# GitHub Analytics

DISSERTATION

Submitted in partial fulfilment of the requirements of the

M. Tech. Software Engineering Degree programme

By

**NIKESH T T**
**(2018HS70029)**

Under the supervision of

**LALLU ANTHOOR**
**(Senior Developer)**

Dissertation work carried out at
SAP Labs, Bangalore

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

April, 2022

# GitHub Analytics

DISSERTATION

Submitted in partial fulfilment of the requirements of the

M. Tech. Software Engineering Degree programme

By

**NIKESH T T**
**(2018HS70029)**

Under the supervision of

**LALLU ANTHOOR**
**(Senior Developer)**

Dissertation work carried out at
SAP Labs, Bangalore

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

April, 2022

# Acknowledgement

I feel proud to present my dissertation project on the topic “GitHub Analytics”.

I would like to show my gratitude for Lallu Anthoor, Senior Developer, SAP for investing time from his busy schedule in mentoring me.

I extend my sincere and heartfelt thanks to my examiners, Sudeep Sukumar &
Prof. Bhaskar K for providing me with the right guidance and advice at the crucial junctures and for showing me the right way.

In preparation of this project, I had to take the help and guidance from many online communities which helped me to get familiar with the technology and its implementations. I would also like to expand my gratitude to my teammates and colleagues who have directly and indirectly helped me whenever I faced any blocker.

This Mid Term report is for my undergoing course of Birla Institute of Technology & Science, Pilani of WILP course in M.Tech Software Engineering**.**

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
SECOND SEMESTER 2021-22

SESAP ZG629T DISSERTATION

| | | |
|---|---|---|
| Dissertation Title | : | GitHub Analytics |
| Name of Supervisor | : | LALLU ANTHOOR |
| Name of Student | : | NIKESH T T |
| ID No. of Student | : | 2018HS70029 |

# Abstract

A large number of organizations are using GitHub as their primary source for collaborative development. GitHub offers a great support for enabling development towards a common goal when it comes organizations having a huge employee strength. One of the examples is the trunk-based development where the changes are made in small increments in separate child branches and merged into the master branch once the development activity is complete. Due to the huge number of robust features which GitHub offers many companies uses GitHub as part of their tech stack.

When it comes to cloud-based development it is important to ensure the cloud qualities of the product. There is a good amount of metadata generated as part of any operations which are performed in the GitHub. If we enable proper ways for extracting this metadata many useful information can be derived using the metadata which will help in ensuring the cloud quality of the product.
GitHub Apps are a great way to retrieve the metadata from GitHub.

## Overview of GitHub Apps

- Modern way for third party application or users to integrate with the GitHub via the API (i.e., building integrations on top of GitHub)
- Can be considered as a bot that you give access to different objects within your GitHub repos and allow those bots to automate/check things for you

## Benefits of GitHub Apps

- Autonomy - The best way for third parties to act autonomously on protected GitHub resources
- Improved security model - Tokens expire for additional security
- Dedicated rate limits
- Used by GitHub to develop its own products
- Offers significant benefits over OAuth app for third party integrators
- Supported by rich ecosystem of libraries and tools

- Granular permissions
- Singular WebHook for all events (across different organizations and repositories)
- Grantable on individual resources (e.g., giving access to only one repo in the organization)
- Better insights to user identity

Once the metadata is retrieved it could be transformed to extract out the useful information and to perform various data analysis on the derived information. This useful information can be visualized in a central UI which will help in ensuring many of the cloud quality aspects of the product which is being developed.

# List of Symbols & Abbreviations Used

| Term | Definition |
|---|---|
| GitHub | GitHub, Inc. is a provider of Internet hosting for software development and version control using Git. |
| GitHub Repository | Repositories in GIT contain a collection of files of various different versions of a Project |
| PR | A pull request is an event in Git where a contributor asks a maintainer of a Git repository to review code they want to merge. |
| GitHub Webhooks | GitHub Webhooks allow you to create or set up integrations on GitHub server that subscribe to specific events. |
| GitHub App | An individual who contacts the SaaS application to delete his or her personal data. |

# List of Tables



# List of Figures

# Table of contents

# Introduction & Background

The GitHub analytics tool will help the user to collect various meta information regarding multiple aspect of the GitHub repositories owned by the user. The tool will help to transform the collected information and derive many useful insights out of it. Meta information includes the events raised by the GitHub while performing create, update, or delete operations on the GitHub repositories. Following are some of the examples of events raised with the meta information: a new branch is created in the repository – CREATE event, a new pull request is created - PULL_REQUEST event, user adds a comment to one of the existing issues – ISSUE_COMMENT event. Using the extracted data we can derive many meaningful insights like state of open PRs, inactive branches in a repository, quality of the PR reviews, ownership of different modules, stability of the pipeline.

Existing process of collecting the information involves a lot of manual work. Activities such as monitoring the list of open PRs, monitoring the state of available branches in a GitHub repository, ensuring the quality of PR reviews, ensuring the stability of the pipeline etc. requires the user to manually collect the data related to each of these categories from the GitHub.

Some of the available solutions involve active monitoring of the GitHub data which put a lot of pressure on the GitHub server.

This application will help the user to derive meaningful insights and analytics from the GitHub using a passive way of collecting the data. This will also reduce the load on the GitHub server side.

# Business Process Flow

The Business Process Flow for the GitHub analytics tool is as follows:

- Data Extraction - Passive data extraction using GitHub Webhooks
    - Creation of GitHub App with the required configurations required for the data extraction
    - Installation of the GitHub App in the target organizations
    - Querying the GitHub APIs to fetch the initial set of data
    - Passive data extraction via GitHub Webhooks
- Data analysis – Data analysis and generation of meaningful insights
    - Persistence of the useful data
    - Defining SQL Views to fetch aggregated results from the bulk data at run time
    - Cleanup of unused data
- Visualization - This feature allows the users to view the derived information using graphs and tables in a centralized UI
    - Plotting various graphs and table using the available python libraries to make best sense out of the available information

# Problem Statement

Existing process of collecting meta data from GitHub involves lot of manual work and it compels the user to manually collect the data related to each of these categories from the GitHub.

Some of the available alternative solutions involved active monitoring of the data:

- Active monitoring require user to query the GitHub APIs to fetch the data, It involves:
  - A lot of data transfer
  - Good amount of processing on the server side
- Data is fetched in a scheduled manner during fixed time periods. Hence the real-time data is not used for the analytics. The fetched data is already old.
- Users need to fetch the complete set of data every time

## Objective of the Project

The objective of the project are as follows:

- To gather the GitHub repository data using passive methods
- Deriving meaningful insights from the gathered data
- Building a centralized UI for the visualization of the information

## Uniqueness of the Project

There are already existing tools to gather the available GitHub data through active monitoring.

The proposed solution is unique in many ways. Here data is gathered using passive data collection methods to reduce the load on the GitHub server. Instead of displaying the gathered data as it is, different aspects of the gathered data is analyzed to derive many useful insights such as state of open PRs, inactive branches in a repository, quality of the PR reviews, ownership of different modules, stability of the pipeline. Aggregated results are calculated using the gathered data. Derived meaningful information and the aggregated results are displayed on a centralized UI so that user will be able view all of the information at a single place.

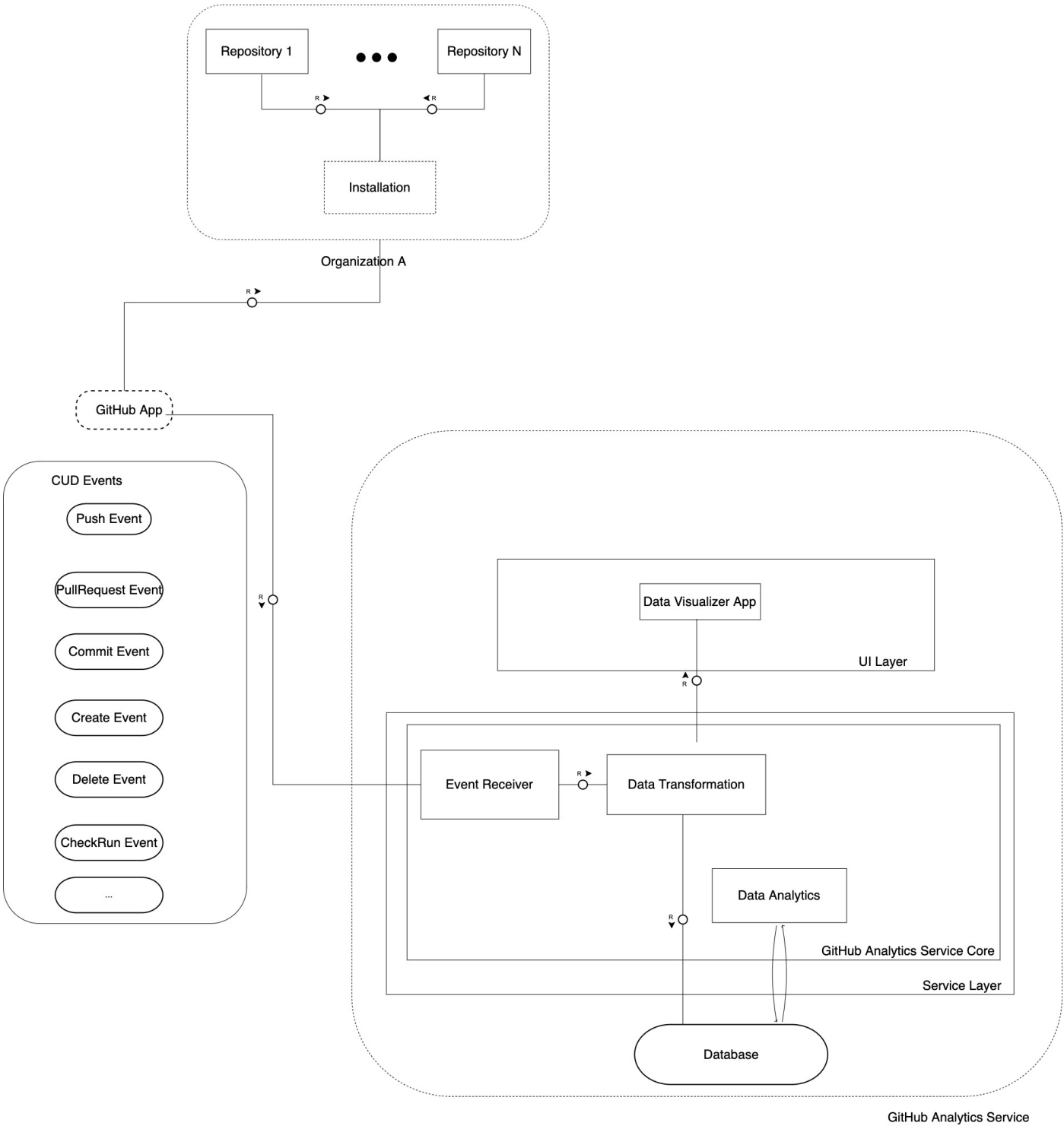
Since the data extraction and the analysis process is dynamic in nature user does not have to put any additional manual effort here. Using Webhooks, the data will be transferred as delta to the baseline, we need to query the system only once to set the initial state. Rest of the changes will come as delta via the Webhook deliveries.

Using the extracted data we can derive many meaningful insights like state of open PRs, inactive branches in a repository, quality of the PR reviews, ownership of different modules, stability of the pipeline

## Scope of Work

The scope of this dissertation is to make use of the available GitHub Enterprise Server APIs to extract the useful GitHub repository data and perform various analysis on the collected data to generate meaningful insights and analytics. The data extraction is passive using Webhooks. This way we are not adding any additional load on the GitHub Enterprise Server. The derived information will be visualized in a meaningful and useful way using static and dynamic time series charts so that the consumer will be able to take necessary actions based on the information.

# Solution Architecture



## GitHub App Installation

- Connects a GitHub app to one or more repositories owned by organization or user
- Here the two installations on the different organizations are independent, so they are isolated from each other
- But they point back to the same GitHub app
- Permissions and WebHook handlers are shared

## GitHub Analytics Service

- GitHub Analytics Service can be majorly categorized into three components
    - UI Layer – Visualization of the information
    - Service Layer – Receiving and processing of the data
    - Database Layer – Persistence of the transformed data

When create, update or delete operations happens in any of the repositories where the GitHub App is installed, it will raise events to the endpoint which is configured as the WebHook URL. As an example, we could consider the following scenarios:

- A new branch is created in the repository – CREATE Event
- A new pull request is created - PULL_REQUEST Event
- User adds a comment to one of the existing issues – ISSUE_COMMENT Event

These events will be received by the controller layer present inside the GitHub Analytics Service Core. Once the payload is received Payload Conversion Service takes care of mapping and converting the payload with the help of predefined model classed. Data transformer service takes care of converting the data to extract out the useful information from the data, it also interacts with the Data Analytics module to generate the aggregated results.

Once the data is transformed to the desired format it is persisted inside the data base layer.

UI layer takes care of representing the useful information to the user in a centralized UI by plotting various graphs and table. UI layer also interacts with the service layer in order to fetch the persisted data.

# Resources Needed

The various resources required for the project are:

Hardware Requirements:

- Operating System – Windows 10 Enterprise Edition (64 bit)
- Processor – Intel Core i5-6300U CPU
- RAM – 16 GB

Software Requirements:

- GitHub App
- Java
- Python

# Project Plan & Deliverables

| Serial Number of Task/Phases | Tasks or subtasks to be done | Start Date-End Date | Planned duration in weeks | Specific Deliverable in terms of the project |
|---|---|---|---|---|
| 1 | Identifying available GitHub Enterprise Server APIs | 27/02/'22 – 05/03/'22 | 1 | Detailed list of information about usable GitHub Enterprise Server APIs |
| 2 | Application module to handle GitHub APIs | 06/03/'22 – 12/03/'22 | 1 | GitHub APIs management application module |
| 3 | Explore extractable GitHub repository data | 13/03/'22 – 19/03/'22 | 1 | Information about extractable GitHub repository data |
| 4 | Application module to extract the useful GitHub repository data | 20/03/'22 – 02/04/'22 | 2 | GitHub repository data management application module |
| 5 | Explore various data analysis methods/tools | 03/04/'22 – 16/04/'22 | 2 | Details regarding various data analysis methods/tools in tabular format with pros and cons for each |
| 6 | Data analysis and generation of meaningful insights | 17/04/'22 – 07/05/'22 | 3 | Extracted meaningful insights from the gathered data |
| 7 | Testing and Bug Fixes | 08/05/'22 – 14/05/'22 | 1 | Make the application stable |
| 8 | Explore Python based visualization tools | 15/05/'22 – 28/05/'22 | 2 | Details of Python based visualization tools |
| 9 | Visualization of the information using static and dynamic time series charts | 29/05/'22 – 18/05/'22 | 3 | Consumer UI for visualization of the information |

# Key challenges faced during the project

- Identifying & categorizing the useful information available as part of GitHub WebHook payload
- Understanding the architecture of the GitHub App
- Developing Application module to handle GitHub APIs

# Plan for Remainder of the Project

| Serial Number of Task/Phases | Tasks or subtasks to be done | Start Date-End Date | Planned duration in weeks | Specific Deliverable in terms of the project | Status |
|---|---|---|---|---|---|
| 1 | Identifying available GitHub Enterprise Server APIs | 27/02/'22 – 05/03/'22 | 1 | Detailed list of information about usable GitHub Enterprise Server APIs | Done |
| 2 | Application module to handle GitHub APIs | 06/03/'22 – 12/03/'22 | 1 | GitHub APIs management application module | Done |
| 3 | Explore extractable GitHub repository data | 13/03/'22 – 19/03/'22 | 1 | Information about extractable GitHub repository data | Done |
| 4 | Application module to extract the useful GitHub repository data | 20/03/'22 – 02/04/'22 | 2 | GitHub repository data management application module | Done |
| 5 | Explore various data analysis methods/tools | 03/04/'22 – 16/04/'22 | 2 | Details regarding various data analysis methods/tools in tabular format with pros and cons for each | In Progress |
| 6 | Data analysis and generation of meaningful insights | 17/04/'22 – 07/05/'22 | 3 | Extracted meaningful insights from the gathered data | In Progress |
| 7 | Testing and Bug Fixes | 08/05/'22 – 14/05/'22 | 1 | Make the application stable | In Progress |
| 8 | Explore Python based visualization tools | 15/05/'22 – 28/05/'22 | 2 | Details of Python based visualization tools | In Progress |
| 9 | Visualization of the information using static and dynamic time series charts | 29/05/'22 – 18/05/'22 | 3 | Consumer UI for visualization of the information | In Progress |

***FORM BITS – D - 1***

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

**SECOND SEMESTER**

**SESAP ZG629T DISSERTATION**

**MID SEMESTER EVALUATION FORM**

**Section I**

ID No. 2018HS70029

Name of Student: NIKESH T T

Name of Supervisor: LALLU ANTHOOR

Name of the Examiner(s): SUDEEP SUKUMAR, Prof. BHASKAR K

Dissertation Title: **GitHub Analytics**

**Section II**

**Comments on the dissertation from Examiner and Supervisor (Select Y or N)**

1. **Quantum of work**
   a. **Justifiable as efforts for 8 weeks duration** **Y**
   b. **Work is in line with the commitments made in outline** **Y**

2. **Type of work**
   a. **Client assignment** **N**
   b. **Organization specific task** **Y**
   c. **General study project such as white paper** **N**
   d. **Any other (kindly elaborate below in a line or two if Y)** **N**

3. **Nature of work**
   a. **Routine in nature** **N**
   b. **Involved creativity and rational thinking** **Y**

   **Kindly elaborate below if answer for above is "Y"**

   The candidate has to identify the parameters that are relevant for an organization for determining the quality of their engineering processes, identify the information that are available from the GitHub server and come up with a set of metrics that are easily calculatable and provide value to the organization.

4. **Evaluation methodology**
   a. **Evaluation done based on presentation to supervisor and examiner** **Y**
   b. **Evaluation done through Viva conducted by supervisor and examiner** **Y**
   c. **Student regularly interacted with supervisor and incorporated the suggestions made** **Y**

**d. Brief description on the report submitted, quality of presentation and suggestions given for improvement**

| |
|---|
| Additional sheet may be used if more space is required<br><br>Inexperience of the candidate in preparing report/diagrams were clearly visible in the initial version of the document. But he was able to incorporate the comments fully and produce a good quality report. |

**5. <u>Mid semester evaluation matrix</u>**
**Tick the appropriate box (1 is lowest and 5 is the highest)**

| **Dimension** **Rank→** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| **Student abilities in general** | | | | | |
| Understanding of the subject of dissertation | | | | ✓ | |
| Creative thinking: ability to come up with new ideas | | | | ✓ | |
| | | | | | |
| **Viva / Seminar presentation** | | | | | |
| Communication ability | | | | | ✓ |
| Organization of material | | | | ✓ | |
| Response to review questions | | | | | ✓ |
| Cohesive thinking ability | | | | | ✓ |
| **Report submitted** | | | | | |
| Report structure and format | | | | | ✓ |
| Technical content of the report | | | | | ✓ |
| Explanation on the significance of the assignment | | | | | ✓ |
| Analysis of alternative approaches | | | | | ✓ |

Any other comments:

Date: __13-04-2022____ Signature of examiner(s) Signature of Supervisor

**Note: Additional paper can be used for including further comments that is relevant to the work, if required**