Software Design Document for Standard Analytics on Datasets

Adrija Chakraborty
(2020201063)

B Sindhu
(2020201048)

Likhitha Gaddi
(2020201066)

CS4.409.S22.Data Foundation Systems

International Institute of Information and Technology,
Hyderabad

10-03-2022

Contents

1 Introduction

1.1 Purpose of This Design Document

This Design Document tracks and documents the necessary information required to effectively define architecture and system design in order to help develop the architecture of the system. Its intended audience are the developers and anyone willing to understand the product design.

2 System Overview

This project aims at performing various types of standard analytics on a set of available datasets. It also aims at performing end user analysis of these datasets. It aims to perform the following things:

- Display a list of available datasets
- Fetch top 50 rows of the user's chosen dataset and display.
- Perform standard analytics like mean, median, mode on the attributes of a selected dataset.
- Track Number of downloads for each dataset
- Monthly analyse the number of downloads and report the analysis
- Display graphs, plots, and comparisons of multiple datasets.

3 Design Considerations

3.1 Assumptions and dependencies

- Efficient uploading of databases are handled already. We will assume that datasets are already available in the database.
- Efficient downloading of huge datasets are handled already.
-

3.2 Design Constraints

3.2.1 Financial

The full implementation of the project could be financially significant. This is because of needing a huge storage area for large datasets and also having fast, distributed servers to efficiently fetch required datasets.

3.2.2 Technical

This project depends on other existing sister projects like Efficient download manager, efficient uploading of datasets, and efficient storage of large datasets.

3.3 Development Method

We will follow agile method of development:

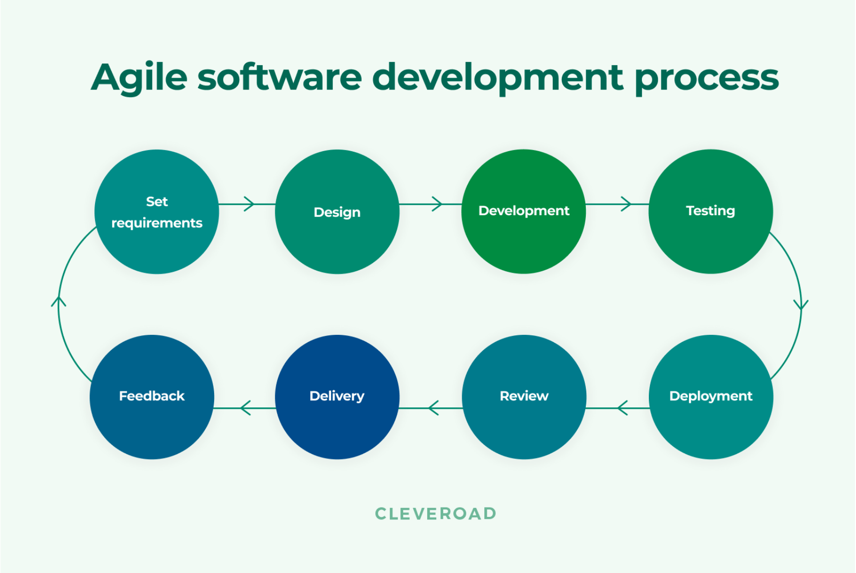


Figure 1: Agile Method of Software Development

4 Architecture Design

4.1 Logical View

Database structure:

- Dataset name
- Dataset(file)
- Download count
- Summary of dataset:
  - Number of columns
  - Number of missing values for each column

  - Datatype of each column etc.

1. Download dataset(API calls)- Key points:

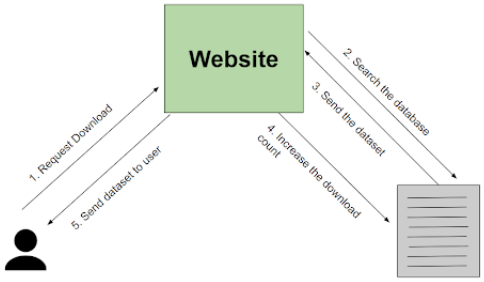


Figure 2: Download API call

   - As we have to maintain the number of downloads per dataset
   - We increase the count when a download request is received

2. Selects a dataset for analysis(API calls)- Key points:

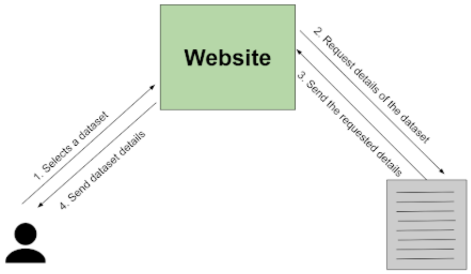

Website
1. Selects a dataset
4. Send dataset details
2. Request details of the dataset
3. Send the requested details

Figure 3: Select Dataset API call

   - When a user selects a dataset for analysis, an API call will be sent and summary of the dataset will be returned

3. Calculating mean,median etc., on columns- Key points:

   - When a query like calculating the mean of a column is requested
   - Select only that particular column from the database to decrease the load on the channel

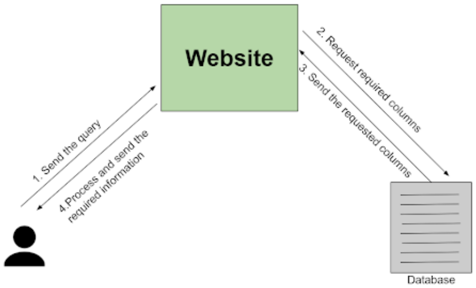


Figure 4: Analysis Procedure

4.2 Software Architecture

4.2.1 Class Diagram

Dataset class
This class contains the information about the dataset fetched. It contains following attributes:

- dataset_id: contains unique id of the dataset
- dataset_name: contains name of dataset
- download_count: contains how many times dataset has been downloaded

It contains the following methods:

- **SendDownloadCount(str): int**
  This function takes the id of dataset whose download count is to be fetched and sends the required information
- **SendColumn(str): str**
  This function takes the particular column name that the user wants to retrieve for analysis

StandardAnalytics class
This class helps in performing analysis internally on the dataset. It contains following attributes:

- dataset_id: contains unique id of the dataset

It contains the following methods:

- **GetDatasetID(str): void**
  This function takes the id of dataset on which the analysis is to be done

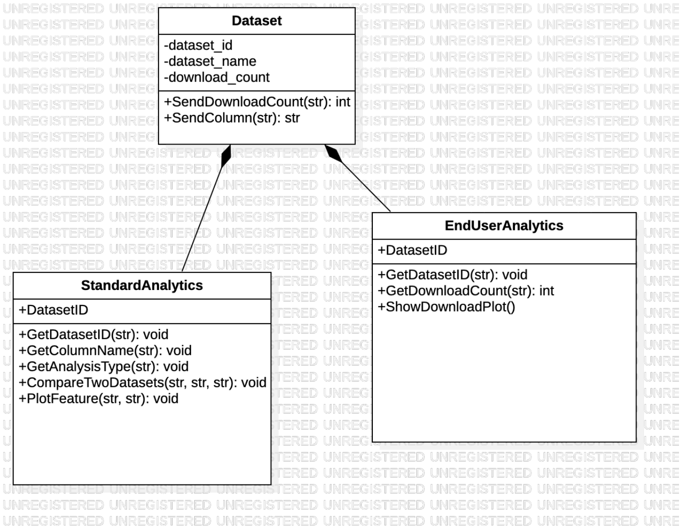


Figure 5: Class Diagram

- **GetColumnName(str): void**
  This function takes the particular column name in which the user wants to perform analysis

- **GetAnalysisType(str): void**
  This function takes what type of analysis (mean/median/etc) is to be done on the desired column

- **CompareTwoDatasets(str,str,str): void**
  This function takes ids of two datasets, and the type of analysis to be done on them.

- **PlotFeatures(str,str): void**
  This function visualizes the column and the analysis value

EndUserAnalytics class
This class helps in performing end user analysis on the dataset. It contains following attributes:

- dataset_id: contains unique id of the dataset

It contains the following methods:

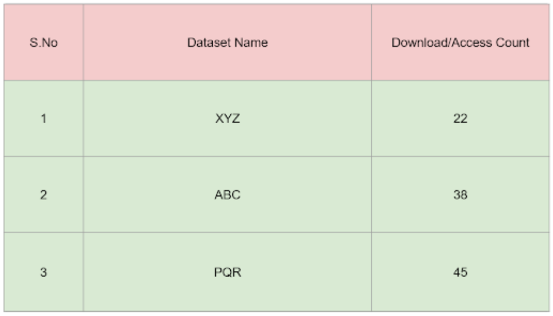
- **GetDatasetID(str): void**
  This function takes the id of dataset on which the analysis is to be done

- **GetDownloadCount(str): int**
  This function takes the dataset id and returns how many times it has been downloaded

- **ShowDownloadPlot(str): void**
  This function visualizes the dataset and its download information

4.2.2 UI Design

1. List of dataset and their download counts:



S.No	Dataset Name	Download/Access Count
1	XYZ	22
2	ABC	38
3	PQR	45

Figure 6: Displaying List of Datasets

2. Analysis of some selected column of selected dataset

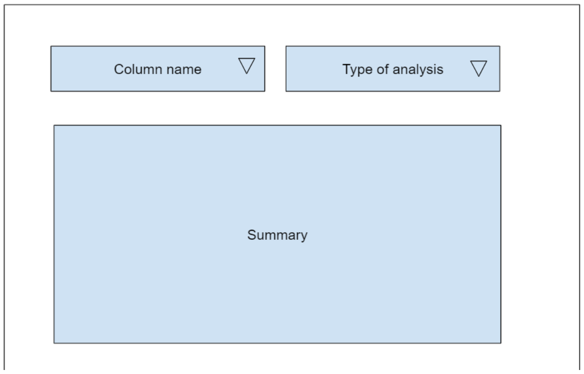


Figure 7: Selection of criteria of analysis