CREATING DASHBOARD USING SUPERSET BY API CALL TO DIFFERENT SOURCE

[1]Anita Halundi, [2]Dr Minal Moharir
[1]Student, [2]Associate Professor
[12]Department of Computer Science Engineering,
[12]R.V. college of Engineering, Bengaluru, India

Abstract *:* As failure of product or project is due to not keeping track at each step. This happens when bulk data needs to be analyzed and then visualized. In order to make things easier to analyze, we have to opt to data visualization and exploration tool, which provides better visuals for given dataset. Superset is one of the opensource business-intelligence(BI) tool used for visualization and data exploration. Superset tool provide wide variety of charts and helps to build interactive dashboard. The dataset can be brought from any source like defect management tools through API calls. The response data form the API calls will be either in json or xml format. The data has to be processed and stored into database, so that the same dataset can be populated in the dashboard using superset. Superset supports wide range of SQL databases by using python ORM. The proposed system provides big data analysis within short period and perform accurate charts on the superset dashboard.

***IndexTerms* - Superset, Business intelligence(BI), SQLAlchemy**

I. INTRODUCTION

Data visualization is a practice of transforming data information into visual elements such as graphs , identify patterns and outlier in large dataset. Main objective, to build dashboard is to keep track of large dataset fetched form different source. As bulk data is stored in different project management tools related to application lifecycle management, project defects and related information are to be fetched from these source. In the existing system, the analysis of data had to be exported to excel sheet where the visualization had to be done manually, which may not be accurate and time consuming. So to automate the visualization, we use business intelligence tool called apache superset. The dashboard that we create need to be simple attractive and interactive. In superset, executing of query is fast and the UI of superset is simple as well as user-friendly.

1.1 Benefits of Visualization

- Dashboard provides better understanding of next step that must be taken to improve the organization, by keeping track of application lifecycle and defects in the project.
- Ability to grasp information quickly, provide better insights and decision can me made faster based on the problem statement.
- In traditional reporting system, some insights of the projects or trends may be missed, where it cannot be missed in data visualization tools.
- Project team can communicate with each other for findings in more productive ways to be enhanced with the help of visualization tools.

1.2 Overview

The dashboard created ought to arrange API calls, local storage of bulk data into database and then populate the same data in the form visualization.

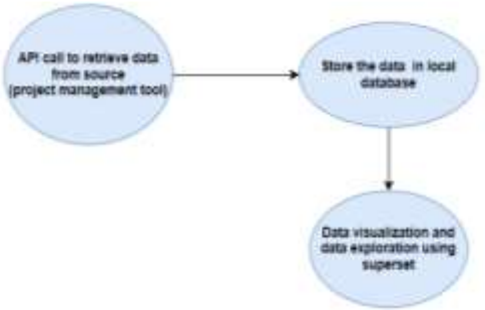

API call to retrieve data from source (project management tool)
Store the data in local database
Data visualization and data exploration using superset

Fig. 1 Basic Workflow

This dashboard is mainly built for project managers, so that they can keep track of verification, project defects, application lifecycle management etc. Dashboard is divided into few modules performing various tasks like filtering, trendline and various chart.

Visualization of data from different source involves the following steps :

- Authenticate and fetch data from various source like defect management tool to get the defect data of a particular project through API call.
- Retrieved data has to be processed and store the same data into local Database.

- Fetch the data from local database and then populate the data in the form of visualization charts.

II. RELATED WORK

In the trending market research there is an increasing point on visualization. Data visualization allows to obtain useful trendline and data with simple and fast processing.

Some projects also opt the same approach by using superset, as a business intelligence(BI) tool for data Analytics. There are many other tools such as Tableau, Google Data Studio, Microsoft Power BI that offer the possibility of use API calls and that could allow integration with superset tool. Complex dashboards can be created with maximum flexibility by using Tableau. But the limitation is that the free version is allowed only by a desktop application.

Google Data Studio tool is very efficient but it does not allow to use more than one dataset within the same dashboard. that was one limitation. Google data studio tool has a very few data connectors that is only MYSQL and Postgres.

In Plot.ly it allows You to create and share the interactive dashboard very quickly. But the limitation is that it only accepts datasets with a max size of 5 MB and the dashboard published must be public or else it is necessary to pay a subscription amount to make it private. Among the opensource tools available ,we have distinguished three visualization tools: Superset, Metabase and Redash. Superset and Redash are almost same. Both provide a large dataset source .In addition to this they provide a interactive interface for executing and writing SQL queries.

III. METHODOLOGY

3.1 Overview Of Modules

3.1.1. Analyze the project management tools

First step keeping in mind that, what dataset need to be populated and from which source it has to be fetched need to be known and then user need to analyse the tool.

3.1.2. API call to fetch data from different source

API request is made using http GET method to fetch the data from the source. If required, the authorization parameter has to be passed in the headers, while requesting data fetch.

3.1.3. Converting the response data to json format

Data fetched from the http response is then converted to json format, then iterate through the json object to get appropriate data.

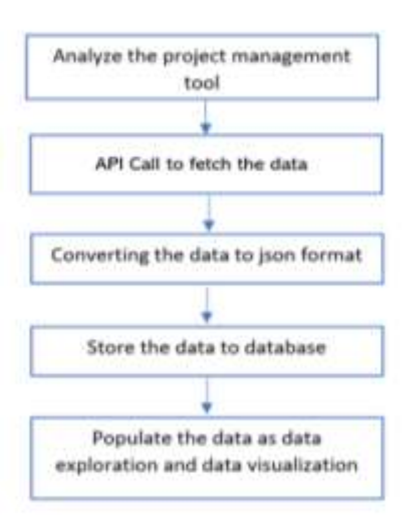

Analyze the project management tool
API Call to fetch the data
Converting the data to json format
Store the data to database
Populate the data as data exploration and data visualization

Fig. 2 Methodology

3.1.4. Store the formatted data to Database

While iterating through the json file, map the data to the database column and store the same in to PostgreSQL database.

3.1.5. Populate the data for data exploration

At this stage, the connections to apache superset is done using SQLAlchemy and data is pulled from PostgreSQL database and create a dataset in superset. Charts can be created with this data and interactive dashboard can be build with combining all the charts in a single dashboard.

3.2 Implementation

REST is the acronym for REpresentational State Transfer. RESTful Web Services are Web Services that are built on the REST Architecture. Everything is a resource in the REST Architecture. Each endpoint is a location where APIs may have access to the resources through URL , they require to do their tasks. To get access from the resources usually the format would be JSON or XML.

For coding we use Python language. In python for API call it provide requests library . when the request is made, the response from the API would be status code which tells whether the request was successful or not. In order to make 'GET' request, we will use requests.get() function which requires URL as a parameter.

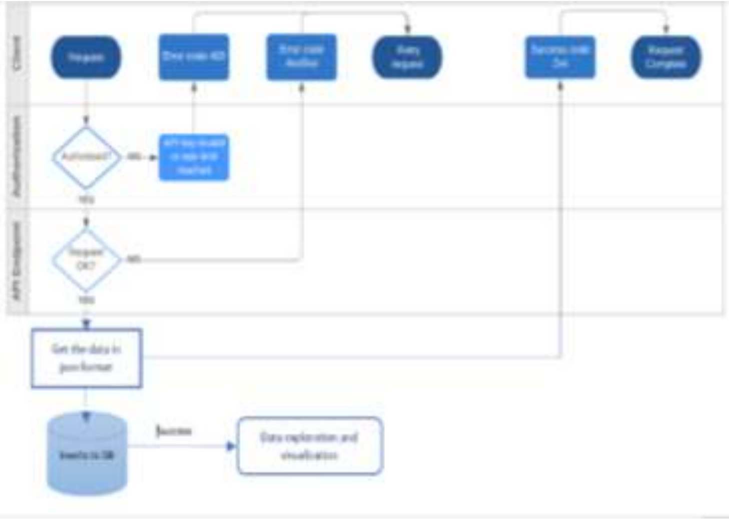


Fig. 3 Implementation of proposed system

Connections to database are done through SQLAlchemy toolkit and the processed json data is pushed into db. OpenAlchemy toolkit is used to provide data models through yml file and directly create a dataset tables in PostgreSQL database without manual creation of columns. Connection to superset is enhanced with the same toolkits.

The data Response data from API call is then converted into json file . then iterating through this json dictionary and map to each columns in the PostgreSQL database. As the database have all the necessary data available ,it has to be reflecting into superset.

In superset create a database and datasets. while creating a dataset, provide table name and save the dataset in superset. Now you can use the same dataset to populate the data and visualize it. Create wide variety of charts and combine all the charts into a single dashboard. There is editing of CSS to make the dashboard more interactive styling. There are multiple charts that you can explore, such as filter box that provides dynamic filter. Hence these filters can be applied to multiple charts which are combined in the dashboard. You can also query for external functionality to make the dashboard look more interactive .

IV. RESULTS

In Fig.4 the snapshot of the postman tool is shown where you can test your API calls for better exploration. In this tool you can provide request URL, pass the header information as a parameters, authorization field suggest provide login credentials for login to restricted websites. If the status code of the API call is 200 then the response header displays the json formatted data.

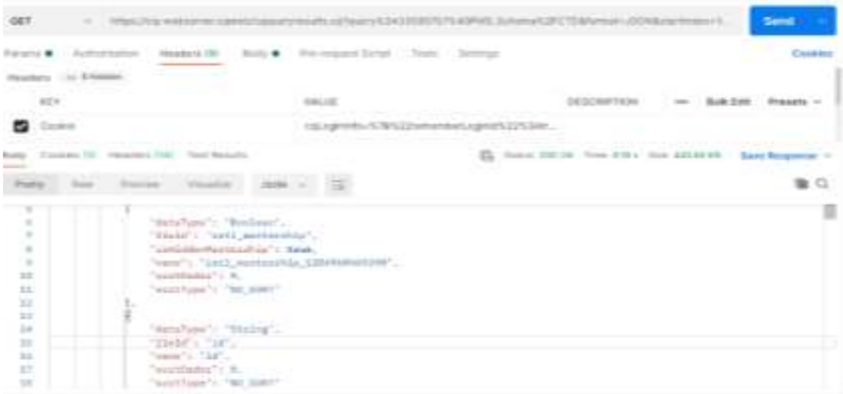
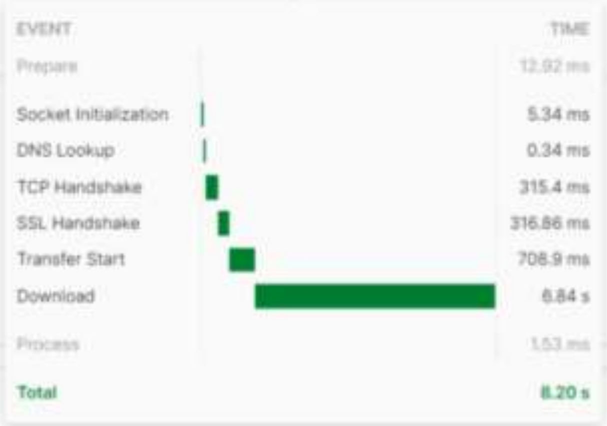


Fig. 4 Snapshot of Postman tool for testing API calls



EVENT	TIME
Prepare	12.92 ms
Socket Initialization	5.34 ms
DNS Lookup	0.34 ms
TCP Handshake	315.4 ms
SSL Handshake	316.86 ms
Transfer Start	708.9 ms
Download	6.84 s
Process	1.53 ms
Total	8.20 s

Fig. 5 Performance of API call To fetch bulk data

Fig.5 gives the overview of time taken to establish the connection and fetch data from source. As the data is bulk it takes about 6.84 s to fetch the data. Overall time take to request and get the response would be 8.20 seconds

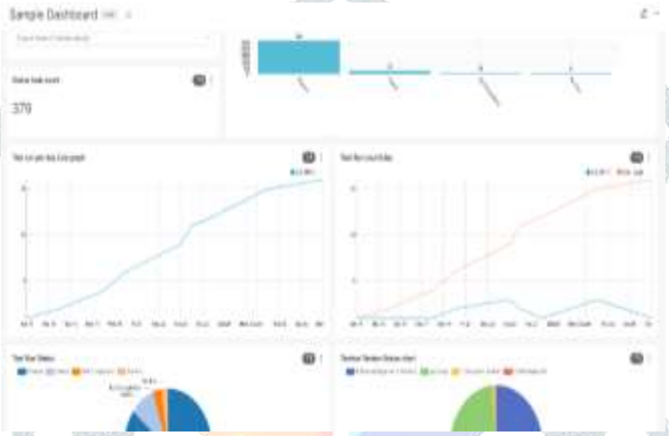


Fig. 6 Superset Dashboard

Fig 6 shows the dashboard view with line graph, big number chart and many more chart, which makes the visualization of data easier.

V. CONCLUSION

The paper presents a way to build interactive dashboard by fetching the data from various source like project management tools and store into local database. The proposed system provides big data analysis in a very short time and perform accurate charts on the dashboard. As bulk data is pulled from the source management tools ,it takes about 6.84 s to fetch the data. Hence overall time taken to request and get the response would be 8.20 seconds. This helps in taking appropriate decision in the project lifecycle as well. As the requirement changes, the dashboard can be altered dynamically according the user needs i.e the charts can be altered whenever required.

REFERENCES

[1] Michele Petito, Francesca Fallucchi and Ernesto William De Luca, “Create Dashboards and Data Story with the Data & Analytics Frameworks,” DOI:10.1007/978-3-030-36599-8_24 December 2019,pp.272-283

[2] Matthew, Adebowale, Cajetan and Sarhan , “DATA VISUALIZATION” December 2016

[3] J. L. V. Sancho, J. C. Dominguez, and B. E. M. Ochoa, “An approach to the taxonomy of data visualization,” Revista Latina de Communicacion Social, vol. 69, 2014, pp. 486-507.

[4] Desouza KC, Jacob B (2017) Big Data in the Public Sector: Lessons for Practitioners and Scholars. Adm Soc. https://doi.org/10.1177/0095399714555751

[5] Apache Software Foundation, Apache Superset Contributors (2018) Apache Superset 12 — Apache Superset documentation.