

# Business Requirement Document

## - Introduction

The main idea behind developing this kind of project is for keeping a track of your project and contributors. An API like this is widely used in the IT Sector for project tracking and detailed data on the contributors.

It also allows you to manage and get statistical data for your repositories, branches, take note of all the pull requests on the code and various other management tasks.

## - Stakeholders

Name	Role
Mr Deepak Mishra	Client / Project Owner
Mr Dinesh Behera	Coordinator / Advisor

## - Project Requirements

Initially, we create a user token for each user to access the API, through that we send a server request to the Github API, after receiving the data we trim and normalize it according to the client's requirements.

After processing the data we either display the data into the console or into a GUI.

The GUI to be built depending on the tech stack preferred by the client

- Tkinter GUI
- HTML & CSS webpage

So the input requirements are as follow :

- A Github token or key (Provided by Github REST API)
- Github ID ( Query to send data request with )

The output will be as follow :

- Repository Name
- A list of all the contributors to the repository
- All the list of commits ( Start - End Date )
- After the end of the query, it will return if the request was successful or failed

Deploying the API :

- This project will be deployed on the AWS Lambda function and all the request shall be made from the AWS servers

## **- Tech Stacks**

### **• Tech Stack (1) [As a web app]:**

- Using frontend & Backend Technologies (HTML & CSS) + (Javascript)
- Sending a fetch request to the GitHub API using javascript
- Receiving bulk information on the repository and trimming the data
- Displaying the data through dynamic DOM Rendering

### **• Tech Stack (2) [GUI using python tkinter]:**

- Using python and external dependencies like (requests, pyGithub) to send requests to the API
- Receiving bulk data and trimming it to required data
- Displaying in on a GUI using Tkinter