Interactive English Dictionary

A PROJECT REPORT

Created By

PRITAM GANGULY

Saroj Mohan Institute of Technology(degree div)

Techno India Group

JUNE – JULY 2020

Declaration

I hereby declare that this project is based on my original work except for quotation and citation which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at SMIT or any other institution.

Name: Pritam Ganguly

Date: 28th JULY

TABLE OF CONTENTS

Introduction

The following is a documentation for the project “Interactive English Dictionary” which is a dictionary desktop application that features an interactive interface that shows meaning to any English word entered. The project was created solely using python and its tkinter module. No third party application was used.

1.1 Objective:

The main goal of this project is to learn the various ways of presenting data to the user which the user can access quickly and easily. More precisely, creating an interface for accessing dictionary database swiftly. The project mainly focuses on using user given data to fetch related data from the json data base which in this case holds the data of dictionary.

1.2 Scope:

Although this project isn’t made with much flexibility and adaptability, the core elements of this project is sufficient to provide simple explanation of the word provided. Hence, this can be used in places of learning or reading such as library, school, etc, as accessing meaning of complicated words is made easier with a simple click of a button.

System Analysis

2.1 Identification of Need:

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studies to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The System is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and Decisions variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem area are identified. The designer now function as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user .The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

2.2 Feasibility Study:

Feasibility study is made to see if the project on completion will serve the purpose the organization for the amount of work.

Effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system

proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provide the feasibility of the project that is being designed and lists various area that were considered very carefully during the feasibility study of this project such as Technical, Economic and operational feasibilities.

2.3. Hardware and Software Requirements:

HARDWARE REQUIREMENTS

- Computer that has a 1.6GHz or faster processor
- 2 GB (32 Bit) or 4 GB (64 Bit) RAM (Add 512 MB if running in a virtual machine)
- HDD 20 GB Hard Disk Space and Above Hardware Requirements 5400 RPM hard disk drive

SOFTWARE REQUIREMENTS

- WINDOWS OS
- python installed

3.1 Code With Explanation:

The following project was created by using three files. These are

1. data.json
2. mech.py
3. script.py

data.json

This is a json file that holes all data of dictionary in json format. The main goal of this project is to retrieve data from this data set. To do this we will use two python file called mech.py and script.py. The data stored in json file are kind of similar to the python dictionary module. Hence data retrieved from the json file can directly be used as a dictionary with a little bit of change.

Example:


```
"abandoned industrial site": ["Site that cannot be used for any purpose, being contaminated by pollutants."], "abandoned vehicle": ["A vehicle that has been discarded in the environment, urban or otherwise, often found wrecked, destroyed, damaged or with a major component part stolen or missing."], "abiotic factor": ["Physical, chemical and other non-living environmental factor."], "access road": ["Any street or narrow stretch of paved surface that leads to a specific destination, such as a main highway."], "access to the sea": ["The ability to bring goods to and from a port that is able to harbor sea faring vessels."], "accident": ["An unexpected, unfortunate mishap, failure or loss with the potential for harming human life, property or the environment.", "An event that happens suddenly or by chance without an apparent cause."], "accumulator": ["A rechargeable device for storing electrical energy in the form of chemical energy, consisting of one or more separate secondary cells.\\n(Source: CED)"], "acidification": ["Addition of an acid to a solution until the pH falls below 7….
```


mech.py

mech.py is actually the backend portion of this entire application. It does all the retrieving of data from the user given input.

First:

It imports *json* library and *get_close_matches* module from the *difflab* library. The json library is used to load the data.json file and the difflab’

get_close_matches function is used to fetch relative data of the provided user input in case the user misspelled.

```
import json
from difflib import get_close_matches
```

second:

Then it creates an instance of the json data set and names it *data*.

```
data = json.load(open("data.json"))
```

Third:

Now we create a function called *translate* that takes a single string as an input and uses that input as a key to retrieve corresponding values of the key from the data dictionary. The function also checks if the provided input is misspelled of not and tries to recommend possible solution. Then the function returns all retrieved data as a list.

```
def translate(w):
    w = w.lower()  # ignores the case type
    l = []
    # fetches the meaning if its in the database
    if w in data:
        return data[w]  # returns it
    # if it doesnt exists then finds the nearest key possible and asks for correction.
    elif len(get_close_matches(w, data.keys())) > 0:
        l.append("Did you mean {} instead?".format(get_close_matches(w, data.keys())[0]))
    # if it doesnt match any key in any way then informs the user that its not in the current dictionary
    else:
        l.append("The word doesn't exist. Please double check it.")
    # returns the retrieved lists of meanings for the given key
    return l
```

Script.py:

First it imports all necessary library. The *tkinter* library is used to create the User Interface for the end user. All the component classes of the *tkinter* module is used to create various UI components of he window. Next is the previous *mech* file that we just created as it holds the core retrieving process of dictionary data.

```
from tkinter import *
import mech
```

Next is the *onsearch* method which when called gets the use data and passed it to mech's translate method. Translate method then returns list containing all the meanings of the provided word. Then all the data are inserted into the text section of the user interface.

```
def onsearch():
    data = list(mech.translate(input_word.get().strip()))  # retrieves the meanings as a list
    # displays  each element of the meaning list into the text field.
    for r in data:
        text.insert(END, '->' + r + '\n')
    text.insert(END, '----------\n')  # extra line to show mark the end of a input .
```

The following process uses the tkinter module and creates a user interface.

1) creating the window

```
window = Tk()
```

2) providing the title

```
window.wm_title('English Dictionary')
```

3) creating a label called "search here"

```
l = Label(window, text="Search here")
l.grid(row=0, column=0)
```

4) adding an entry portion to the level

```
input_word = StringVar()
e = Entry(window, textvariable=input_word)
e.grid(row=0, column=1)
```

5) creating a text field

```
text = Text(window, height=20, width=50)
text.grid(row=1, column=0, rowspan=10, columnspan=3)
```

6) creating a scroll bar

```
sb = Scrollbar(window)
sb.grid(row=1, column=4, rowspan=10)
```

7) configuring the scroll bar with the text field

```
text.configure(yscrollcommand=sb.set)
sb.configure(command=text.yview)
```

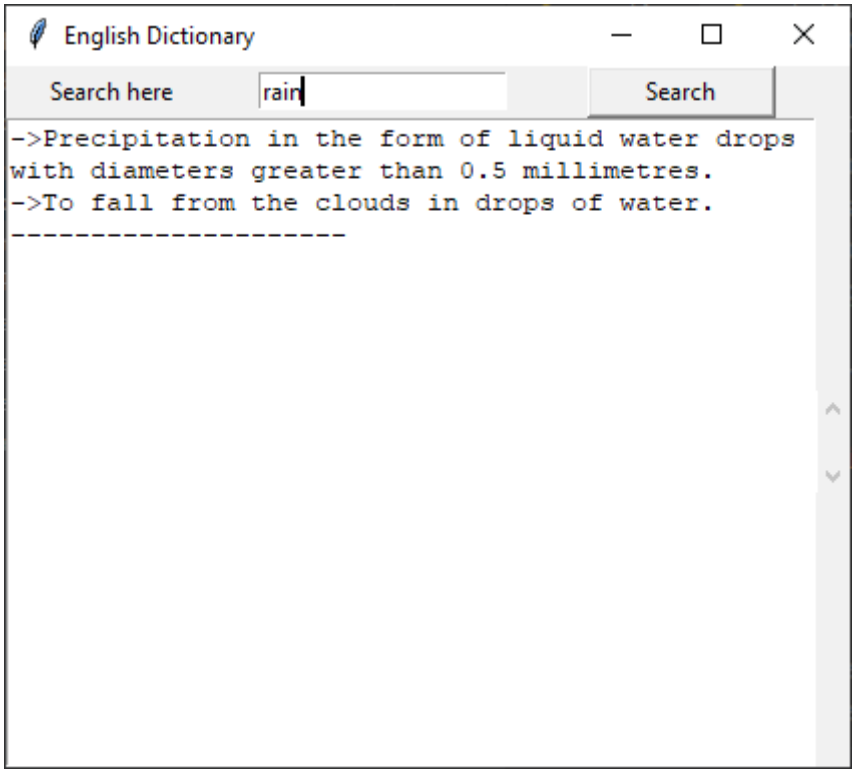
8) creating a button called search

```
b = Button(window, text='Search', width=12, command=onsearch)
b.grid(row=0, column=2)
```

9) keeping the window from disappearing

```
window.mainloop()
```

After this if we run the application we will see the following user interface.



Now its a fully fictional dictionary application.

Conclusion

With the usability of python as a coding language increases day by day, keeping up with all its application became necessary. Simplicity of python allows us to create otherwise complicated application in an instance. Python is dominating the industrial marketplace as the simple, easy to learn and at the same time powerful alternative to other coding languages. This project demonstrates a simple application of python that finds meaning of words in an instance. This is an example of how powerful Python as a coding language is.