##### High Level Design (HLD)

##### Online EDA Automation (Web Application)

Revision Number: 0.1

Date of revision:

#### **Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| 07/11/2021 | 0.1 | Initial HLD - v0.1 | Sunny |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

##### **Table of contents**

[High Level Design (HLD) 1](#_Toc1)

[Online EDA Automation (Web Application) 1](#_Toc2)

[**Document Version Control** 2](#_Toc3)

[Table of contents 3](#_Toc4)

[Abstract 4](#_Toc5)

[1 Introduction 5](#_Toc6)

[1.1 Why this High-Level Design Document? 5](#_Toc7)

[1.2 Scope 5](#_Toc8)

[1.2 Definitions 5](#_Toc9)

[2 General Description 6](#_Toc10)

[2.1 Product Perspective 6](#_Toc11)

[2.2 Problem Statement 6](#_Toc12)

[2.3 Proposed Solution 6](#_Toc13)

[2.4 Further Improvements 7](#_Toc14)

[2.5 Technical Requirements 7](#_Toc15)

[2.6 Data Requirements 7](#_Toc16)

[2.7 Tools Used 7](#_Toc17)

[2.8 Constraints 8](#_Toc18)

[2.9 Assumptions 8](#_Toc19)

[3 Design Details 9](#_Toc20)

[3.1 Process Flow 9](#_Toc21)

[Proposed methodologies 9](#_Toc22)

[3.1.2 Deployment Process 10](#_Toc23)

[3.2 Event Log 10](#_Toc24)

[3.3 Error Handling 10](#_Toc25)

[4 Performance 11](#_Toc26)

[4.1 Re-usability 11](#_Toc27)

[4.2 Application Compatibility 11](#_Toc28)

[4.3 Resource Utilization 11](#_Toc29)

[4.4 Deployment 11](#_Toc30)

[5 Dashboards 12](#_Toc31)

[5.1 KPIs (Key Performance Indicators) 12](#_Toc32)

[6 Conclusion 12](#_Toc33)

##### **Abstract**

EDA being the manual activity which can automated. EDA methodology is one of the important activity while gaining the critical insights from the data which is practiced manually step by step. The objective is to automate the entire process through developing some pre-build basic EDA steps on the data set being tested. With addition to that it gives the ability to change the pre-build step with one’s own steps. In this context, this tool will give you graphs with proper insights including with some basic statistics.

This app will minimize the step by step activity at the most of it capability which will result in more time to analyze the data and get rich insights.

##### 1 Introduction

##### 1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model of coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

* Present all the design aspects and define them in detail
* Describe the user interface being implemented
* Describe the hardware and software interfaces
* Describe the performance requirements
* Include design features and the architecture of the project
* List and describe the non-functional attributes like:
* Security
* Reliability
* Maintainability
* Portability
* Re-usability
* Application compatibility
* Resource utilization
* Serviceability

##### 1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

##### 1.2 Definitions

|  |  |
| --- | --- |
| **Term** | **Description** |
| EDA | Exploratory Data Analysis |

##### 2 General Description

##### 2.1 Product Perspective

The Online EDA Automation is a web based exploratory data analysis application which will help us to automate the EDA process with pre-built in process and also to customize it.

##### 2.2 Problem Statement

To create an online EDA automation web based application and to implement the following use cases.

* Allow user to upload data set
* Design some prebuild EDA steps
* Allow users to customize existing prebuild steps
* Allow users to dynamically create charts and graphs based on the dataset
* Allow users to save any custom EDA steps for future use on another dataset
* Allow users to download the code if possible
* Allow users to graphical interface for execution progress
* Maintain Dashboard based on user EDA activities

##### 2.3 Proposed Solution

The solution proposed here is an online web based EDA application which can be implemented to perform above mentioned use cases.

In first case, user will be able to upload data set on which they want to perform the EDA.

In second case, the data set will worked upon some prebuild EDA steps which can further be customized by the user in the third case.

In fourth case, user will have the permission to customize and create dynamic charts and graphs based on the data set.

In fifth case, the user can also save the EDA steps for future use just like presets/template.

In sixth case, the functionality to download code for the EDA steps could be implemented if possible

In seventh and eighth step the user will interact using GUI for all the activities and dashboard can be prepared based on user EDA activities respectively.

##### 2.4 Further Improvements

Online EDA can be added with few more features like reading the data from different file formats like .json, .xml, .xls etc.

##### 2.5 Technical Requirements

This document addresses the requirements of the web application. User can perform the EDA part with the data set once logged in. Users and can reuse their saved EDA preset/template . Administrators will have access administrative abilities based on permissions given to them to view the dashboard based on EDA activities of the user.

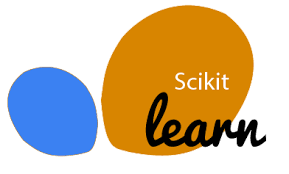
##### 2.6 Data Requirements

Data requirement completely based on the data set that the user provides to perform the EDA activities.

There care numerous file format which the user can use.

* CSV - Comma Separated Values
* JSON - JavaScript Object Notation
* XML - Extensive Markup Language
* XLS - Microsoft Excel Spreadsheet

##### 2.7 Tools Used



* VSCode is used as an IDE
* For visualization of the plots, Matplotlib, Seaborn and Plotly are used
* AWS is used for deployment of the model
* Tableau/PowerBI is used for dashboard creation
* Cassandra is used to retrieve, insert, delete and update the database
* Front end development is done using HMTL/CSS/Bootstrap
* Python, Flask is used for backend development
* GitHub is used as version control system

##### 2.8 Constraints

Online EDA Automation web based application should be minimalistic and easy to use and user friendly and prebuilt EDA step should be as informative as possible with good features.

##### 2.9 Assumptions

The main objective of the project is to implement the use cases defined previously in this document (2.2 [Problem Statement](#_2.2_Proble)) for the dataset provided by the user on which EDA needs to be performed. Pre-defined EDA steps is used for giving user with detailed knowledge about the data including the graphs and charts. It is also assumed all aspects of this project have the ability to work together in the way the designer is expecting.

##### 3 Design Details

##### 3.1 Process Flow

For the EDA part we will use some pre defined EDA steps which can be modified later on.

Below is the process flow diagram is as shown below.

##### Proposed methodologies

Process data with prebuild EDA steps

Upload the dataset

Log in screen for the user

Save the template

Customize the pre build steps

##### 3.1.2 Deployment Process

Start

Load data set

Prebuild EDA

Step Customization

Populate charts/graphs

Dashboard

##### 3.2 Event Log

The system should log every event so that the user will know the process is running internally.

**Initial step-by-step description:**

1. The system identifies at what step logging required
2. The system should be able to log each and every system flow
3. Developer can choose logging method. You can choose database logging/File logging as well
4. System should not hang even after using so many logging. Logging just because we can easily debug issues so logging is mandatory to do.

##### 3.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong. An error will be defined as anything that falls outside the normal and intended usage.

##### 4 Performance

The Online EDA Automation is a web based exploratory data analysis application which will help us to automate the EDA process with pre-built in process to save time and effort on the initial EDA activity but user can also customize it.

##### 4.1 Re-usability

The code written and the components used should have the ability to be reused with no problems.

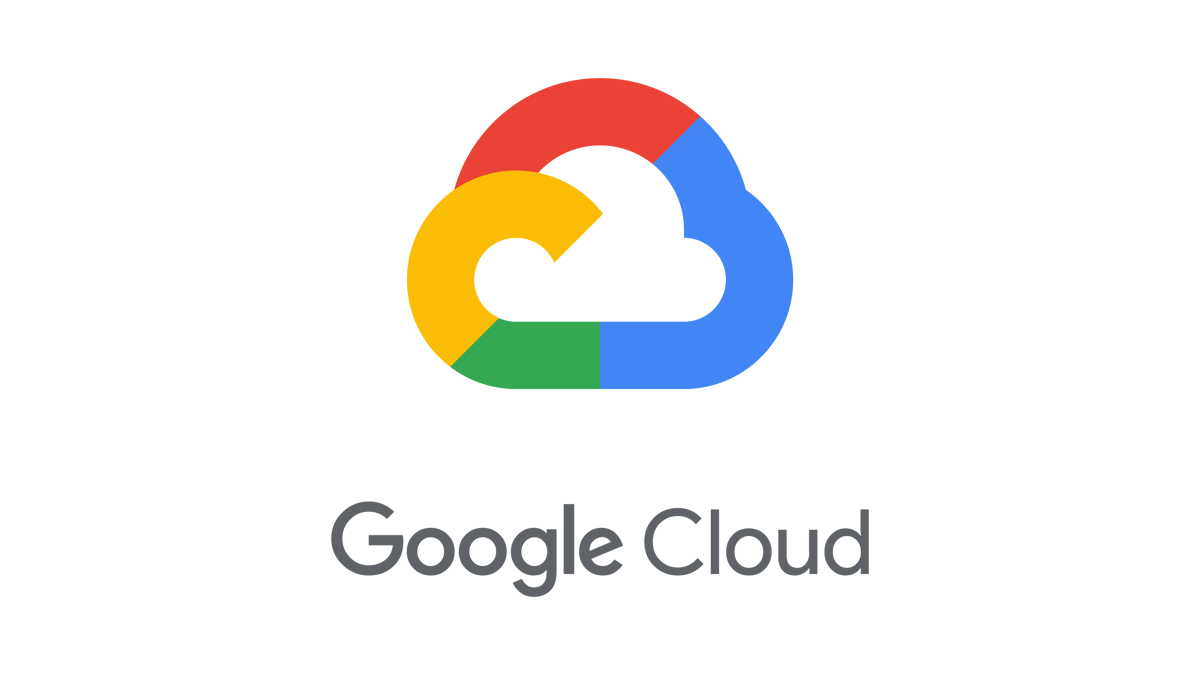
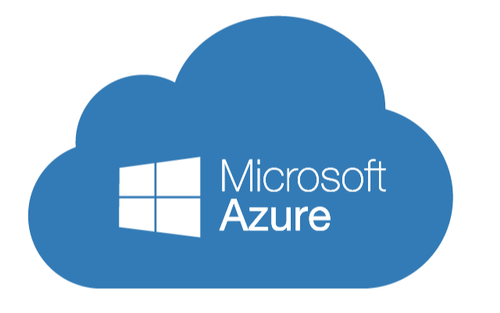
##### 4.2 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

##### 4.3 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

##### 4.4 Deployment



##### 5 Dashboards

Dashboards will be implemented to display and indicate certain KPIs and relevant indicators for the unveiled problems that if not addressed in time could cause catastrophes of unimaginable impact.



##### 5.1 KPIs (Key Performance Indicators)

1. Key indicators displaying the summary of the customized EDA steps based on the user
2. Time and effort reduction on EDA using the Online EDA Automation web application
3. To detect the data sets
4. Data visualization methods
5. No of concurrent projects for EDA

##### 6 Conclusion

The Online EDA Automation is a web based exploratory data analysis application will help us to automate the EDA process with pre-built in process to save time and effort on the initial EDA activity but user can also customize the steps as they see fit and save it as a template for future use.