

DEVELOPING A TOOL FOR VALIDATING A WEBSITE

A PROJECT REPORT

Submitted by

ABHISHEK FRANCIS KUJUR

In partial fulfilment for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

Under the guidance of

Dr. Vinod Pathari

(Assistant Professor, Dept. of Computer Science and Engineering)



Department of Computer Science and Engineering,

National Institute of Technology, Calicut

NITC Campus PO, Calicut

Kerala, India 673601

JUNE 2015

DECLARATION

“I, hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or the other institute of higher learning, except where due acknowledgement has been made in the text.”

Place: NIT Calicut

Signature:

Date: June, 2015

Name: ABHISHEK FRANCIS KUJUR

Reg. No.: M120378CA

CERTIFICATE

*This is to certify that the project report entitled: **Developing a Tool For Validating a Website** submitted by **Mr. ABHISHEK FRANCIS KUJUR** (Roll No.: **M120378CA**) to the National Institute of Technology Calicut towards partial fulfilment of the requirements for the award of the Degree of **MASTER OF COMPUTER APPLICATIONS** is a bona fide record of the work carried out by him under our supervision and guidance.*

(Dr. Vinod Pathari)

Place:

Date:

Signature of Head of the Department

(Office seal)

Table of Contents

1. Introduction	6
1.1 Problem Definition	6
1.2 Motivation.....	6
1.3 Proposed System.....	6
2. Theoretical Aspect	7
2.1 Product Description	7
2.1.1 User Interface.....	7
2.1.2 Hardware Interfaces	7
2.1.3 Software Details	7
2.1.4 Communication Interface	7
3. System Design	8
3.1 Class Diagram:.....	8
3.2 Activity Diagram:.....	9
3.3 Use Case Diagram:	10
3.4 User Characteristics	13
3.5 Non-functional requirements	13
3.5.1 Performance Requirements.....	13
3.5.2 Constraints	13
4. Implementation	14
4.1 Module.....	14
4.2 Software - GUI.....	15
4.2.1 Screenshot	16
5. Conclusion.....	19
6. References	20

Abstract

We develop an application tool to extract words in a website. This application takes the URL of that website to crawl every page of a website and do spell check on them. It extracts the content of the website and checks for spelling. It identifies the spelling mistake in that website. Display the mistake in the content of the website. The output shows the corresponding links in which the mistakes exist allowing the administrator to correct them.

1. Introduction

1.1 Problem Definition

This software helps to rectify spelling mistakes in the data. It uses its engine to process and scan all pages. Given a URL, it will scan every page linked to that URL.

The crawler begins with an initial URL to scan a website called seed. The first task of the crawler is to identify the hyperlinks on the webpage and use those links to further crawl called crawl frontier. If a website contains large volume of information it may take some time for the crawler to download. So the crawler has to select what information it wants to send down

Following process is used by the software. Web crawler fetches the page. Download and parse the page to retrieve information. For every link retrieved, repeat the task. The downloaded pages will be checked with a dictionary for error. Suggestions for word replacement will be given by the software.

1.2 Motivation

This problem is to be done because it will allow validation of a website. The software tool will benefit our own College website. It will keep it validated from time to time. We want proper information to be conveyed to the user of the website. It will ease down the workload of the site administrator by reducing the time to check/validate the website.

1.3 Proposed System

The tool developed is novel in various aspects. It uses an initial link to extract the URLs. It uses this URL to get information from webpage. Additional functionality added to the tool is Spell checking. It extracts visible (readable) contents from a page, checks for incorrect words. It has an added feature of word suggestion which allows user to select the correct word. It is a combination of all the elements that is not present in the existing system. All these features make a great tool for website validation.

2. Theoretical Aspect

2.1 Product Description

2.1.1 User Interface

The software offers a completely user friendly interface running on any screen resolution except mobile device .The user interface will be GUI based one requiring input from keyboard.

The Main Interface includes:

Show error - Display the word which has to be corrected.

URL box- It will allow user to input a link to specifically check that webpage for validation.

2.1.2 Hardware Interfaces

None.

2.1.3 Software Details

PYTHON – Python is a high level programming language. Its design focuses on code readability. Programmers are able to express the concept in few lines which would be not possible in other language.

It supports multiple paradigms for programming. It includes object oriented, imperative and procedural styles. Programmers bargain on features like automatic memory management, dynamic type system and a standard library.

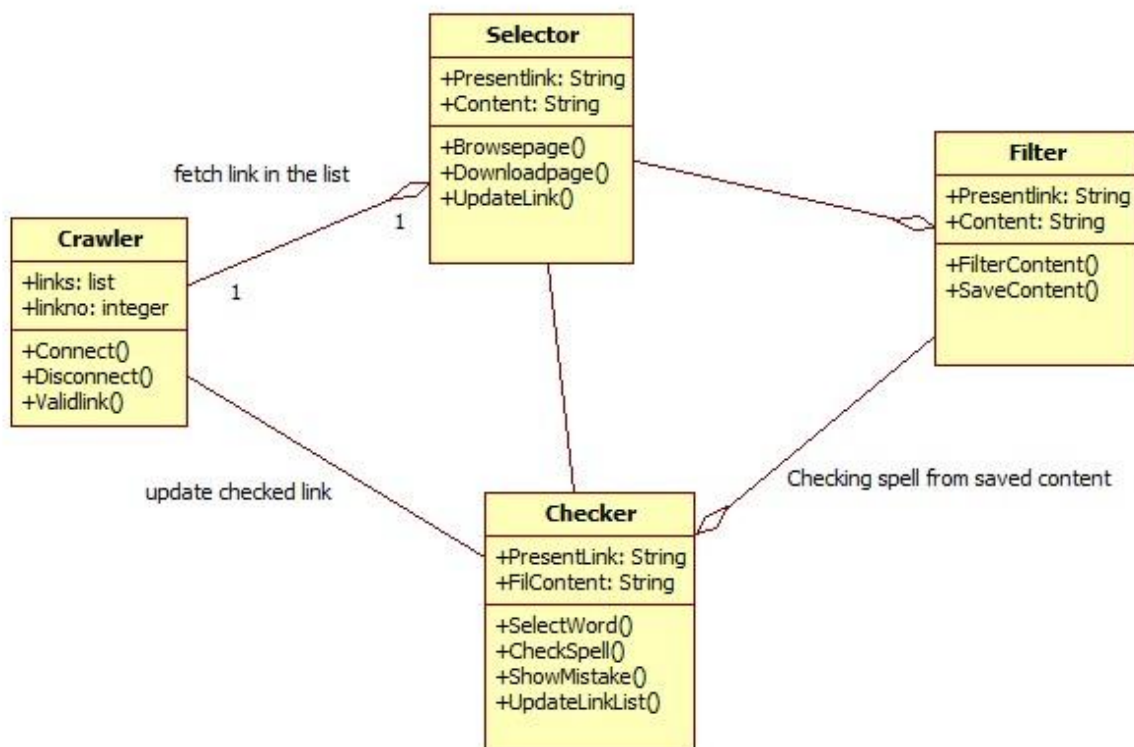
IDLE (Python)- IDLE stands for Integrated Development Environment. It provides an environment for python coding .Tkinter GUI toolkit and Python is used to develop it.

2.1.4 Communication Interface

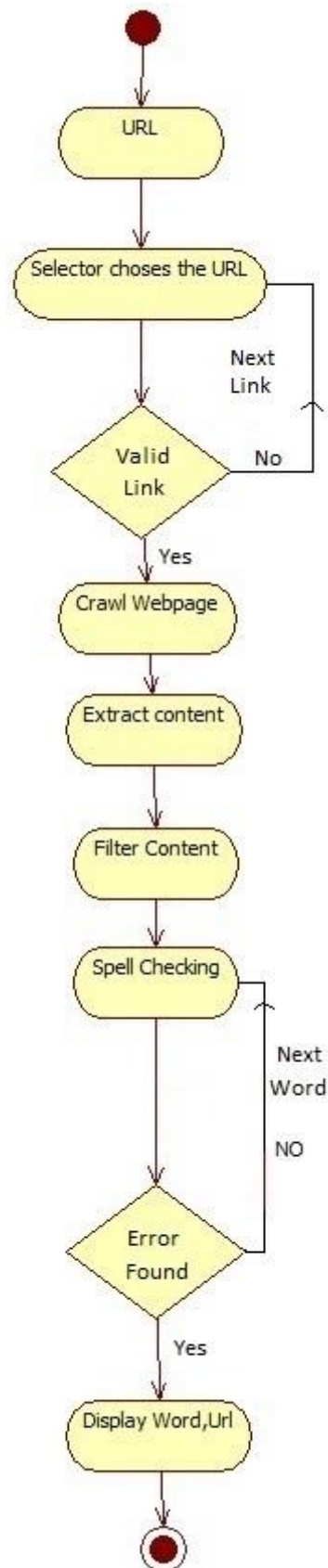
The communication is managed by the underlying operating system with the help of standard protocols like the TCP. The software use Beautiful Soup package from python language to parse the website and extract the content or to download the webpage.

3. System Design

3.1 Class Diagram:



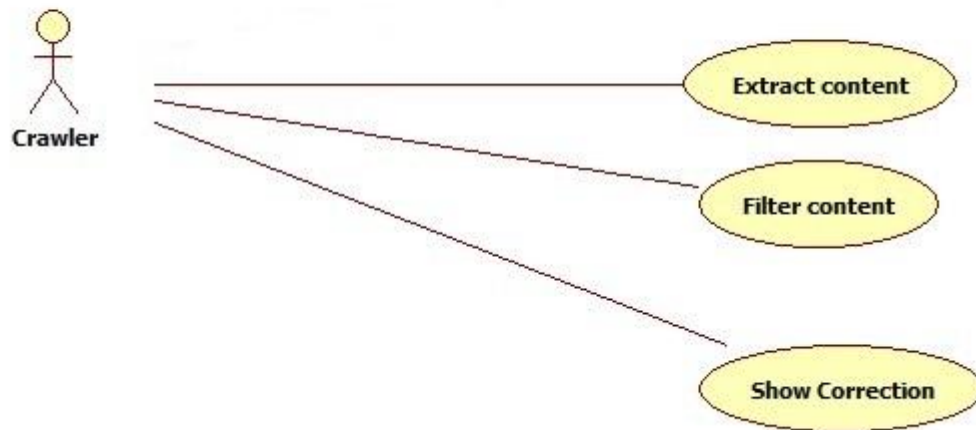
3.2 Activity Diagram:



3.3 Use Case Diagram:

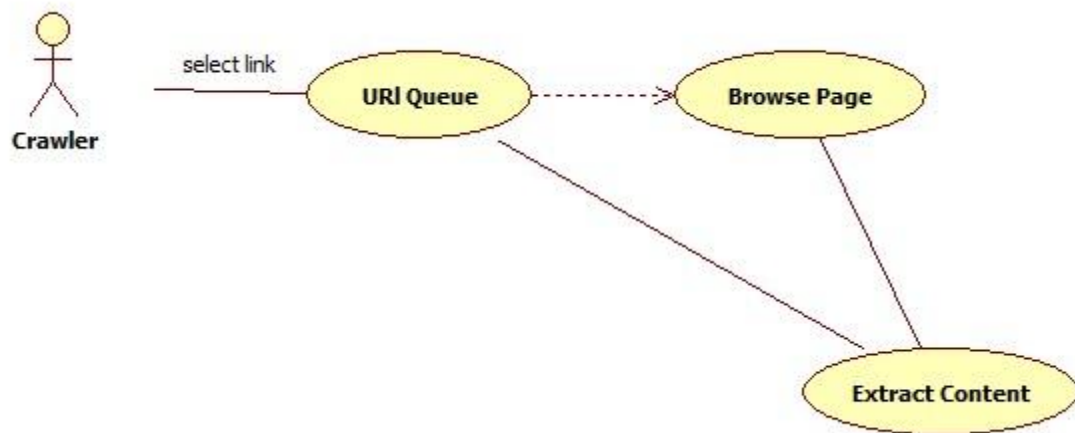
This section outlines the use cases for each of the active actors separately.

Main User Use Cases:



Use case: Extraction

Diagram:



Brief Description:

Initial Steps:

1. The Selected URL is used by the software to browse.
2. It goes the link of the website.
3. Download the webpage.

Use case: Filter the content

Diagram:



Brief Description: The crawler downloads the content of the webpage to filter.

Initial Steps:

1. Download the web page.
2. Pass through the filter.
3. Filtered content is separated.

Use case: Spelling check

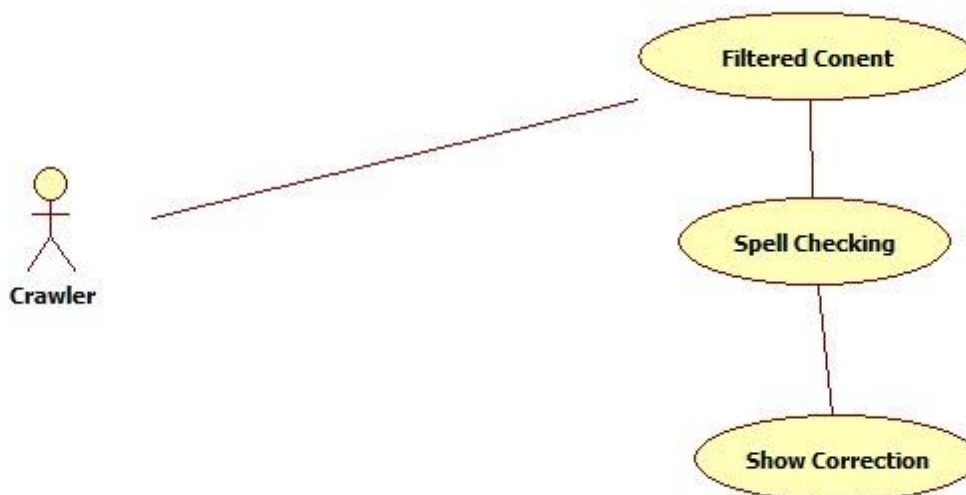
Diagram:

Brief Description: The extracted content is passes through a checker.

Initial Steps:

1. Pass the extracted content through spell checker.
2. Checks for uncorrected spelling.

Use case: Show Correction

Diagram:

Brief Description: After passing the content through the checker it shows where the correction is to be made.

Initial Steps:

1. Uncorrected word found.
2. Display the uncorrected word corresponding with URL.

3.4 User Characteristics

Administrator: Should be the person who is looking after the website.

3.5 Non-functional requirements**3.5.1 Performance Requirements**

Capacity: The system uses one user from where the tool is installed.

Network Connection: Whenever the software is used it should be connected to the internet.

Resource Usage: It will use maximum 100mb.

3.5.2 Constraints

One of the main constraints of the Validation of website for typos is the time to parse all the pages of the website. Accessing all the webpage, filtering and checking the spelling error takes time.

Another constraint is bandwidth speed.

4. Implementation

4.1 Module

Crawler module is the heart of this application which performs several vital processes like use of list of URL to download the webpage and extract content.

In implementing the web crawler, several python packages are used for extracting and manipulating the web pages. The list of python packages which are used in this application are as follows:

Beautiful Soup

Beautiful Soup is a library used in python to fetch data from Xml and html files. It provides a correct way to navigate, search and modify parse tree.

Three Features:

It provides simple method to search navigate and modify parse tree.

It converts incoming data to Unicode and outgoing data to UTF-8.

Beautiful Soup is very flexible by allowing us to try out different parsing techniques or trade speed for workability.

Requests

Request allows us to integrate with the web service.

Urllib

It helps to open URL using defined function and classes.

Re

Re library is used for regular expression. It is used to match strings.

4.2 Software - GUI

The screenshot shows a window titled "Form" with a pink border. It contains four input sections, each with a text label, an input field, and a button. Lines connect the buttons to their descriptions on the right:

- ENTER URL** (text) | (field) | **Get url** (button) → ENTER THE URL TO GET THE URLS
- DEPTH** (text) | (field) | **Get url (Depth)** (button) → URLS TILL A CERTAIN DEPTH
- CHECK SPELLING ERROR** (text) | **Spell Check** (button) → CHECK SPELLING FOR THE URL EXTRACTED
- SPELL CHECK A PAGE** (text) | (field) | **Page Check** (button) → USED TO CHECK THE CONTENT FOR A SINGLE PAGE

GET URL BUTTON - It extracts the URLs of the website.

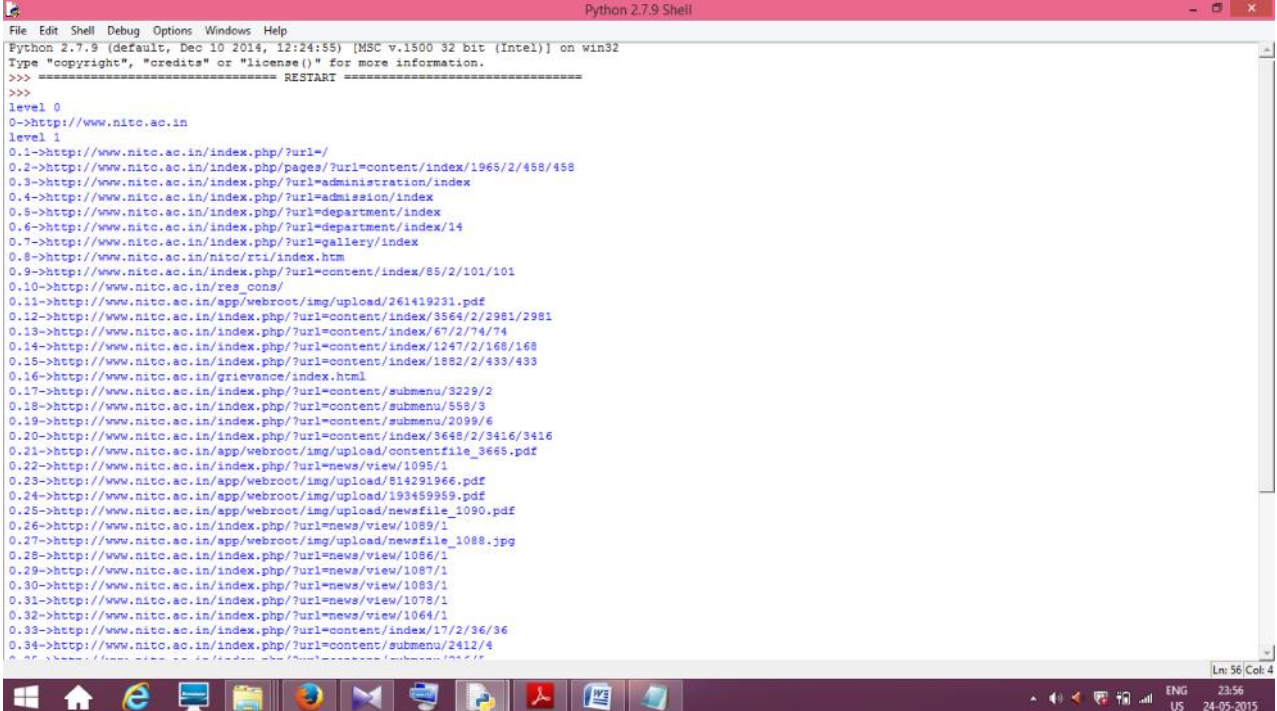
GET URL (DEPTH) BUTTON - It extracts URL of the website to the depth which is taken as input.

CHECK SPELLING ERROR BUTTON- It shows the spelling error that is in the webpage which was previously extracted.

PAGE CHECK BUTTON- It checks a single page for spelling error.

4.2.1 Screenshot

Extraction:



```
Python 2.7.9 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
level 0
0->http://www.nitc.ac.in
level 1
0.1->http://www.nitc.ac.in/index.php?url=/
0.2->http://www.nitc.ac.in/index.php/pages/?url=content/index/1965/2/458/458
0.3->http://www.nitc.ac.in/index.php?url=administration/index
0.4->http://www.nitc.ac.in/index.php?url=admission/index
0.5->http://www.nitc.ac.in/index.php?url=department/index
0.6->http://www.nitc.ac.in/index.php?url=department/index/14
0.7->http://www.nitc.ac.in/index.php?url=gallery/index
0.8->http://www.nitc.ac.in/nitc/rsl/index.htm
0.9->http://www.nitc.ac.in/index.php?url=content/index/85/2/101/101
0.10->http://www.nitc.ac.in/res/cons/
0.11->http://www.nitc.ac.in/app/webroot/img/upload/261419231.pdf
0.12->http://www.nitc.ac.in/index.php?url=content/index/3564/2/2981/2981
0.13->http://www.nitc.ac.in/index.php?url=content/index/67/2/74/74
0.14->http://www.nitc.ac.in/index.php?url=content/index/1247/2/169/169
0.15->http://www.nitc.ac.in/index.php?url=content/index/1882/2/433/433
0.16->http://www.nitc.ac.in/grievance/index.html
0.17->http://www.nitc.ac.in/index.php?url=content/submenu/3229/2
0.18->http://www.nitc.ac.in/index.php?url=content/submenu/558/3
0.19->http://www.nitc.ac.in/index.php?url=content/submenu/2099/6
0.20->http://www.nitc.ac.in/index.php?url=content/index/3648/2/3416/3416
0.21->http://www.nitc.ac.in/app/webroot/img/upload/contentfile_3665.pdf
0.22->http://www.nitc.ac.in/index.php?url=news/view/1095/1
0.23->http://www.nitc.ac.in/app/webroot/img/upload/814291966.pdf
0.24->http://www.nitc.ac.in/app/webroot/img/upload/193459959.pdf
0.25->http://www.nitc.ac.in/app/webroot/img/upload/newsfile_1090.pdf
0.26->http://www.nitc.ac.in/index.php?url=news/view/1089/1
0.27->http://www.nitc.ac.in/app/webroot/img/upload/newsfile_1088.jpg
0.28->http://www.nitc.ac.in/index.php?url=news/view/1086/1
0.29->http://www.nitc.ac.in/index.php?url=news/view/1087/1
0.30->http://www.nitc.ac.in/index.php?url=news/view/1083/1
0.31->http://www.nitc.ac.in/index.php?url=news/view/1078/1
0.32->http://www.nitc.ac.in/index.php?url=news/view/1064/1
0.33->http://www.nitc.ac.in/index.php?url=content/index/17/2/36/36
0.34->http://www.nitc.ac.in/index.php?url=content/submenu/2412/4
```

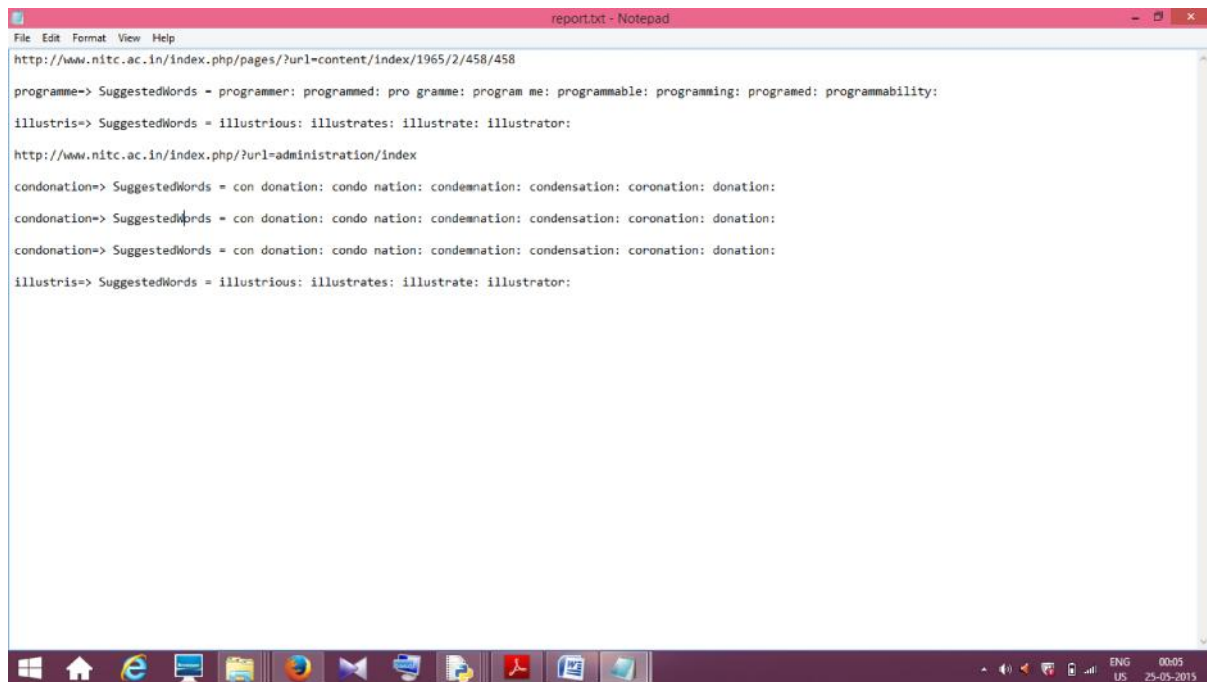

Spell Checking:

```

Python 2.7.9 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
http://www.nitc.ac.in/index.php?url=/
Words Found => 330
programme ['programmer', 'programmed', 'pro gramme', 'program me', 'programmable', 'programming', 'programed', 'programmability']
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php/pages?url=content/index/1965/2/458/458
Words Found => 309
condonation ['con donation', 'condo nation', 'condemnation', 'condensation', 'coronation', 'donation']
condonation ['con donation', 'condo nation', 'condemnation', 'condensation', 'coronation', 'donation']
condonation ['con donation', 'condo nation', 'condemnation', 'condensation', 'coronation', 'donation']
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php?url=administration/index
Words Found => 369
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php?url=admission/index
Words Found => 163
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php?url=department/index
Words Found => 203
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php?url=department/index/14
Words Found => 269
atNational ['at National', 'subnational', 'nationality', 'nationalism', 'nationalist']
anInstitute ['an Institute', 'instituter', 'institutes', 'institute', 'constitutes']
askiitansranking ['multitasking', 'frankincense', 'ranking', 'crankiness']
etcofor ['etc for', 'etcher', 'etcoeters', 'effort', 'etching']
sureah ['surest', 'sure sh', 'sureness', 'surety', 'surely', 'surer']
nitc ['nit', 'nits', 'nit c']
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/index.php?url=gallery/index
Words Found => 44
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/nitc/rti/index.htm
Words Found => 60
http://www.nitc.ac.in/index.php?url=content/index/85/2/101/101
Words Found => 223
illustris ['illustrious', 'illustrates', 'illustrate', 'illustrator']
http://www.nitc.ac.in/res_cons/

```

Report Generated:



```
report.txt - Notepad
File Edit Format View Help
http://www.nitc.ac.in/index.php/pages/?url=content/index/1965/2/458/458
programme=> SuggestedWords = programmer: programmed: pro gramme: program me: programmable: programming: programmed: programmability:
illustris=> SuggestedWords = illustrious: illustrates: illustrate: illustrator:
http://www.nitc.ac.in/index.php?url=administration/index
condonation=> SuggestedWords = con donation: condo nation: condemnation: condensation: coronation: donation:
condonation=> SuggestedWords = con donation: condo nation: condemnation: condensation: coronation: donation:
condonation=> SuggestedWords = con donation: condo nation: condemnation: condensation: coronation: donation:
illustris=> SuggestedWords = illustrious: illustrates: illustrate: illustrator:
```

5. Conclusion

Work done:

- 1) Extraction of URL from Webpage is achieved.
- 2) Extraction of content from webpage.
- 3) Spell checker incorporated to identify misspelt words.
- 4) Extra feature added (suggestion of words).

This system extracts, collects and integrates the data from different levels of a website successfully. Each wrongly spelt word is identified and corresponding suggestions shown. Regular expressions were effectively employed to get rid of false positives – Names, URLs, acronyms etc.

Future Work

Future development of the software involves notifying the part of the page where the incorrect word occurred. This will allow the administrator to find the incorrect words quickly and replace it with the relevant word.

6. References

- [1]. <https://docs.python.org/3/tutorial/index.html> Accessed on Feb 2015.
- [2]. <http://www.tutorialspoint.com/> Accessed on March 2015.
- [3]. <http://www.devbistro.com/articles/Misc/Implementing-Effective-Web-Crawler>
Accessed on March 2015.
- [4]. <https://pypi.python.org/pypi/wikipedia> Accessed on March 2015.
- [5]. <https://media.readthedocs.org/pdf/requests/latest/requests.pdf> Accessed on March 2015.
- [6]. <https://docs.python.org/2/library/urllib2.html> Accessed on May 2015
- [7]. <https://www.youtube.com/watch?v=4Mf0h3HphEA&list=PLEA1FEF17E1E5C0DA>
Accessed on March 2015