Stack Overflow Auto Search Tool

G.S.Wedpathak[1], Shweta Appasaheb Paraganve[2], Shreeya Narayan Mali[3]
[1]Assistant Professor, Dept of Computer Science and Engineering
[2, 3]Dept of Computer Science and Engineering
[1, 2] Sharad Institute Of Technology College Of Engineering, Yadrav(Ichalkaranji) , India

Abstract- *Stack Overflow could be a saviour to any or all the programmers and developers out there. a bit bug within the program and also the next step is to Google it and finish up on Stack Overflow page. A GUI application developed using Python & Tkinter Module which automatically searches for the error solutions for the chosen file on StackOverflow using Stack Exchange API. this is often only supported for the Python for now. the most purpose is to extract the desired data from Stack Overflow.*

Keywords- python, Tkinter, GUI application, error

I. INTRODUCTION

Stack Overflow may be a source of helpful information for software development related queries. Some years post its inauguration, Stack Overflow became one in every of the norms of software development and is now a fundamental component of software development culture. While coding, all folks get entangled in one or the opposite form of error fairly often, where the help of websites like Stack Overflow comes into picture. the overall pattern of our everyday software development life involves coding and testing the code. just in case we encounter errors, we hunt for those errors on Google, Stack Overflow, etc., fix them and continue this cycle till we get our code running the way we want. during this project we are visiting create a tool which will check our program for errors by executing it and on encountering any errors, it'll seek for it on Stack Overflow and open up some threads in applications programme tabs. the most purpose is to extract the specified data from Stack Overflow. We'll write a python script that may do a dry run for a few other program we are aiming for and check for errors present there. If it's no errors, it'll show us "No errors found". Otherwise, it'll open tabs of the Stack Overflow page targeting the errors we've therein particular code. tool which will execute a Python program and check for errors during the execution. If there are errors, the tool will explore for Stack Overflow threads that relate to the errors and open them as tabs in a very applications programme.

II. MOTIVATION

One of the foremost powerful attributes of Stack Overflow (SO) is that the accumulation of developers' knowledge over time. Community members have contributed over 18 million questions and 27 million answers. When a developer is stuck on a coding problem, they search through this vast trove of knowledge to work out if an answer to their particular conundrum has already been offered. Some use the interior SO search feature while others use search engines like Google or Bing, narrowing down the search to the stackoverflow.com domain. Python script, which automatically detects the error from code, search about it on Stack Overflow, and disclose the few tabs associated with our error that were previously been answered also.

III. SCOPE OF WORK

Create a wrapper Python script, which will run the test Python code, to test for errors. Simply print "No errors found", just in case there are not any syntactic or runtime errors. On encountering a mistake, extract the specified error type and also the error message. Then we'd like to create a REST API call to Stack Exchange API. From the JSON response obtained, we'd like to extract at the most N number of links to Stack Overflow threads and open the identical during a application.

IV. LITERATURE SURVEY

The Question Answering Mechanism on Stack Overflow Stack Overflow provides a platform for the asking and answering of questions. However, at the foremost one answer may possibly be accepted by the asker because the accepted answer to point that this particular answer is that the foremost suitable/correct one. Within the rest of the paper, we confer with a problem, its corresponding answers (i.e., both accepted and not-accepted answers) and each one the associated comments with these answers together as a matter thread. We seek advice from an answer (could either be accepted or not-accepted answers) and its comments as an answer thread. Users tag questions3 into well-defined categories. Tags capture the topics with which a problem is associated. Each question

can have at the foremost five tags and must have a minimum of 1 tag. Askers must specify the tags once they create a difficulty. Within the rest of the paper, we are saying that an answer is said to a particular tag if the answer belongs to a difficulty that's associated with that tag.

V. PROBLEM STATEMENT

The main problem with project is when programmer using interpreter or compiler. If programmer using compiler then it get too many errors at same time. For this we are using tkinter we are detecting digit and the by using stack we are going to search for error solution one by one.

VI. EXPERIMENTAL SETUP

1. **Modules used**

The subprocess module is employed to run new applications or programs through Python code by creating new processes. The subprocess module was created with the intention of replacing several methods available within the os module, which weren't considered to be very efficient.

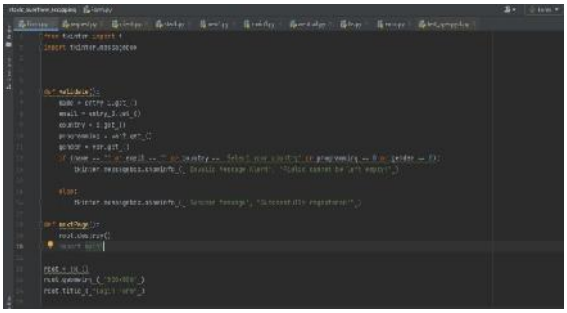
- The requests module to send HTTP requests using Python.
- The webbrowser module allows us to launch the net browser.

2. Approach

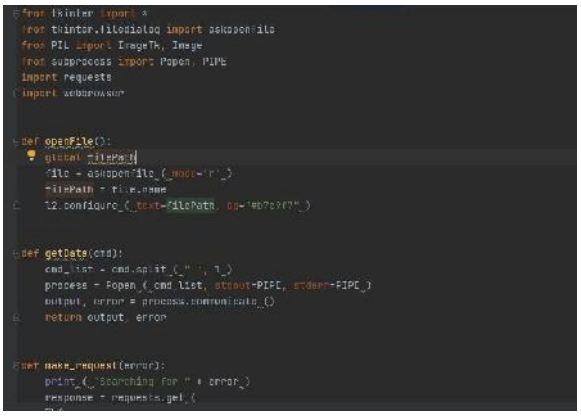
- **execute_return(cmd**): On the primary function, we are visiting write code to read and run python file, and store its output or error.
- **mak_req(error):** This function will make an HTTP request using Stack Overflow API and also the error we get from the first function and at last returns the JSON file.
- **get_urls(json_dict)**: This function takes the JSON from the 2nd function, and fetches and stores the URLs of these solutions which are marked as “answered” by StackOverflow.

VII. WORKING MODEL

1. Extracting the error message :



2. Using the Stack Exchange API’s search feature to generate requests



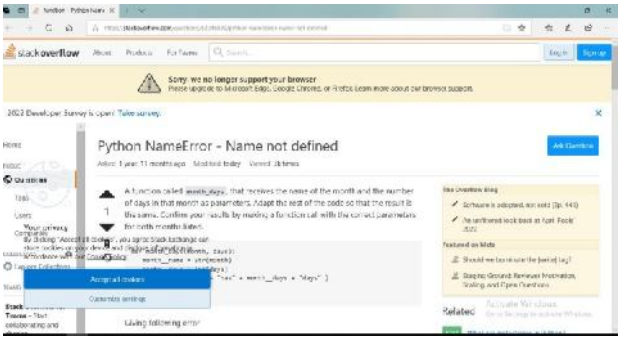
3. Use the web browser module to open the extracted links in a web browser

```
import webbrowser
def getlinks(rjson):
    url_list = []
    countlinks=0

    for i in rjson["items"]:
        if i["is_answered"]:
            url_list.append(i["link"])
        countlinks+=1
        if(countlinks==5 or countlinks==len(rjson["items"])):
            break

    for i in url_list:
        webbrowser.open(i)
```

VIII. RESULT



Errors :

IX. CONCLUSIONS

Stack overflow auto-search tool this will useful for coders , programmers . While programming every programmer get errors, for beginner or for intermidiate code it is not possible to find error quickly . this project definitly helpful for them. Stack overflow auto search tool help them to find errors quickly . Another one feature of this project is this will be optimize the time of programmers because they are going to code , run and find errors on same place they don't need to go anywhere for solution.

Question and answer websites similar as stack overflow are extensively used by programmers andexperiments , forming a large depository of precious , knowledge related to the software development , computing , and data wisdom diligence. Software inventors calculate on similar websites to acquire knowledge, break problems, seek particles of law for exercise, ameliorate their own law, and specialized generalities .

REFERENCES

[1] https://scholar.google.co.in/scholar?q=literature+review+of+stackoverflow&hl=en&as_sdt=0&as_vis=1&oi=scholart
[2] https://www.researchgate.net/publication/323072822_A_survey_on_mining_stack_overflow_question_and_answering_QA_community
[3] https://www.researchgate.net/publication/342986750_A_Systematic_Literature_Review_on_Using_Machine_Learning_Algorithms_for_Software_Requirements_Identifica
[4] https://hcis-journal.springeropen.com/articles/10.1186/s13673-017-0091-8
[5] https://www.researchgate.net/publication/323072822_A_survey_on_mining_stack_overflow_question_and_answering_QA_community
[6] https://api.stackexchange.com/
[7] https://hcis-journal.springeropen.com/articles/10.1186/s13673-017-0091-8
[8] https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0253010