“Intelligent Codemate”

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

by

Roll No	Name of Project Member
13	Sanyam Bharani
57	Suraj Pawar
62	Vijay Sahani

under the guidance of

Supervisor (s): **Mrs. Asma Parveen I. Siddavatam**

Department of Information Technology

Vivekanand Education Society’s Institute of Technology

2018-19

Vivekanand Education Society's

Institute of Technology

(Affiliated to University of Mumbai, Approved by AICTE & Recognized by Govt. of Maharashtra)

Department of Information Technology

CERTIFICATE

This is to certify that **Mr. Suraj Pawar, Mr. Sanyam Bharani, and Mr. Vijay Sahani** of Fourth Year Information Technology studying under the University of Mumbai have satisfactorily presented the project entitled **"Intelligent Codemate"** as a part of the PROJECT-I for Semester-VII under the guidance of **Mrs. Asma Parveen I. Siddavatam** in the Academic year 2018-2019.

Date: 12-11-2018

(Name and sign)
External Supervisor

Dr. Mrs. Shalu Chopra
Head of Department

Mrs. Asma Parveen I. Siddavatam
Supervisor/Guide

Abstract

The aim of the project is to develop an application which can help and improve error solving process ie., Debugging program. This application can guide the student to debug a program written in any languages like python, c, c++, and java.

The application program has a feature to extract error given by user's program and will scrape Stack-Overflow website and give suggestions to the user. It will show a detailed explanation of every possible solution provided by Stack-Overflow website along with question upvotes, answer acceptance status, and question total views. An application will be trained and tested in such a way that it will give an optimum solution to the related query. Then, a user has to click on question title to see the solution.

Keywords: *Terminal-application, Web-scraping, Artificial-Intelligence*

Table of Contents

Chapter 1

Introduction

Every Computer Science aspirant needs to learn to programme in any language like C, C++, Java, and Python etc. One thing common in every language is a syntax error. Every novice programmer has to deal with it, sometime programmer end up spending more time debugging than coding. It is a quite a challenging task for computer science teachers to teach error handling. Since the inception of first ever programming language Fortran, debugging problem persists despite the presence of abundant information available on the internet. Although tools used for programming changed significantly from decades earlier one. Hence, a simple, as well as an efficient debugging application, is today's need. That's why we came up with this idea to develop a terminal based debugging application which will search over the internet for a possible solution. Then these solutions will be ranked according to various parameters that will help in reducing redundancy while displaying solutions.

1.1 Problem Statement

One of the major problem faced by the programmers is debugging which is very time consuming and require patience. Time and patience both things naive programmers don’t have. In today's world, every problem has a solution but sometimes this application becomes very lagging and feeble. So, our application will provide terminal based easy to use and effective solution for debugging.

1.2 Objectives

- Improves Debugging Speed

  The users need not switch the tab for resolving the error. Solution to the error is displayed on the terminal itself this results in improving the program debugging speed.

- Error extraction

  An error extraction is done by the application so, no need to copy error from the terminal.

- Optimal Solution

  An optimal solution is decided by the maximum number of upvotes and most favourite answer chosen by the users on that platform.

Chapter 2

2.Literature Survey

2.1 Techniques studied

- *Web scraping*: It is used for extracting data from the website. web scraping is a web page involves fetching it and extracting from it. Fetching is the downloading of a page (which a browser does when you view the page). Therefore, web crawling is the main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place. The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet, and so on. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else.

- *Bash shell*: It is terminal on which the program has been done and solve the error.

- *Regular expression:* It is used to mean the specific, standard textual syntax for representing patterns for matching the text. Each character in a regular expression is either a metacharacter, having a special meaning, or a regular character that has a literal meaning. For example, in the

regex a., *a* is a literal character which matches just 'a', while '.' is a metacharacter that matches every character except a newline. Therefore, this regex matches. A very simple case of a regular expression in this syntax is to locate a word spelled two different ways in a text editor.

2.2 Papers

2.2.1 paper 1 -"A Framework for Programming Process Measurement and Compiling Error Interpretation for Novice Programmers" [1]

This paper proposes a framework to support programming education intended for novice programmers at universities. Programming processes can be measured and a degree of understanding can be estimated by retaining the editing history of source codes. Furthermore, the implementation of the proposed framework, which resulted in the early identification of problems faced by students are presented and evaluated.

2.2.2 paper 2- "Compiler-directed instruction duplication for soft error detection [2]

The compiler determines the instruction schedule by balancing the permissible performance degradation with the required degree of duplication. The experimental results show that algorithms allow the designer to perform tradeoff analysis between performance and reliability.

2.2.3 paper 3-"Effective testing and debugging methods and its supporting system with program deltas" [3]

In the maintenance phase of software development, it is necessary to check all features still perform correctly after some changes have been applied to existing software. However, it is not easy to debug the software when a defect is found in those features which have not changed during the modification, even using a regression test. Existing approaches employ program deltas to specify defects.

2.2.4 paper 4- "An approach to automated program testing and debugging" [4]

The most annoying aspect of software development is debugging. We don't mind the kind of bugs that yield a few minutes of inspection. The bugs we hate are the ones that show up only after hours of successful operation, under unusual circumstances, or whose stack traces lead to dead ends.

Chapter 3

3. Requirement and Analysis

3.1 Functional Requirement

- Python library requirements:

    - ❖ Urwid- Urwid is a console user interface library for Python. It includes many features useful for text console application developers including Applications resize quickly and smoothly, Automatic, programmable text alignment, wrapping, and Simple markup for setting text attributes within blocks of text.

    - ❖ Regex- This module provides regular expression matching operations similar to those found in Perl. Both patterns and strings to be searched can be Unicode strings as well as 8-bit strings.

    - ❖ BeautifulSoup4- Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching and modifying the parse tree. It commonly saves programmers hours or days of work.

    - ❖ Requests- Requests allows you to send organic, grass-fed HTTP/1.1 requests, without the need for manual labor. There's no need to manually add query strings to your URLs or to form-encode your POST data.

- Optimal solution suggestion: Our Debugging Engine will search all over the internet and solution will be displayed on the terminal. It gives the result based on the parameter such as the maximum number of upvotes

and number of views.

3.2 Non Functional Requirement

- Response time: 1-2 seconds
- Memory requirement: depends on machine.
- Usability requirement: terminal will acts as a user interface, as many developers and programmers are used to with it so, no special training is not required.

3.3 Constraints

- The Internet is a must.
- The solution provided will be from the World Wide Web only.
- All package are required for smooth execution.

3.2 Software Requirement

- operating system: Windows/Linux
- Terminal or Bash Shell
- Python 3.0+
- Required libraries

3.3 Hardware Requirement

- Computer with Internet connection
- Browser(optional)

Chapter 4

4.Design

4.1 Architecture Design

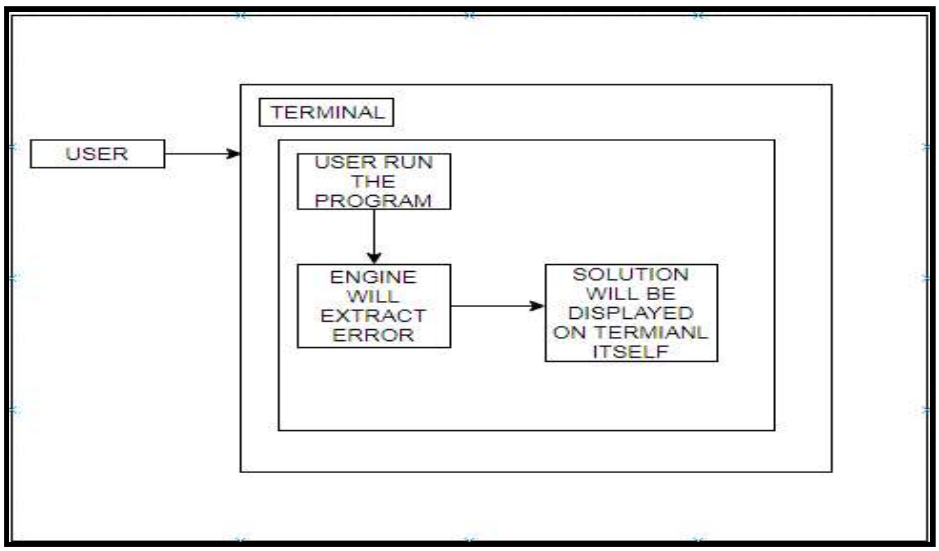
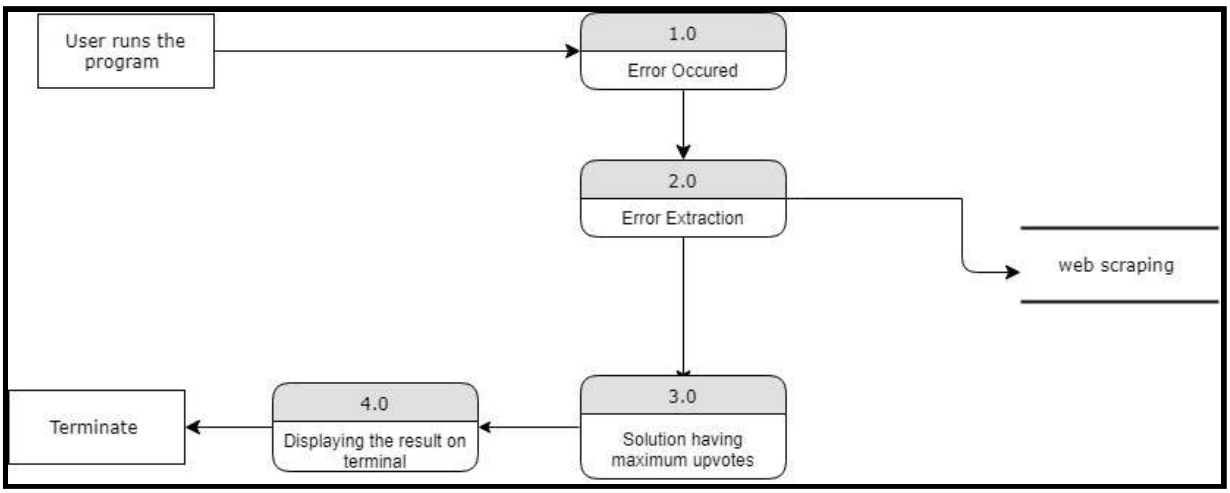
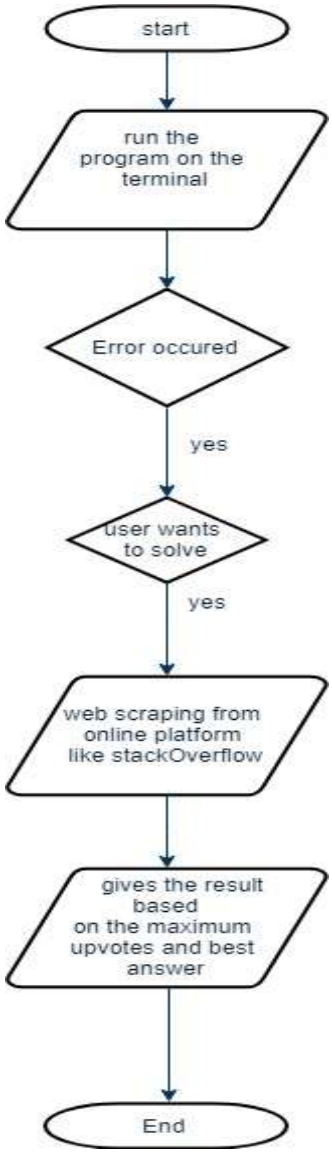


Fig. 1 Block Diagram

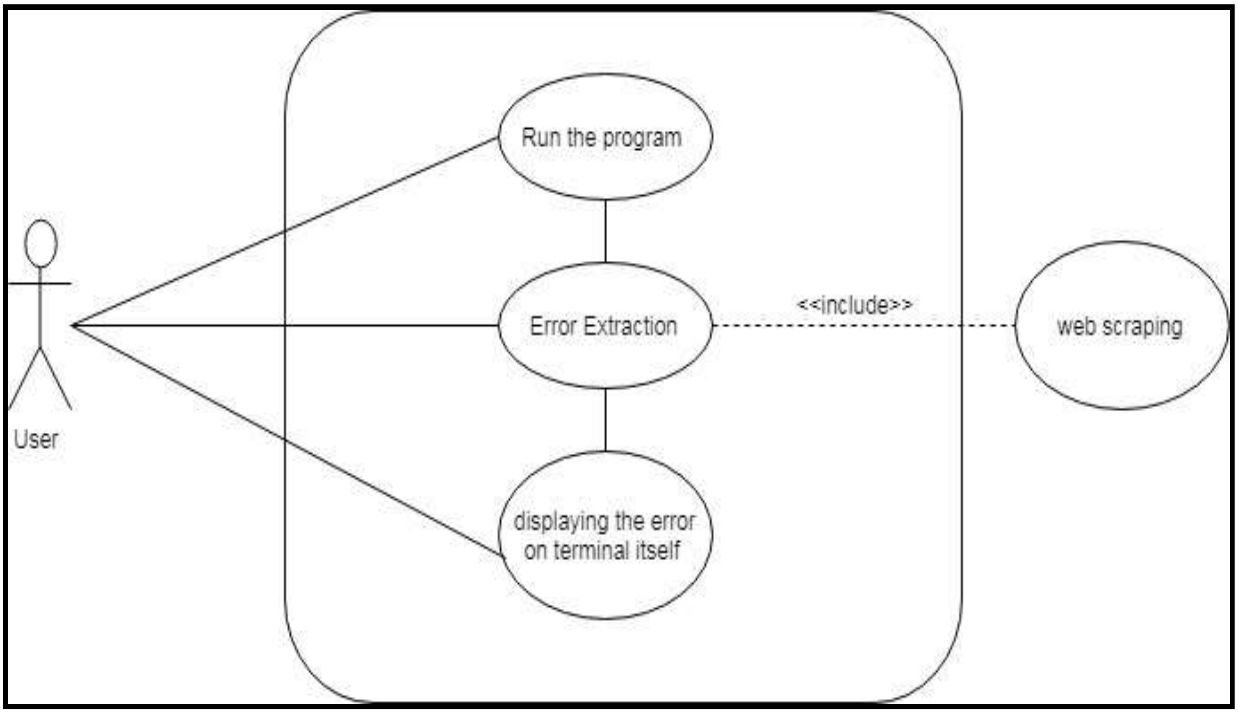
4.2.1 System Design



4.2.2 Flowchart



4.2.3 Use case


Run the program
Error Extraction
<<include>>
web scraping
User
displaying the error
on terminal itself

Chapter 5

5.Results and Conclusion

5.1 Expected Results

SCENARIO	EXPECTED RESULTS
To verify whether the solution obtained for a given query is correct.	Mapping is done properly.
Verifying whether the best optimal solution is available.	The solution is generated on basis of maximum upvotes.
Verifying whether the error can be handled by the user or not.	The user has to know about the error to solve the problem.

5.2 Conclusion

The intention of developing a terminal based application is to improve the debugging speed of the program by making it easier to search for a solution over the internet without opening web-browser which helps in increasing productivity and decreases chances of distractions.

Chapter 6

6. References

IEEE Papers

[1] Haruaki Tamada, Akihiro Ogino, Hirotada Ueda: “A Framework for Programming Process Measurement and Compiling Error Interpretation for Novice Programmers”

[2] Kenji Fujiwara, Kyohei Fushida, Haruaki Tamada, Hiroshi Igaki, Norihiro Yoshida, "Why Novice Programmers Fall into a Pitfall?: Coding Pattern Analysis in Programming Exercise"

[3] T. Dogsa, I. Rozman, "CAMOTE - Computer Aided Module Testing and Design Environment"

[4] D. M. Cohen, S. R. Dalal, M. L. Fredman, G. C. Patton, "The aetg system: An approach to testing based on a combinatorial design"

7. Acknowledgment

We would like to express our sincere gratitude to our project mentor, **Mrs. Asma Parveen I. Siddavatam,** for the direction guidance, help, support, and valuable advice. A special thanks to the department for providing the support and resources needed to our team for working so hard and facing all the hard times together.

Sanyam Bharani Suraj Pawar Vijay Sahani