**A Web Search Engine – Client-server Architecture documentation**

The Process Architecture

The 4+1 View Model

Scenarios

Development View

Physical View

Process View

Logical View

User Manual

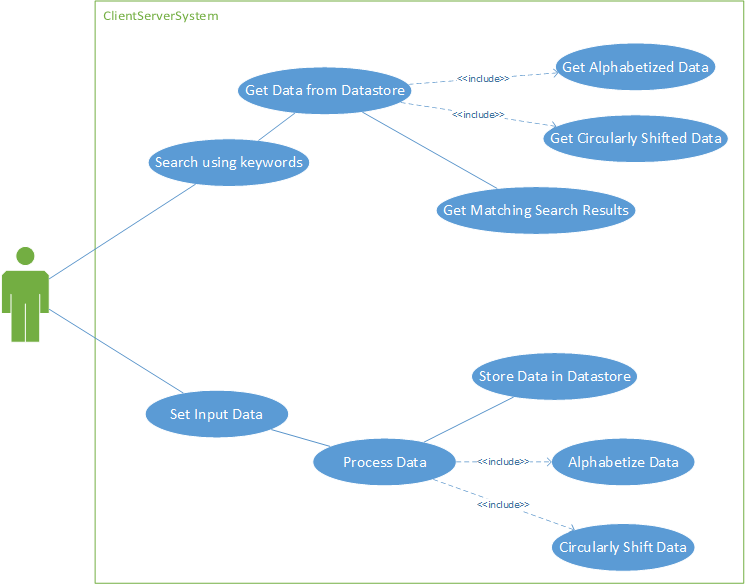
The Process Architecture

Sophie did the parts of the documentation. Brandy worked on the code implementation and on designing the diagrams and the rest of the documentation. These particular roles were assigned based on each member’s ability to perform the task. We communicated by email and used GitHub.

**The 4+1 View Model**

*Scenarios*

Use Case Diagram

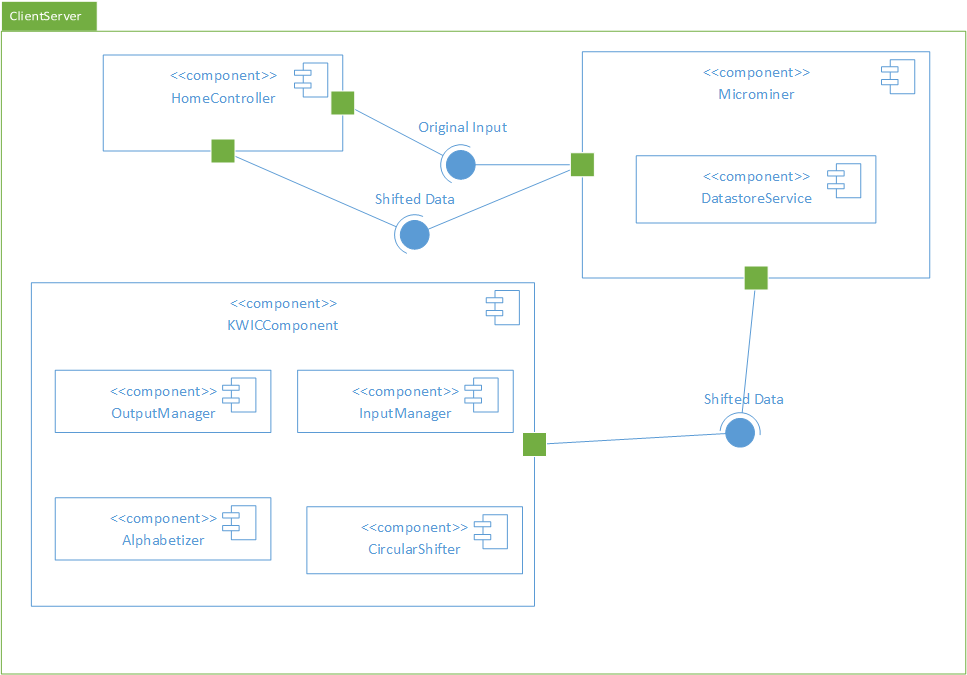


The user starts at the data input screen where he/she can set the data that will be used for searching. This searchable data consists of keywords (such as “University of Central Oklahoma”) and then a url (such as “http://www.uco.edu”), which are separated by a space. When the searchable data is set, the data is circularly shifted, alphabetized, and then stored. Once the data has been set, the user is redirected to the search page. The user can also navigate directly to the searching page, but without any searchable data, the user won’t be able to search for anything.

On the searching page, the user is given a text box to use for input. The system searches for matching keywords, such as “University” or “texas.” The system is case-insensitive. The system will find matching keywords and then output one instance of the match as well as the matching url. If input given is “Texas” and the system’s searchable data contains “University of Texas,” “Texas State University,” and “Oklahoma State University,” followed by their urls, it will return “University of Texas” and “Texas State University” followed by their respective urls. The system also displays the circularly shifted and alphabetized data from the original searchable input.

*Development View*

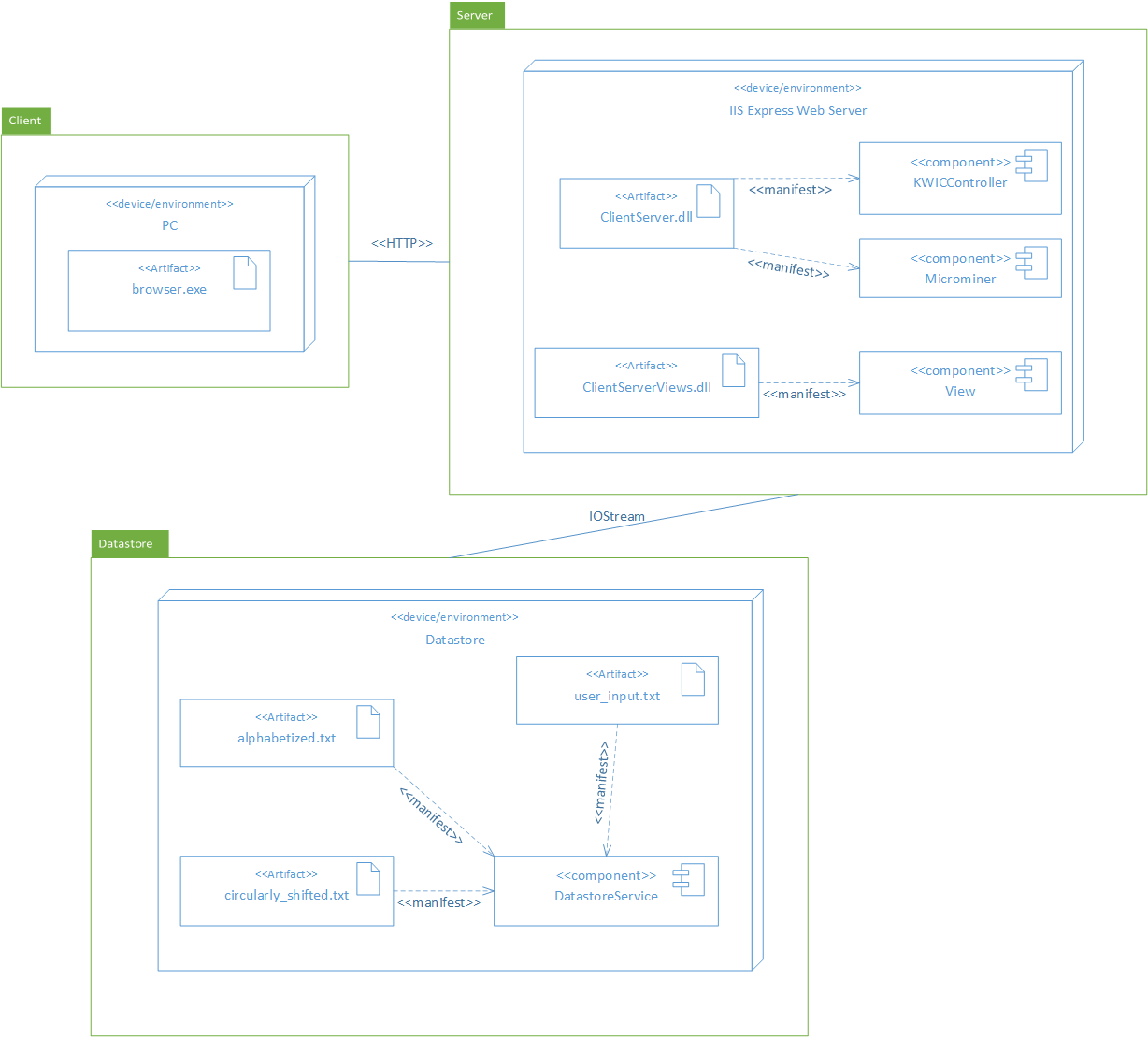
Component Diagram



The Microminer requires the alphabetized and circularly shifted data from the KWICComponent. The HomeController requires the original input and the shifted data from the Microminer. The Microminer contains and manages the DatastoreService.

*Physical View:*

Deployment Diagram



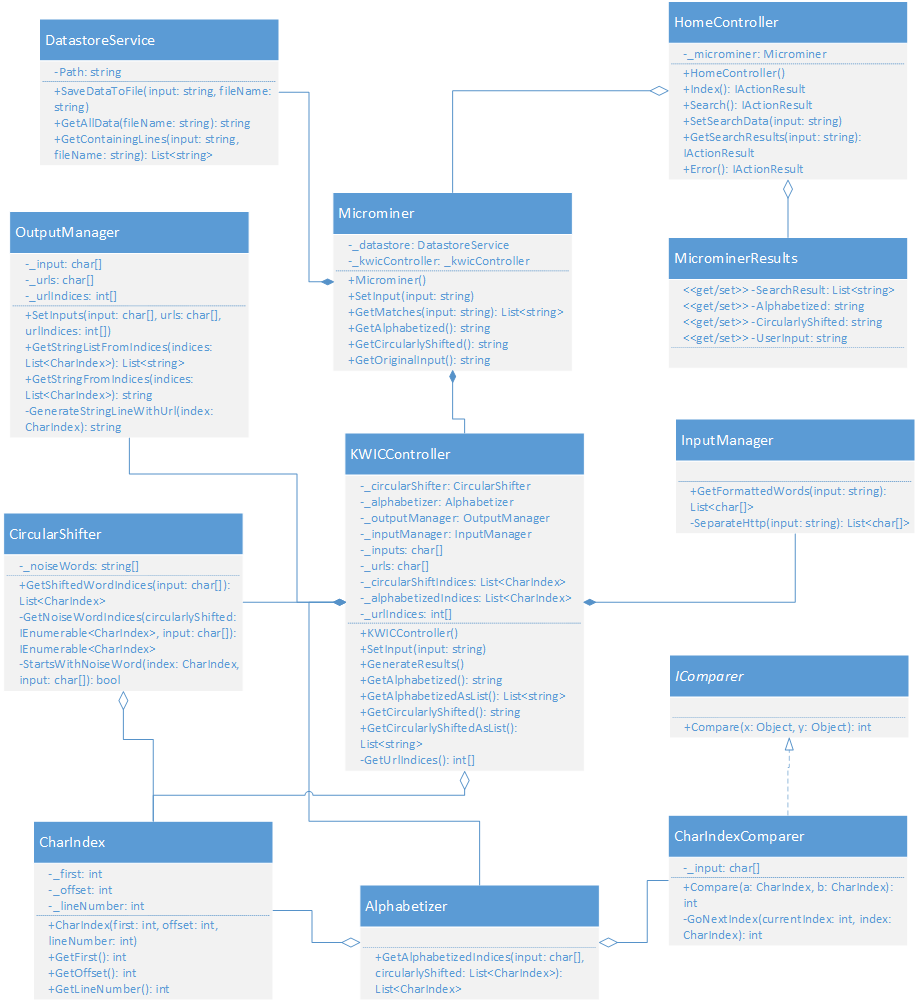
The client is the browser used to access the website (such as Chrome, Edge, Safari, etc). The client and the server communicate using HTTP protocol.

The server is the web server, which in this case is the IIS Express Web Server. It contains the Views, the KWIC component, and Microminer component.

The datastore consists of the text files that store the data used by the system. The server uses IOStream to read and write files.

*Logical View*

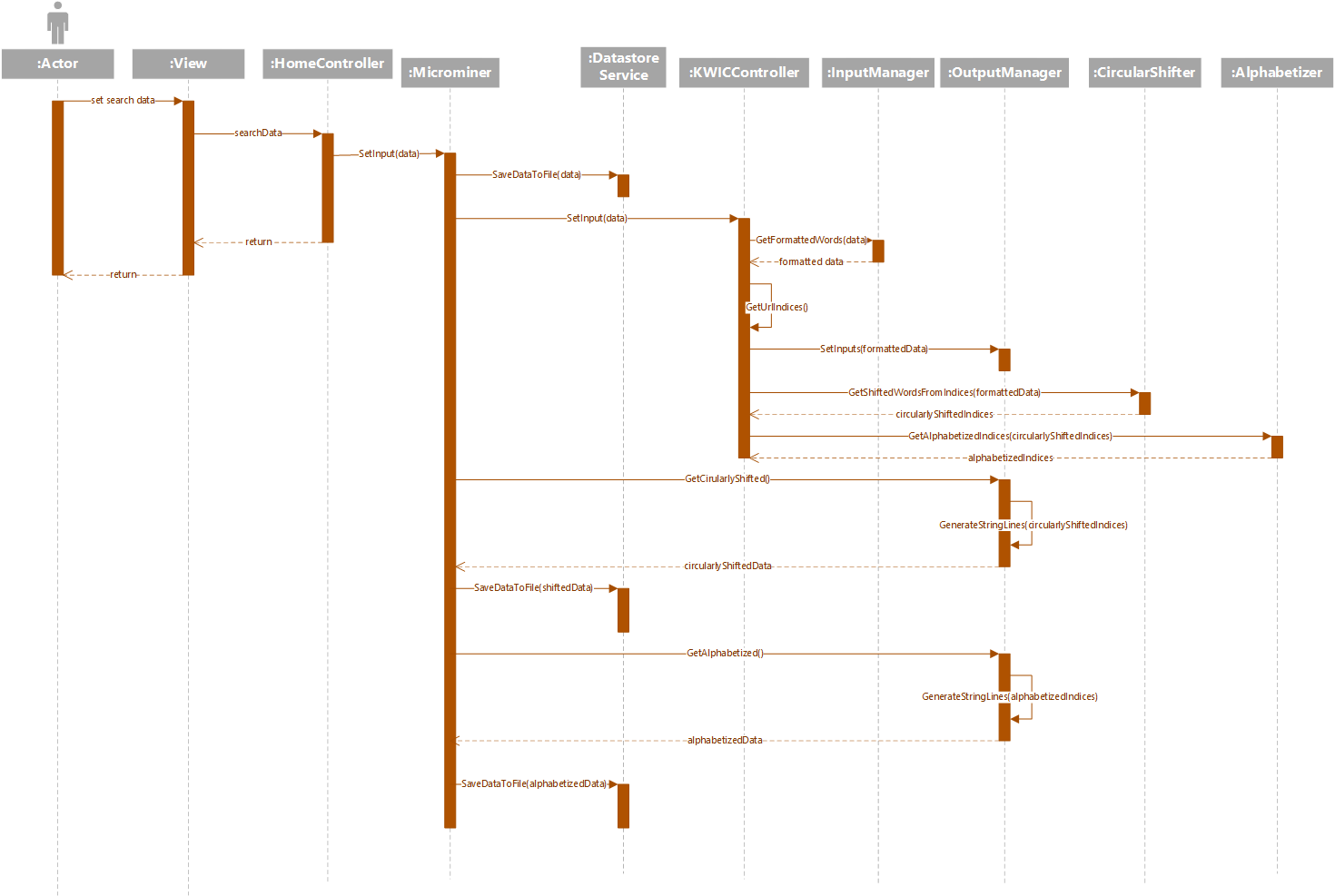
Class Diagram



The program utilizes the KWIC system, which uses the Shared Data architecture. Additionally, the program has the Microminer, which acts as the interface for getting the alphabetized data, circularly-shifted data, and search results. The Microminer uses the DatastoreService to interface with the file system that saves the alphabetized data, circularly-shifted data, and searchable data. The MicrominerResults model is passed to the view to be displayed to the user.

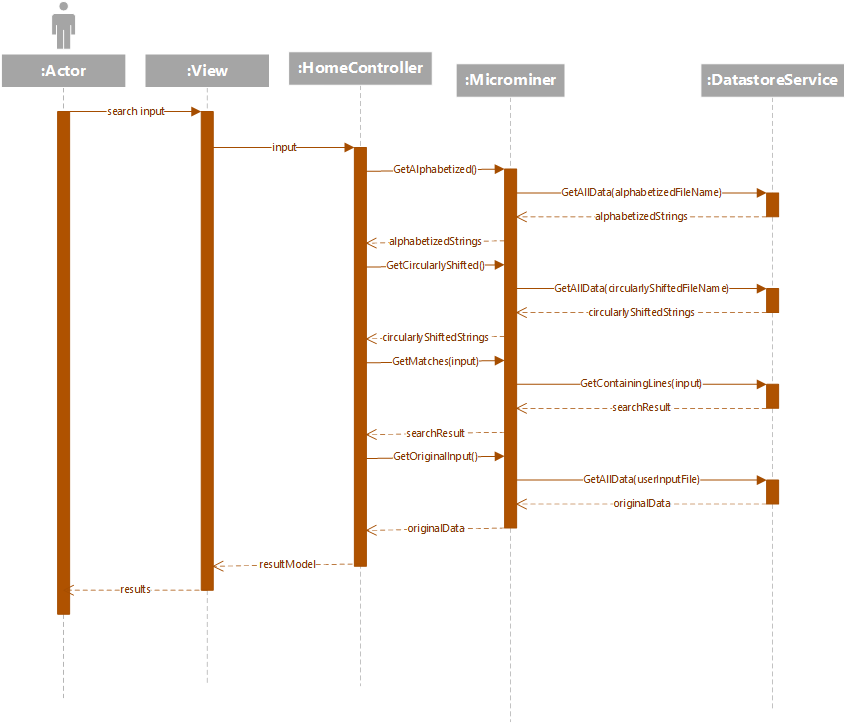
*Process View*

Sequence Diagram for setting the Searchable Data



The user comes in and sets the search data. That data gets passed through the view into the controller then to the Microminer. The original searchable data gets saved to a file. Then, the data is moved through the shared data KWIC system where it is circularly shifted then alphabetized. The circularly shifted data and alphabetized data are saved to files as well.

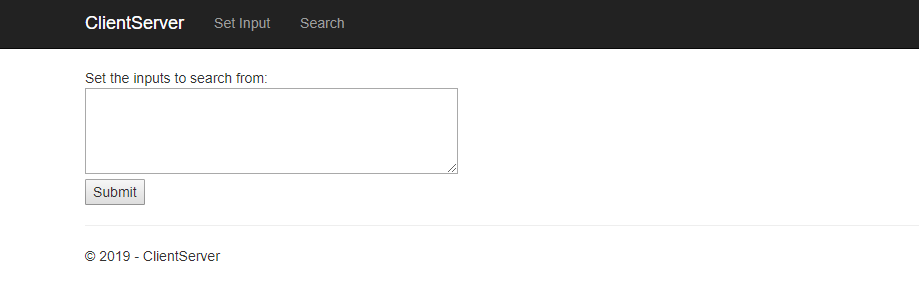
Sequence Diagram for searching



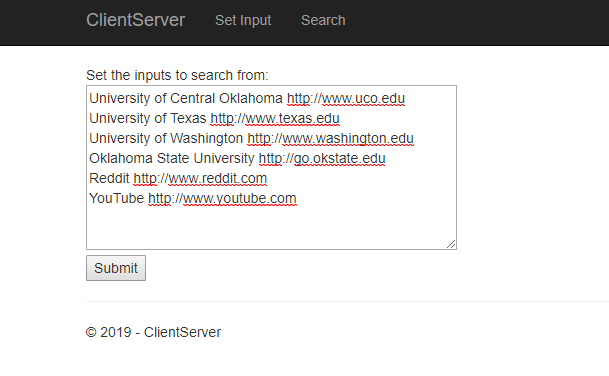
The search input is moved from the view to the controller then to the Microminer. The Microminer calls the DatastoreService to get the stored alphabetized strings and circularly shifted strings. It also uses the Datastore to find each line containing the given search input within the searchable data file. It also retrieves the original searchable data. All of the retrieved data gets sent back to the view and is rendered.

User Manual

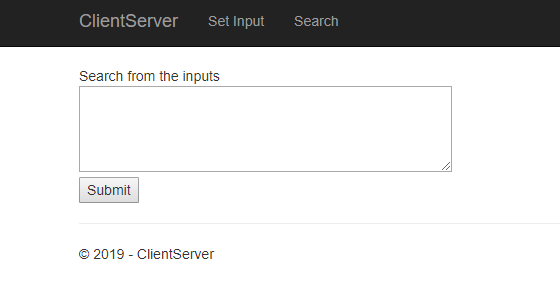
1. Load the website. The first page loaded will be the page used for inputting the “searchable” data.



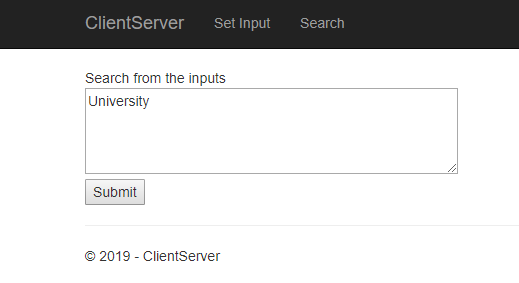
1. Input searchable data that you want stored in the database. Click “Submit”



1. You will be redirected to the search page as seen here:



1. Enter in keywords used for searching and then click “Submit.”



1. The program will display the search results. Underneath the search results, the program will display the original searchable data, the circularly-shifted data, and alphabetized data.

