iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** VII **Month of publication:** July 2020

DOI: http://doi.org/10.22214/ijraset.2020.7061

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com

Automated Performance Indicator System for CI/CD & DevOps Developers of Software Industry using Python

Lodhi Sachin[1], Vishwakarma Shivam[2], Singh Divakar[3]
[1, 2]*Computer Science and Engineering Undergraduate;*
[3]*Head of Dept. Computer Science and Engineering, UIT, Barkatullah University*

Abstract: ***Github is the medium to collaborate on the projects and it is very important for the professional developers and the individuals who are working on the projects together. In more technological language it is termed as Web based VCS or Version Control System. During these tough times like COVID19 all the people are requested to stay at home and work from there only. In this case the individuals who are working on their own can do their jobs in old fashion and they are not going to face many issues as they are the sole creator and manager of their work. But this paper focuses on the people who are working in collaboration on some project. Before this pandemic people were working on the projects locally and there was no need of any additional platform as many organizations have their own framework/platform to collaborate and develop the software product but it is not possible now. While working locally, on company's platform, it is quite easy to track progress and the work done by individual. This paper emphasizes the use of Github for that purpose and proposes the methodology for keeping record the performance of the individual and the team.***

I. INTRODUCTION

Github[5] is the website which primarily focuses on the feature of providing platform to provide VCS[12] services or Version Control System services, using Git[6], and act as a platform for the developers of the team or individual to collaborate on the same project.

For keeping record of the project contributors it is required to check individual's profile and then record the data and every essential element which may be helpful to evaluate the profile of the user. This paper proposes the method to automating the jobs of generating reports, individual and aggregately, of the fed user in the program. The manual method to check the information and store that manually is too obsolete and the difficulty level in doing so in pretty high.

Nevertheless Web Scraping[10] can be more than suitable for these jobs. Web scraping is the practice of gathering data through any means other than a program interacting with an API (or, obviously, through a human using a web browser). (Ashiwal, Tandan, Tripathi & Miri, 2016).

The approach is to get (scrap) data of various fields of Github profile and put them inside the desired file i.e. word, text, odt, xlsx etc. so the profile of particular user can be analysed based on the values/parameters stored in those files. Further use of this data can be lead to providing ranking, generating results and considering entire profile as parameter of some other activity or process.

II. MOTIVATION

During this time when there are advisories of maintaining social distance all the time, getting the crucial work done can be tough. Every sector has been hit and the aftermaths can be very exacerbating for everyone.

Let us talk about the software industry. People here are working from home, individually and in collaboration. The teams/individuals/developers who are working and using Github sometime feel the need of the all the data from their Github profile and especially the coordinators and head of the groups who are leading the teams that is making the software or any application. The leaders of the group would need the data of the individual participant of the team to evaluate the profile and the progress of the project and the work done by them. The people working in DevOps[3] industry or the developers who are developing software products in CI/CD[14] aka Continuous Integration and Development can be considered as beneficiaries of this tool.

By automating the job of getting data from the internet can be very time saving and powerful measure to ensure the compliance of the project with the predefined timeline of completion.

III. PROPOSED METHODOLOGY

This paper suggests the methodology in the following manner :

A. *Validation of the Usernames*

1) The enumerated list of users is fed in CSV[13] format.
2) After that program checks the validity of the usernames by checking that if user's profile exists or not.
3) If user is not present on Github then an exception is thrown.
4) If user is present it is kept inside the file.
5) Finalized list of user is generated which contains only valid users.

B. *Getting Basic Details*

1) Then program automates the task of getting basic details from Github.com.
2) These basic details include primitive profile details like username, location, bio, work etc.
3) In addition to that it scrap the data which provides overview of the work user has done. For instance these details would include number of repositories[8].

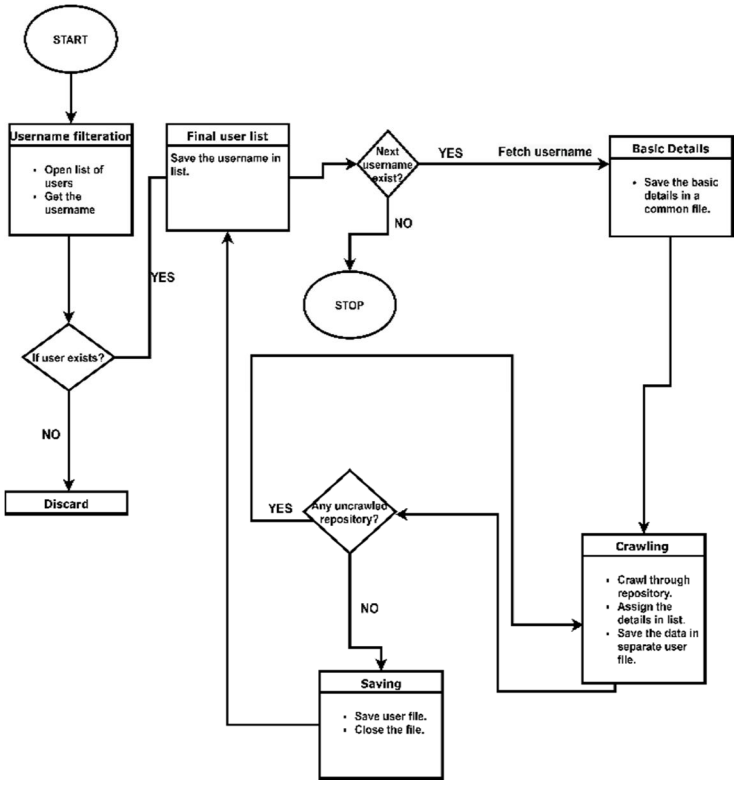
C. *Getting the in-depth Details*

1) After that the program Follows every repository present in the collection of repositories in the user profile.
2) For every repositories there would be more in-depth details which would be generated and would be stored inside the CSV file.
3) In-depth details include the number of commits[8], forks[8], branches[8], stars[8] etc.

D. *Storing scraped data inside CSV file*

1) After getting data from each repositoryit is stored in the CSV file for further use.
2) It can also be converted in xlsx format to process in MSOffice[15] products.

E. *The Flowchart Depicts The Execution Phase Of The Program And Various Operations*



Flowchart(I)

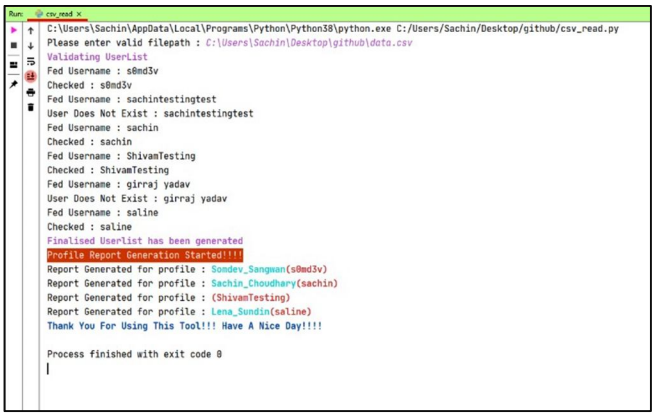
The program will check that if user exists or not. If user doesn't exist or in case of invalid username the program will throw an error and will indicate that username is wrong or user doesn't exist. If username is valid then program will go the homepage of the user and will scrap basic details about the user and then it will go through each repository which user has and then scrap every detail or required details from the user's repository and put them in the file. It will do so till last name in finalized username list.(See Flowchart(I).

IV. EXECUTION PHASE

The tool, in running phase, would ask for the filepath (Fig(I)) and the file must be in csv format and then it would scan for the invalid username and then filter them, generating the file with valid username. In case of invalid path, program will exit. There are two methods to feed file in program :

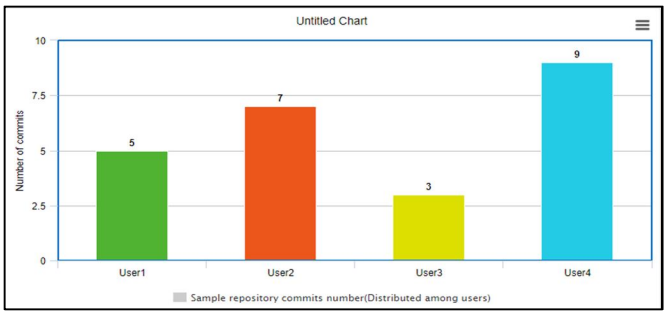
Keep the file in same directory or provide the full path of the file if it exists somewhere else. Alternates are available but providing full path of file is recommended.

Following are some depiction of the program in running phase and the output they produce in form of CSV file Fig(II). Graphs can also be generated Fig(I).



Fig(I)

Fig(II)



Fig(III)

V. TECHNOLOGIES USED

Following tools and technologies were employed in the project :

A. Python 3.7(Recommended) [16]

This is the backbone of every implementation made possible in this project. Every other component works under the subordination of Python 3.7.

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python was conceived in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system capable of collecting reference cycles. Python 3.0, released in 2008, was a major revision of the language that is not completely backward- compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language, i.e. Python 2.7.x, was officially discontinued on January 1, 2020 (first planned for 2015) after which security patches and other improvements will not be released for it. With Python 2's end-of-life, only Python 3.5.x and later are supported

B. Beautifulsoup[1][2][18]

Beautiful Soup is a Python library designed for quick turnaround projects like screen-scraping. Three features make it powerful:

Beautiful Soup provides a few simple methods and Pythonic idioms for navigating, searching, and modifying a parse tree: a toolkit for dissecting a document and extracting what you need. It doesn't take much code to write an application

Beautiful Soup automatically converts incoming documents to Unicode and outgoing documents to UTF-8. You don't have to think about encodings, unless the document doesn't specify an encoding and Beautiful Soup can't detect one. Then you just have to specify the original encoding.

Beautiful Soup sits on top of popular Python parsers like lxml and html5lib, allowing you to try out different parsing strategies or trade speed for flexibility.

Beautiful Soup parses anything you give it, and does the tree traversal stuff for you. You can tell it "Find all the links", or "Find all the links of class externalLink", or "Find all the links whose urls match "foo.com", or "Find the table heading that's got bold text, then give me that text."

Valuable data that was once locked up in poorly-designed websites is now within your reach. Projects that would have taken hours take only minutes with Beautiful Soup.

Using bs4's module .i.e. Beautifulsoup can be used to post process scraped data.

C. Request[9]

Request module is employed for sending requests to webserver and catch the response and do various post operations on that. Requests allows you to send HTTP/1.1 requests extremely easily. There's no need to manually add query strings to your URLs, or to form-encode your POST data. Keep-alive and HTTP connection pooling are 100% automatic.

D. CSV

The so-called CSV (Comma Separated Values) format is the most common import and export format for spreadsheets and databases. CSV format was used for many years prior to attempts to describe the format in a standardized way in RFC 4180. The lack of a well-defined standard means that subtle differences often exist in the data produced and consumed by different applications. These differences can make it annoying to process CSV files from multiple sources. Still, while the delimiters and quoting characters vary, the overall format is similar enough that it is possible to write a single module which can efficiently manipulate such data, hiding the details of reading and writing the data from the programmer.

The csv module implements classes to read and write tabular data in CSV format. It allows programmers to say, "write this data in the format preferred by Excel," or "read data from this file which was generated by Excel," without knowing the precise details of the CSV format used by Excel. Programmers can also describe the CSV formats understood by other applications or define their own special-purpose CSV formats.

The csv module's reader and writer objects read and write sequences. Programmers can also read and write data in dictionary form using the DictReader and DictWriter classes.

E. Termcolor[7]

Termcolor is a simple library which includes the functionality to print text on console screen in colourful format. *colored* function is employed in this proposed methodology.

F. Html Parser[4]

The html parser is a structured processing tool. It defines a class called HTML Parser, which is used to parse HTML files. It comes in handy for web crawling. It is used in this program to process fetched html pages by Beautifulsoup.

VI. CONCLUSION

By the proposed model or approach anyone can get report of the Github user in just one step and that can be employed in some other works. This program can work as scrapper specifically for Github.

VII. FUTURE WORK

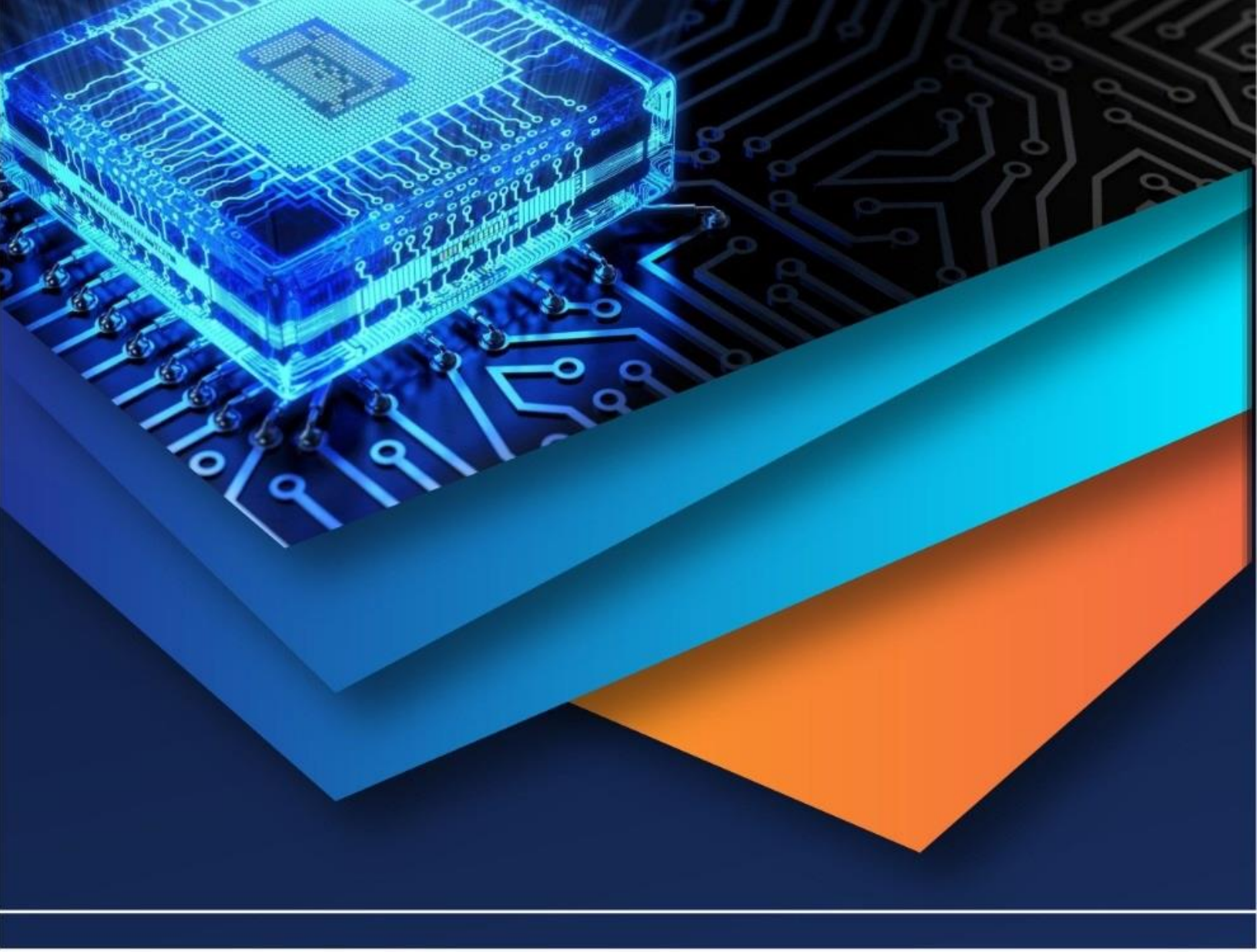
With the help of emerging technologies like Machine Learning the reports generated using this tool can be treated as the data set and a machine learning model can be trained for the prediction of the user rating i.e. the popularity /rating/skills of the respective users. In addition to that the files can also be downloaded from repositories for further inspection or using libraries like urllib2 downloaded codes can be analyzed using APIs like SonarSource[17] for the bugginess of code and MOSS[11] can be used to detect plagiarism of code. The data of repositories can be employed for further applications and the collected data can be used for any suitable purpose.

REFERENCES

[1] Ashiwal, P., Tandan, S.R., Tripathi, P., &Miri, R. (2016). Web Information Retrieval Using Python and BeautifulSoup.(*ISSN:* 2321-9653)
[2] D. PRATIBA, A. M.S., A. DUA, G. K. SHANBHAG, N. BHANDARI and U. SINGH, "Web Scraping And Data Acquisition Using Google Scholar," 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 2018, pp. 277-281, doi: 10.1109/CSITSS.2018.8768777
[3] https://aws.amazon.com/devops/what-is-devops (accessed on 10/02/2020)
[4] https://docs.python.org/3/library/html.parser.html (accessed on 13/04/2020)
[5] https://github.com (accessed on 5/11/2019)
[6] https://git-scm.com (accessed on 26/02/2020)
[7] https://kite.com/python/docs/termcolor.colored (accessed on 09/04/2020)
[8] https://linuxacademy.com/blog/linux/git-terms-explained/ (accessed on 27/03/2020)
[9] https://realpython.com/python-requests
[10] https://scrapinghub.com/what-is-web-scraping (accessed on 7/02/2020)

[11] https://theory.stanford.edu/~aiken/moss (accessed on 22/05/2020)
[12] https://www.atlassian.com/git/tutorials/what-is-version-control (accessed on 05/02/2020)
[13] https://www.howtogeek.com/348960/what-is-a-csv-file-and-how-do-i-open-it (accessed on 27/02/2020)
[14] https://www.infoworld.com/article/3271126/what-is-cicd-continuous-integration-and-continuous-delivery-explained.html(accessed on 17/02/2020)
[15] https://www.lifewire.com/microsoft-office-4156573 (accessed on 02/04/2020)
[16] https://www.python.org/downloads/release/python-377/ (accessed on 02/02/2020)
[17] https://www.sonarsource.com (accessed on 20/05/2020)
[18] Steven, Cibambo. (2019). Web Scraping Wikipedia using Python and BeautifulSoup. 10.13140/RG.2.2.34480.71685.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VII July 2020- Available at www.ijraset.com



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY