

MALCOLM JEFFERS

CONTENT GENERATOR

PURPOSE

This document details the software requirements specifications for the Content Generator application. The purpose of this project is to provide a way to get content from Wikipedia from keywords.

PROJECT SCOPE

Content Generator is a tool that can take two keywords then search Wikipedia for a paragraph containing both keywords. It has a graphical Interface (GUI) and a shell interface(CLI). Along with the Wikipedia search, the GUI can also provide a random address for a state, and the population for the selected state. The CLI will take an input.csv file and search Wikipedia for content. It outputs the results to output.csv. The GUI will also output Wikipedia results to output.csv.

CURRENT AND PLANNED FUNCTIONAL REQUIREMENTS

USER STORIES

1. As a retailer, I want to generate realistic reviews about my new product so I can post them on Amazon and get more sales.

Inputs: Multiple keywords, length

Output: Text of the input length that contains all input keywords

2. As a writer, I want to see words with similar meanings to my input so I no longer have to struggle with finding flowery words.

Inputs: Simple words, number of synonyms to generate

Output: Synonyms

3. As an international student, I want the system to change a sentence with slang into a standard English expression so I can better integrate into the local culture.

Inputs: Slang English word

Output: Formal English equivalent

4. As a writing class student, I want to change some words in a paragraph while keeping the same meaning so that I avoid plagiarizing.

Inputs: English text, indication of which words to change

Output: Paraphrased text with indicated words changed

5. As a data scientist, I want to export the results in CSV format so I can import it into my database as a test dataset for training my machine learning models.

Inputs: Data

Outputs: CSV file with header and one row per line

6. As a college student, I want to be able to enter a primary and secondary keyword, then get a paragraph from Wikipedia with both keywords in it, so I can complete research papers.
7. As a spam-mailer, I want to get a random address, along with Wikipedia contents, so I can send random mail to people.
8. As a call center employee, I want to get a Wikipedia article and the population of a state, so I can impress people on phone calls.

USE CASE

Name: Content Generator

Actor: College Student

Flow:

Basic Flow GUI

1. A student opens the software GUI.
2. A student enters a primary keyword.
3. A student enters a secondary keyword.
4. A student clicks button to generate output.
5. Software searches Wikipedia for the primary keyword.
6. After finding the Wikipedia page, software searches for text for a paragraph matching the primary and secondary keywords.
7. Software displays the results of the search in the output box for the student.
8. Software stores the primary, secondary, and output content in a CSV file.

NON FUNCTIONAL REQUIREMENTS

- Portability: The software must be able to run on the following platforms:–Windows 10–MacOS 10 or newer–Ubuntu 20 LTS or newer
- Performance: Display the results within 15 seconds after submitting input
- Usability: An independent evaluator must determine the GUI reflects at least 2/3rds of the Cognitive Style Heuristics at the “satisfactory” level or higher

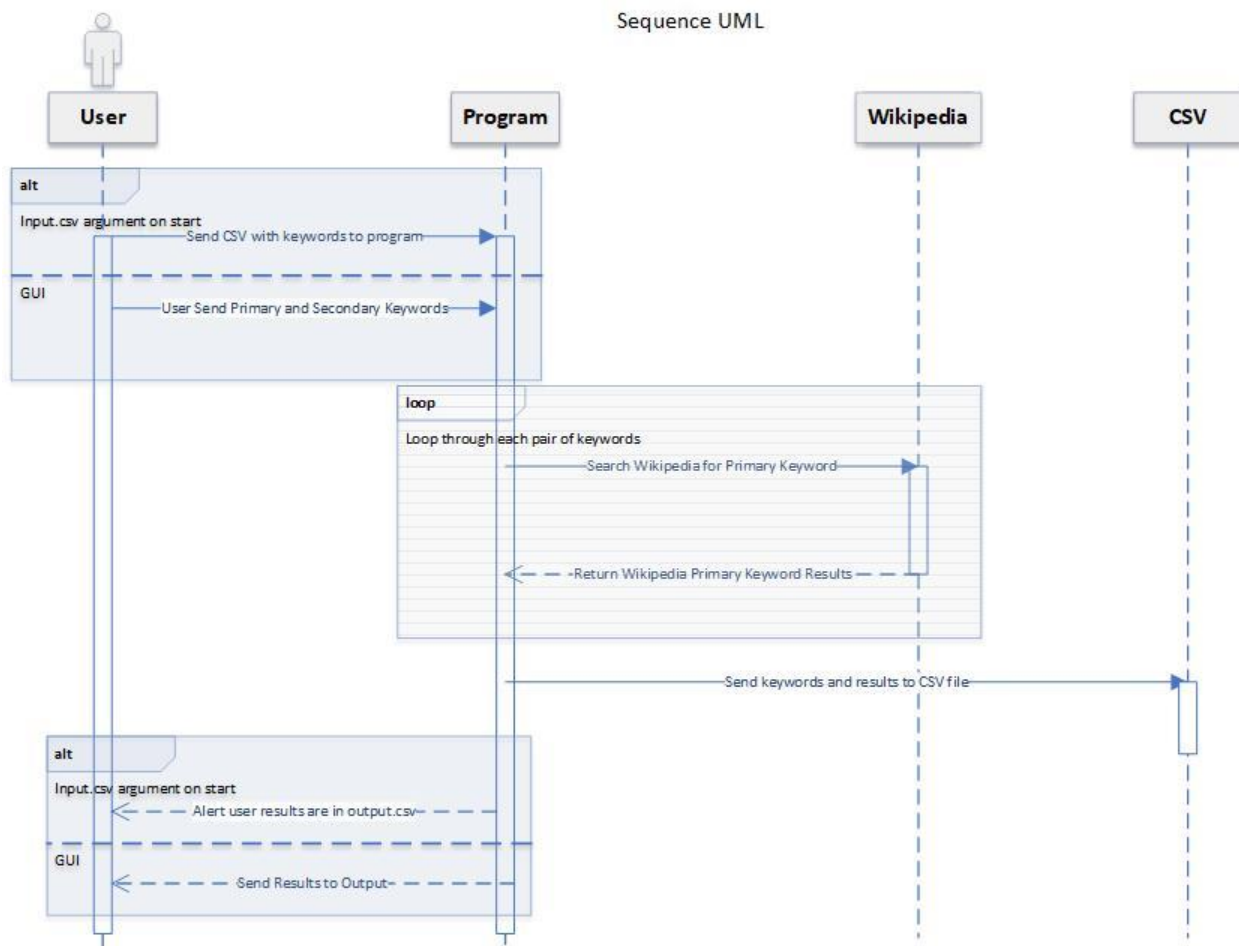
HIGH LEVEL ARCHITECTURE

This program uses Microservice Architecture. It is only dependent on itself for its main functions. Additional functionality can be obtained when the program communicates with Population Generator, or Person Generator.

MICROSERVICE COMMUNICATION

The content generator program can communicate with Population Generator and Person Generator through local ports. Starting a server will allow Content Generator to communicate on local host port 5001. Person Generator uses port 5003 while Population Generator uses 5002. They function using a REST API. To get info from another service a GET request is sent on the required port.

CODE DESIGN



GUI DESIGN

The screenshot shows a window titled "Content Generator" with a menu bar containing "File" and "Help". Below the menu bar, there is a text instruction: "Input a Primary and Secondary Keyword. Then press enter to generate results from Wikipedia." This is followed by two input fields for "Primary Keyword" and "Secondary Keyword". Below these are two checked checkboxes: "Create a random address for a selected state" and "Get population of a selected State". A "Select a State." label is followed by a dropdown menu currently showing "Alabama". To the right of the dropdown is a "Generate Content" button. Below the input fields is a section labeled "Wikipedia Results" which is currently empty. At the bottom, there are two more sections: "Created Random Address" (empty) and "Population of Selected State in 2019" (empty).

The GUI design shows several Cognitive Style Heuristics. The design has heuristic #2 by explaining what each option does. The design also has heuristic #3 in mind. The program has a way to get more information if needed. Selecting Help and see documentation will allow the user to view the current documentation about the software.

HOW TO RUN CODE

To run the code the following packages must be installed: BeautifulSoup 4 and Flask

To install BeautifulSoup 4, run the following command in your terminal.

ex: `pip install bs4`

To install Flask, run the following command in your terminal

ex: `pip install flask`

RUN AS CLI

To Run content-generator.py as CLI have a csv with primary and secondary keyword in the same folder. The file must be named input.csv

ex: python content-generator.py input.csv

The input file needs a csv in the following format. A header named keywords, then the primary and secondary keyword separated by a semicolon. You can have as many keyword pairs as you like. The results will take longer depending on the length. An output.csv file will be created. It stores results with a header as, input_keywords, output_content. The previous keywords are combined in input keywords and separated by a semicolon. The output_content contains the first paragraph from Wikipedia with both keywords in it.

```
keywords
Puppy;Large,
cat;mammal,
```

RUN AS A GUI

ex: python content-generator.py

Enter a primary and secondary keyword in their designated boxes. Then click generate content. The results will appear in the output box and in the output.csv file. If you want to get a random address, you will need to have the server running for Person Generator microservice. You will then get the random address of the state that is selected. To get the population of a selected state, the Population Generator microservice will need to be running.

RUN AS MICROSERVICE

To run as a microservice on your system enter the following command. This will start a service on local host port 5001

ex: python server_app.py

KNOWN ISSUES

Only a few states are supported for creating a random address. To properly run that microservice, the corresponding csv file for the state you want addresses for will need to be downloaded along with the person generator program you wish to use.