Technical Design Document: Analyzing & Predicting NYC's Service Requests

Introduction:

This technical design document provides an overview of the technical infrastructure and tools used to conduct Exploratory Data Analysis (EDA) and predictive modeling on the NYC's 311-service data. The primary goal is to derive valuable insights from the dataset, aiding in improving urban service efficiency and citizen satisfaction.

Coding Language and Tools:

- **Pandas**: Used for data preprocessing, cleaning, and manipulation.
- **NumPy**: Facilitates numerical operations and computations.
- **Matplotlib** and **Seaborn**: Employed for data visualization and insights.
- **Scikit-Learn**: Utilized for predictive modeling and analysis.

Infrastructure and System:

For this project, Google Colab and Jupyter Notebook will be the preferred coding environments. Both platforms offer interactive and collaborative coding capabilities, enabling seamless analysis and visualization. Additionally, the project's code and outputs will be stored on GitHub for version control and easy sharing.

Cloud Services:

This project will primarily be executed on local machines, without extensive reliance on cloud computing resources. However, **Google Colab's cloud-based** environment provides computing resources, ensuring smooth execution, even for resource-intensive tasks.

Data Flow:

The data flow for this project involves the following steps:

1. **Data Loading**: The raw 311 service data will be loaded from the provided CSV file.
2. **Data Preprocessing**: Pandas will be used to clean and preprocess the dataset, addressing missing values and inconsistencies.
3. **Exploratory Data Analysis**: The preprocessed data will be subjected to EDA, involving statistical analysis and data visualization.
4. **Feature Engineering**: Relevant features will be selected or engineered to facilitate predictive modeling.
5. **Modeling**: Scikit-Learn's machine learning algorithms will be employed for predictive modeling, particularly Decision Tree classifiers.
6. **Model Evaluation**: Model performance will be assessed using appropriate evaluation metrics.

Data Size:

The size of the 311-service dataset is around a quarter gigabyte. While the dataset size isn't particularly large, the project's implementation and modeling techniques will be optimized to handle the data effectively.

Cost Estimate:

As this project primarily utilizes local computing resources and open-source tools, the associated costs are minimal. No significant cloud services or specialized computing environments are required, thus mitigating potential costs.

If one wants to dive in deeper into Machine Learning and AI, the cost of a standalone compute engine will depend on the type and number of virtual machines (VMs) used, as well as the duration of use. Assuming you need 100GB of compute engine with 16/32 RAM, you could use a custom VM with 16 or 32 vCPUs, 100GB of memory, and 1TB of SSD storage. The estimated cost for this VM would be approximately $800 - $1600 per month, depending on the selected region and the duration of use.

Additionally, there may be additional costs associated with using PySpark and Google Colab, such as increased memory and storage requirements. For instance, Google Colab Pro: $9.99/month for 16 GB RAM or $49.99/month for 32 GB RAM. However, these costs are likely to be minimal compared to the compute engine and S3 storage costs.

Conclusion:

This technical design document outlines the technical aspects of the Exploratory Data Analysis and Modeling project on the 311 service data. By leveraging Python's libraries, Jupyter Notebook, and Scikit-Learn, this project aims to provide actionable insights into urban service trends, improve resource allocation, and enhance citizen satisfaction. The seamless integration of data preprocessing, EDA, and modeling will pave the way for effective analysis, enabling data-driven decision-making.