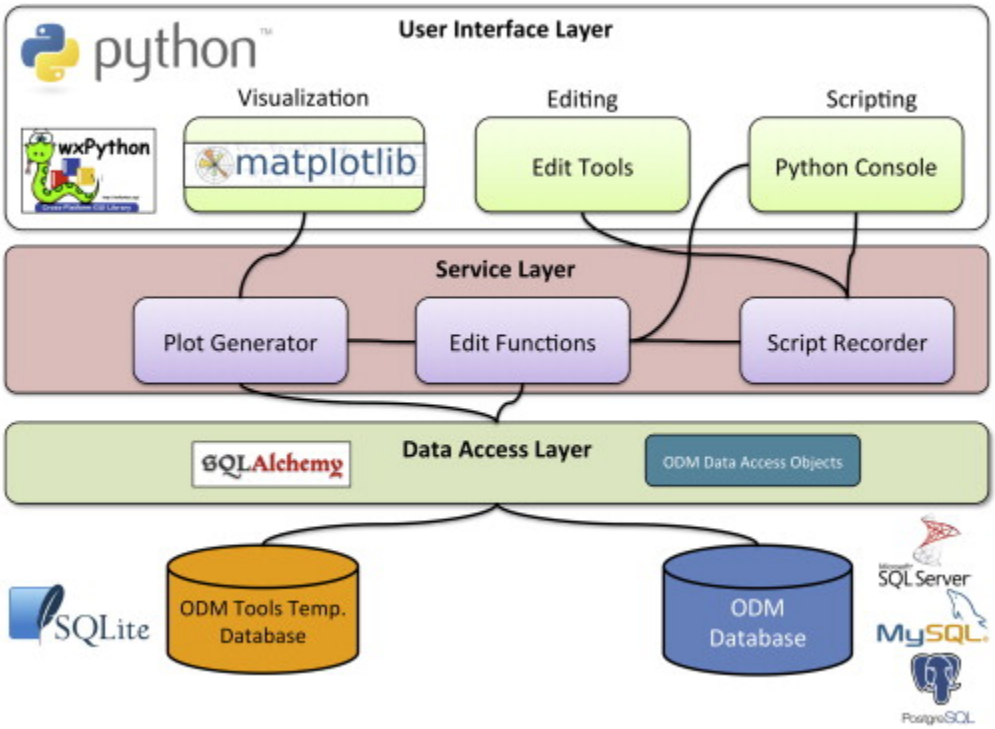
This document covers the architecture for analyzing Airbnb data using Python and visualizing the results with Power BI. The analysis includes host earnings, neighborhood bookings, review quality, and price-related insights.

2. Architecture

2.1 Python Architecture for Data Analysis

The Python architecture for this data analysis includes the following components:

- **Data Ingestion**: Data is imported from a CSV file hosted on GitHub.
- **Data Preprocessing**: Cleaning and preparing data by handling missing values, creating new columns, and summarizing statistics.
- **Data Analysis**: Performing specific analyses such as correlation, group by operations, and regression analysis.
- **Visualization**: Using libraries like Matplotlib and Seaborn to create visualizations that help in interpreting the data.

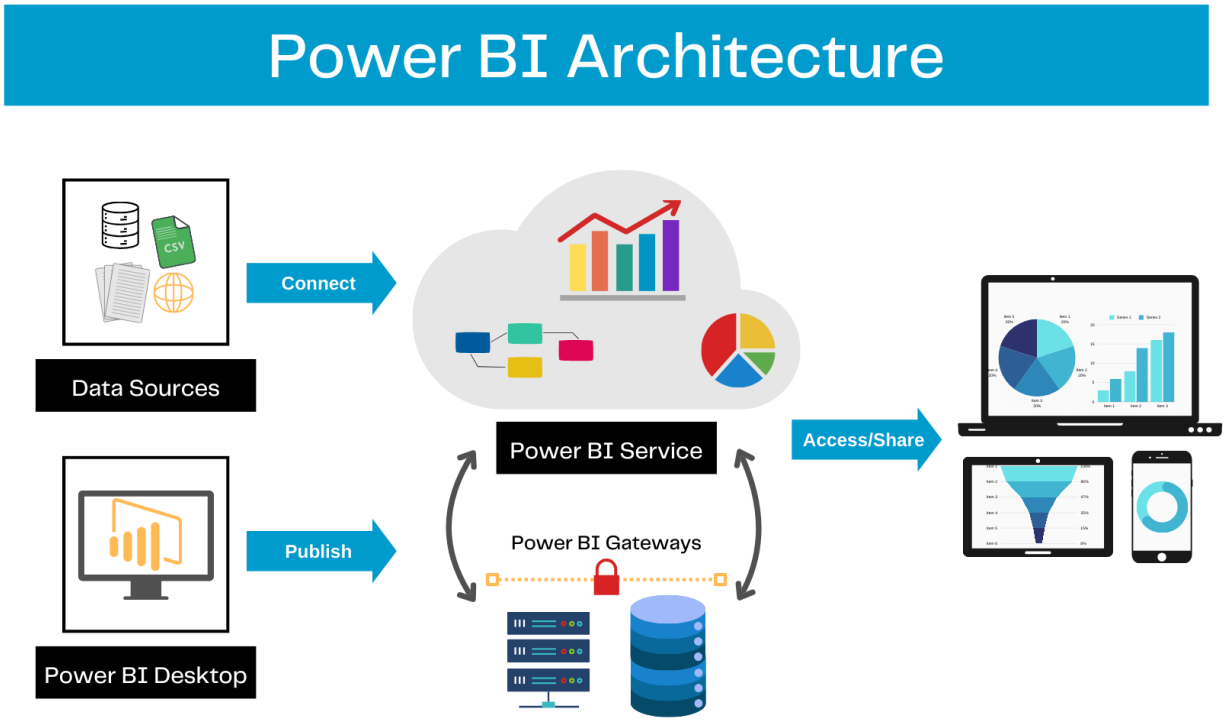


2.2 Power BI Architecture

Power BI is used for advanced visualization and reporting. The architecture includes:

- **Data Import**: Importing the preprocessed data from Python.
- **Data Transformation**: Further refining data using Power Query if needed.

- **Visualization**: Creating interactive maps and charts to display price and booking information geographically.



2.3 Data Ingestion and Preprocessing

In Python, data ingestion and preprocessing steps include:

- Reading data from CSV using `pandas`.
- Handling missing values by filling or dropping them.
- Creating new columns for analysis, such as `total_customers` and `total_earnings`.
- Generating summary statistics and unique value counts for initial exploration.

2.4 Data Storage and Management

Data is stored in memory during the analysis process. For persistent storage, the cleaned and processed data can be saved back to CSV or a database. Key tools include:

- **Pandas DataFrames** for in-memory data manipulation.
- **CSV files** for data storage and sharing.

2.5 Data Analysis and Processing

The analysis includes:

- Identifying top-earning hosts.
- Correlation analysis between prices and earnings.
- Grouping data by neighborhood and calculating booking counts.
- Examining the relationship between review quality and price.
- Statistical modeling using `statsmodels` for regression analysis.

2.6 Visualization and Reporting

Python visualizations use:

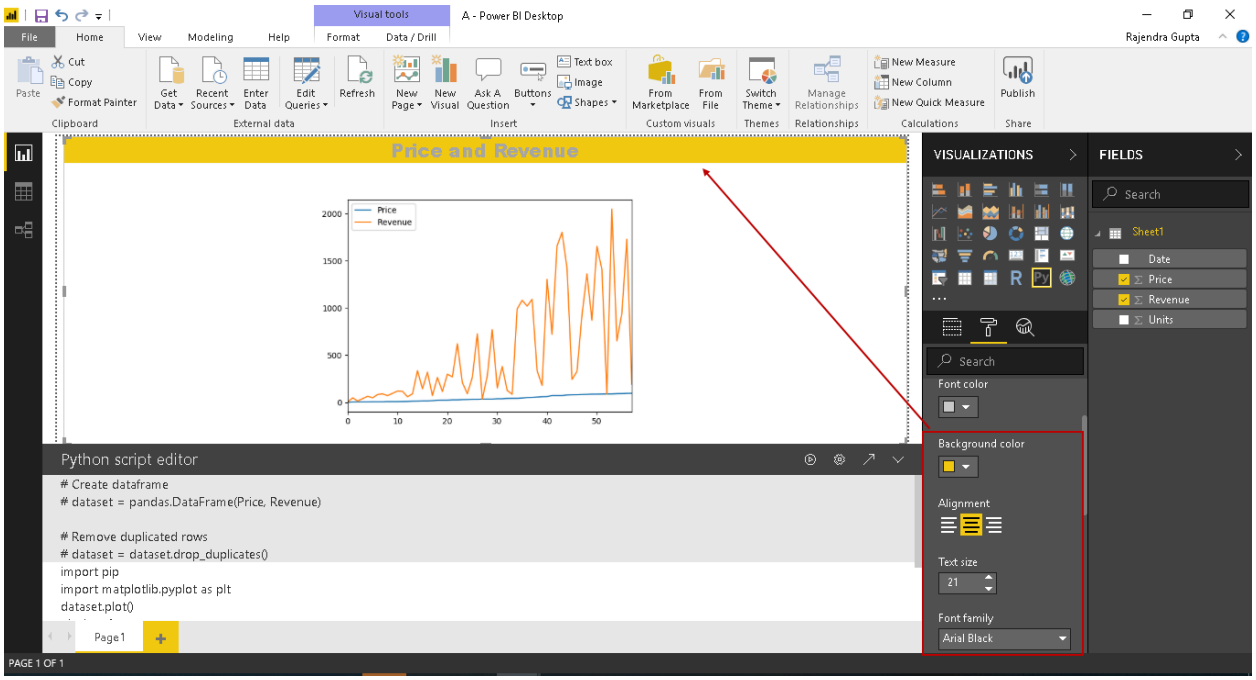
- **Matplotlib** for basic plotting.
- **Seaborn** for advanced statistical visualizations, such as boxplots and regression plots.

Power BI visualizations include:

- **Interactive maps** showing geographical distribution of prices and bookings.
- **Charts and dashboards** for summary statistics and insights.

2.7 Integration between Python and Power BI

Data processed and analyzed in Python is exported to a CSV file, which is then imported into Power BI for visualization. This integration ensures that the heavy lifting of data processing is handled by Python, while Power BI provides a user-friendly interface for interaction and presentation.



2.8 Security and Compliance

- **Data Security**: Ensure data is stored and processed securely, using encryption and access control mechanisms.
- **Compliance**: Adhere to data protection regulations like GDPR, ensuring that sensitive data is anonymized or handled appropriately.

3. Deployment

3.1 Deployment Options for Python and Power BI

- **Local Deployment**: Running Python scripts on local machines and using Power BI Desktop for visualization.
- **Cloud Deployment**: Utilizing cloud services like Azure or AWS for running Python scripts and Power BI Service for sharing reports.
- **Hybrid Deployment**: Combining local data processing with cloud-based visualization and reporting.

3.2 Local Deployment

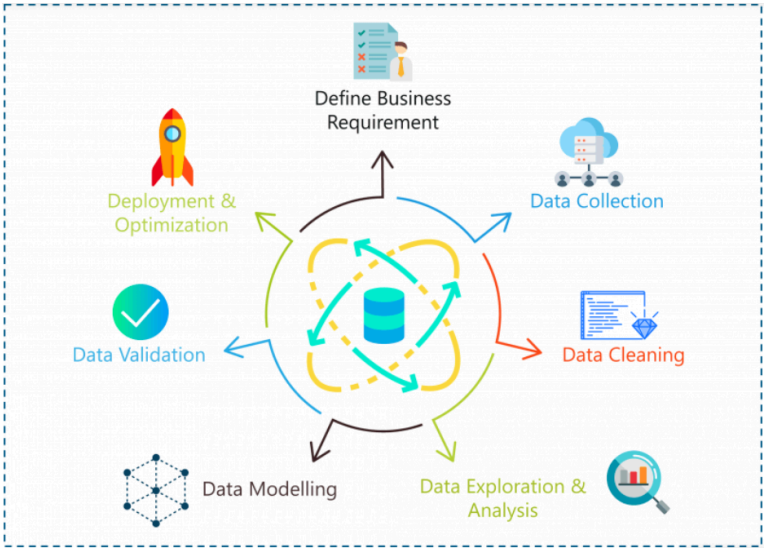
- Python scripts are executed on local machines with necessary libraries installed.
- Power BI Desktop is used for creating and viewing reports.

3.3 Cloud Deployment

- Python scripts are run on cloud platforms (e.g., AWS Lambda, Azure Functions).
- Power BI reports are published to Power BI Service for online access and sharing.

3.4 Hybrid Deployment

- Data analysis is performed locally or on a dedicated server.
- Processed data is uploaded to the cloud for visualization with Power BI Service.



Three separate environments;
DEV, Test, PROD (end-user)