CS7NS1 Scalable Computing
Supplemental Task

Trinity College Dublin, University of Dublin

Lakshay Sharma

August 5, 2018

Summary

This report outlines the work done in preparing a REST service for GitHub. The application provides a very detailed information about the owners of the repositories submitted and their actions based on statistics.
The report is divided into parts according to the requirements of the project described in the task in the email.

All the parts states 3 basic questions:

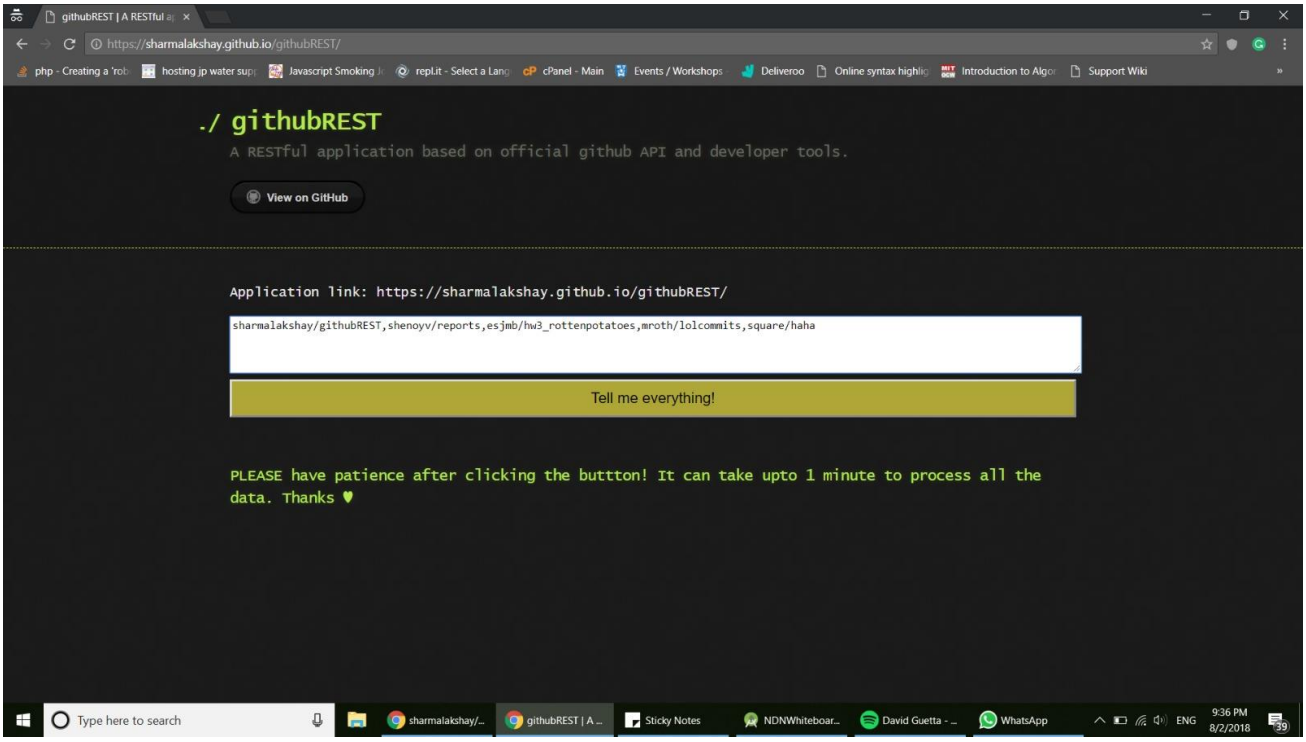
1. What resources I used.
2. How I used those resources, method, and other technical details.
3. What I achieved.

Link to the project repository on GitHub:
https://github.com/sharmalakshay/githubREST

Link to the application:
https://sharmalakshay.github.io/githubREST/

Direct GET requests can be made to:
http://iamlakshay.com/githubREST/?repos= [user] / [repo_name] , [user] / [repo_name] …
Example:
http://iamlakshay.com/githubREST/?repos=sharmalakshay/pantaloons,shenoyv/reports

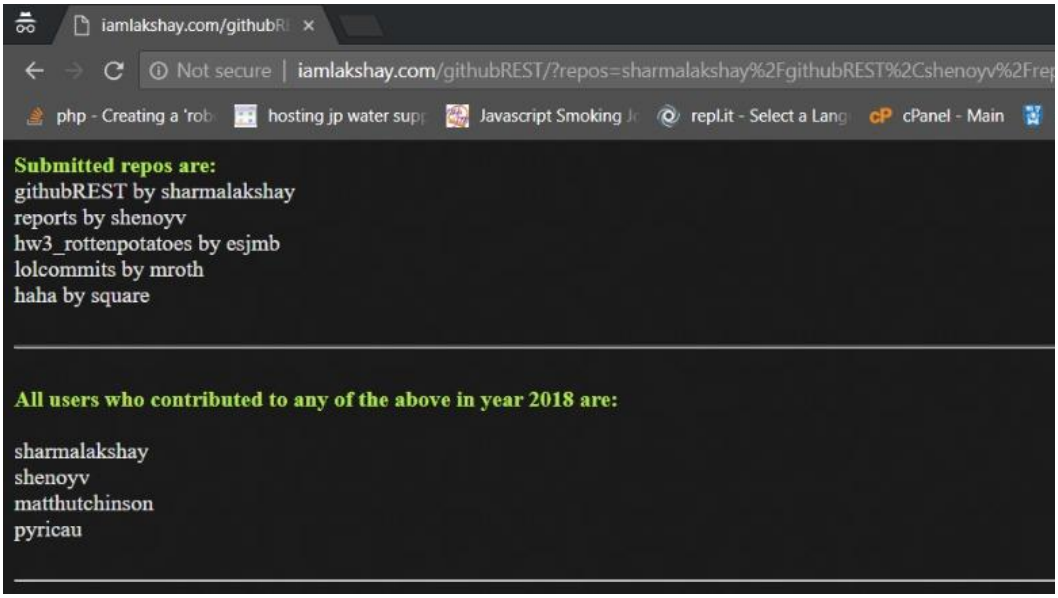
You are required to deliver a REST service, that provides an interface for submitting a set of GitHub repositories (identified as a list of strings of the form "username/repositoryname").



I used GitHub pages to serve my homepage of the application to the users. One can put all the repositories full name (user/repository) separated by commas and press the button that sends the data through GET method to my personal servers. Reason behind this is that GitHub Pages does not support Php or any server-side scripting.

I used one of the Jekyll theme (Hacker theme) for the GitHub Pages.

Your service should identify the set of GitHub developers who have contributed to any of these repositories in 2018



I load results from *https://api.github.com/repos/**repository_name**/commits* for each repository that is submitted and then check if the commit from a user is newer than 2018, if yes, the user is added to an array which is displayed here and used for many more things further in the program.

For this user set, provide an ordering that ranks these users according to each of the following criteria:

1. **Total number of commit contributions to any project to which a user has a contributed.**

I load results from *https://api.github.com/search/commits?q=author:**username*** using cURL. This gives all the commits made by a user along with a total number of results which is displayed on the screen for each user.

It cannot be accessed directly because additional headers are to be added and that was the reason to use cURL.

---

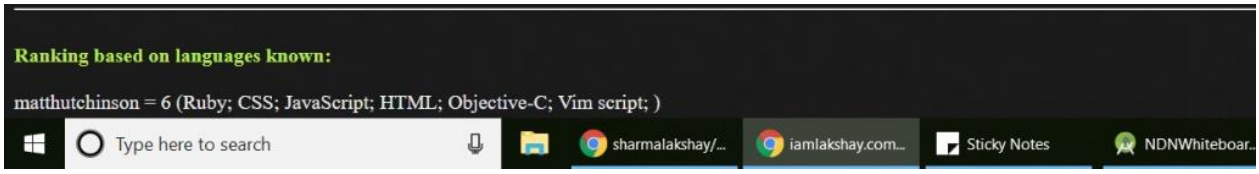
2. **Total number of commit contributions as above but restricted to projects that are members of the original submitted set.**

I used the array of commits for every repository that we loaded before, then filtered out the commits only by the set of users that were active in 2018.

I added the total number of commits by each user from the set, sorted it in the descending order, and displayed it.

---

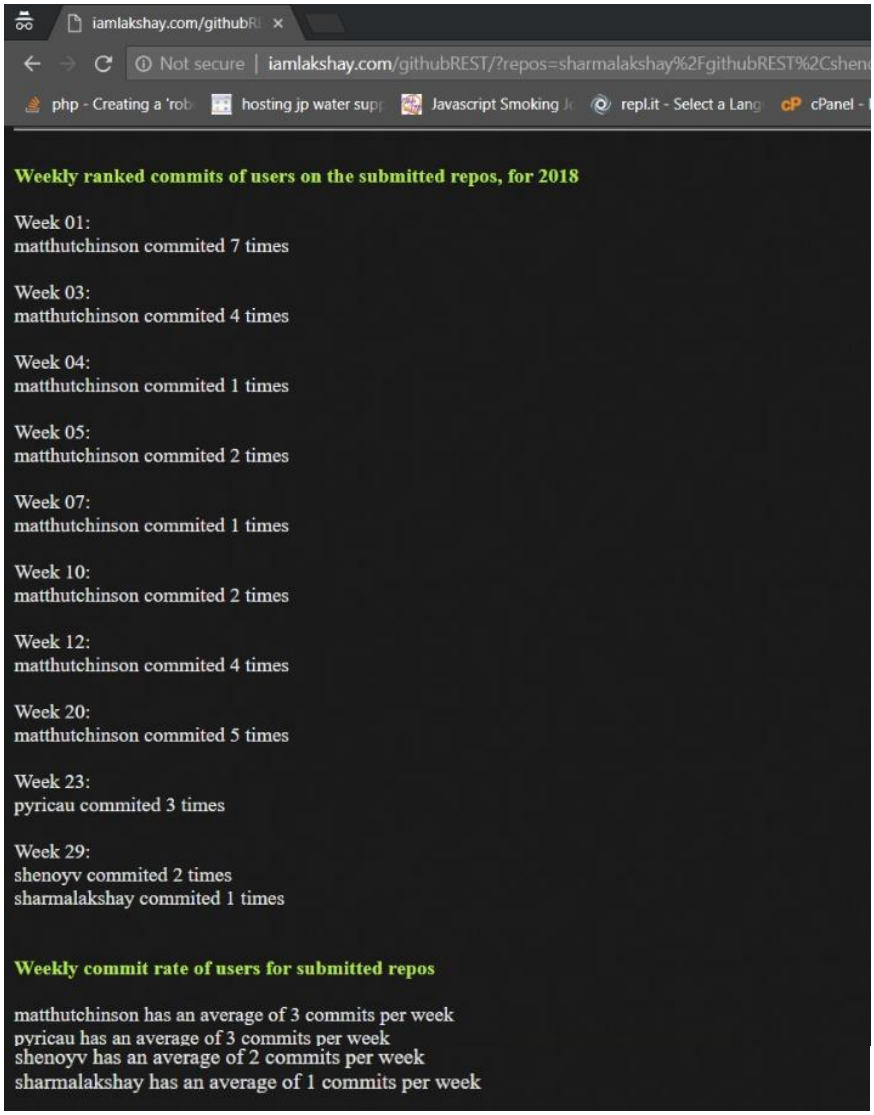
3. **The number of known programming languages for each user (presuming that the languages of any repository committed to are known to the user).**



I load the results from *https://github.com/users/**user**/repos* and for each user and iterate over the repositories checking what language they are in. I add them into array, sort it in descending order, and display the total, along with the languages.

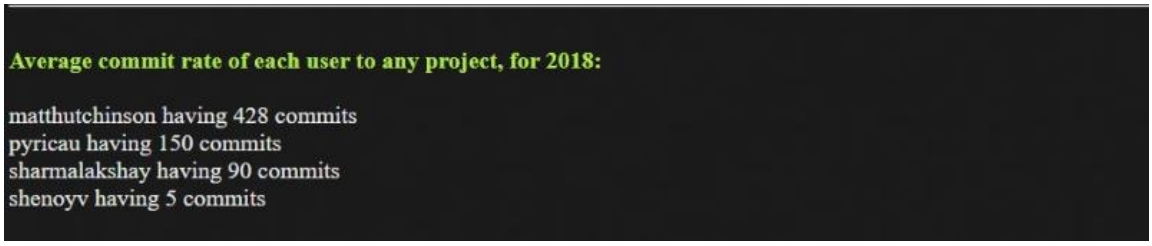
---

4. **The weekly commit rate of users (provide a weekly rank ordering) for the submitted project set, for 2018.**



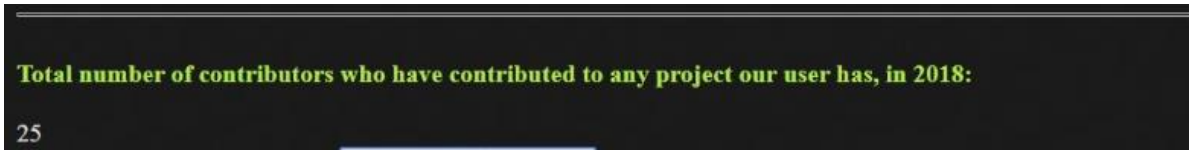
I load results from *https://api.github.com/repos/**user/repo**/stats/contributors* for each repository submitted. For each contributor who has committed, for each commit he has done, if the commit date is later than year 2018, I add plus one to the contributor.
I display the week of the year 2018 in increasing order, and which contributor has committed how many number of times in descending order.
Then I divide the total number of commits a user made by the total number of weeks he had worked on the project which gives me the commit rate. I sort it in descending order and display with the contributor name.

5. **The average commit rate of each user to any project, for 2018.**



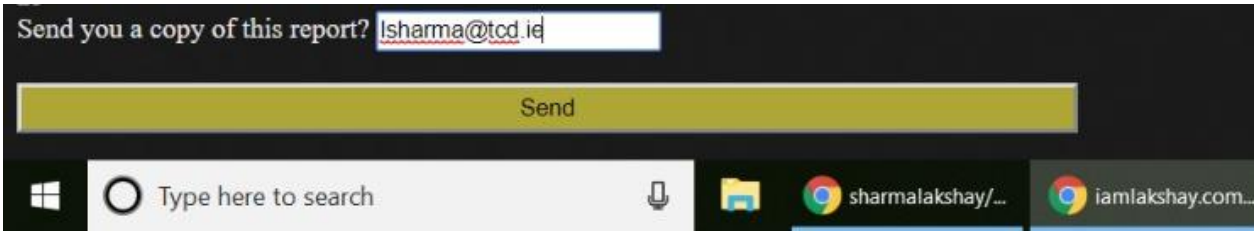
I load results from *https://api.github.com/search/commits?q=author:**user**+committer-date:>2018-01-01* for each user which gives me all the commits done by the user after 2018. I sort the total number of commits in descending order and display them along with the username.

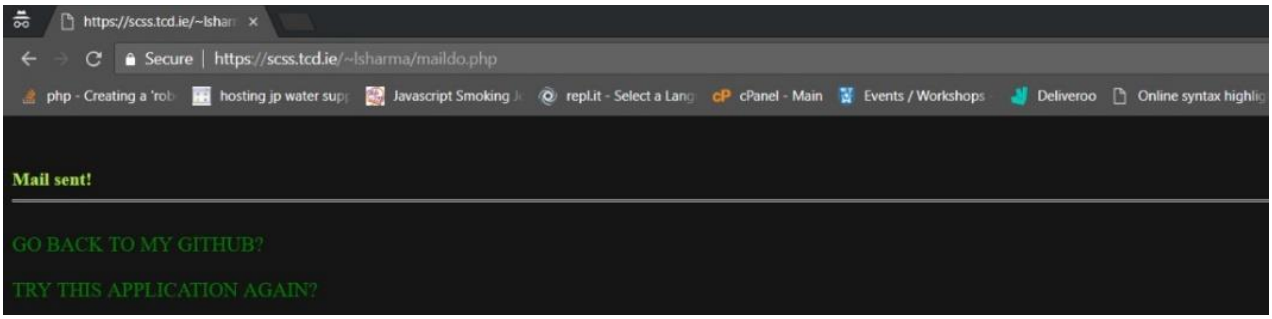
6. **The total number of collaborators in 2018 (ie. a count of other users who have contributed to any project that the user has contributed to).**

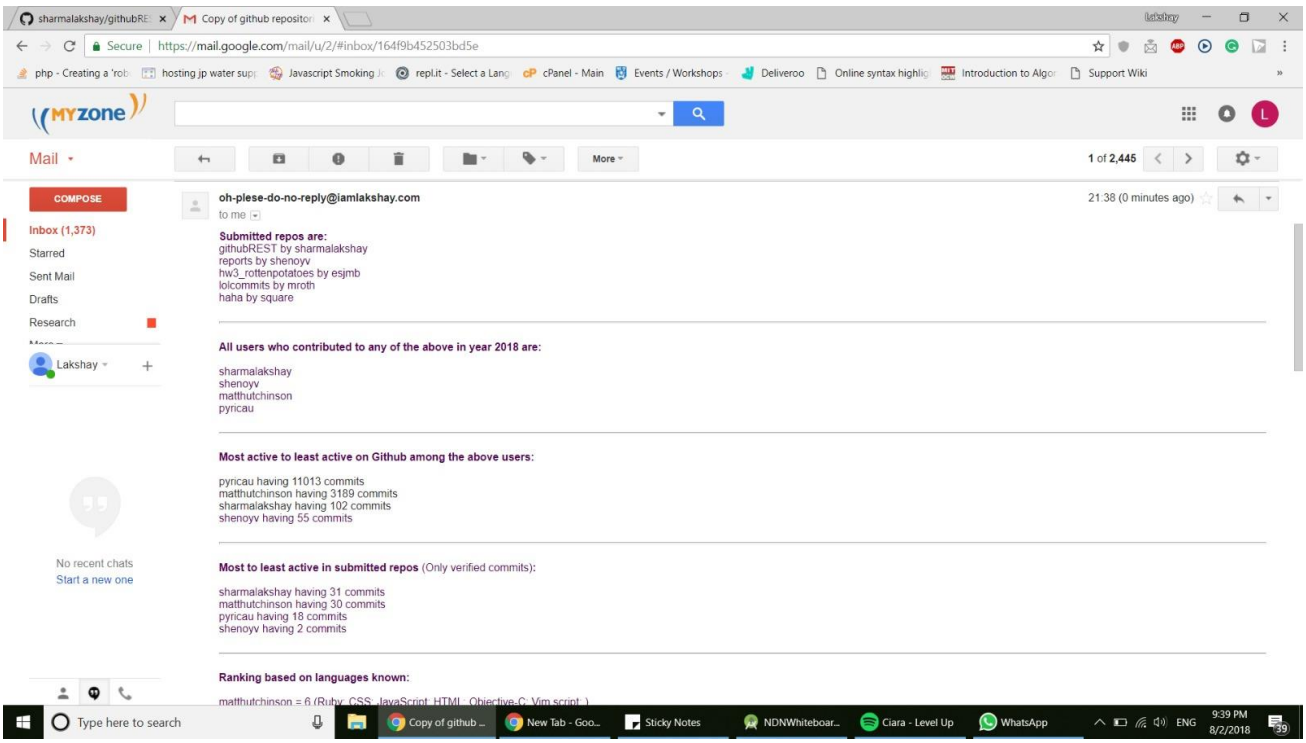


I load results from *https://api.github.com/search/repositories?q=user:**user**+pushed:>2018-01-01* for each user that gives me all the repositories that a user has committed to.
Then I load *https://api.github.com/repos/**repo**/contributors* for each repository that I get from the above. I then check for the new users as they appear on the list of contributors and add them into a new array.
I count the number of items in the array of unique users and display it on the screen.

Your rest service should email the results of calculation to the submitter once complete.







After all the calculations are done, user has an option to enter his/her email address to receive the report in his/her inbox.
The email is sent by using php mail function from a webpage hosted on *https://scss.tcd.ie/~lsharma/*
(I will set this functionality inactive after a while for security reasons!)

THANK YOU

---*---*----*-----*------*-------*--------*-------*------*-----*----*---*--*-*

If you have any queries, please write to me at

LSHARMA@TCD.IE