Final Report

Boqian Fan 1002342529
Cheng Cao 1002265372

PROJECT OVERVIEW:

The project of CSC326 is to build a search engine by bottle web framework with the support of crawler algorithm to capture the information from indicated websites.

The project consists of four phases, with the first phase building basic homepage frontend and data storage by crawler, with the second phase integrating by third party API such as AWS cloud deployment and google auth2client, with the the third phase sorting pages by page rank algorithm and implementing pagination with the database sqlite3 or redis, and with the final phase which is open-ended.

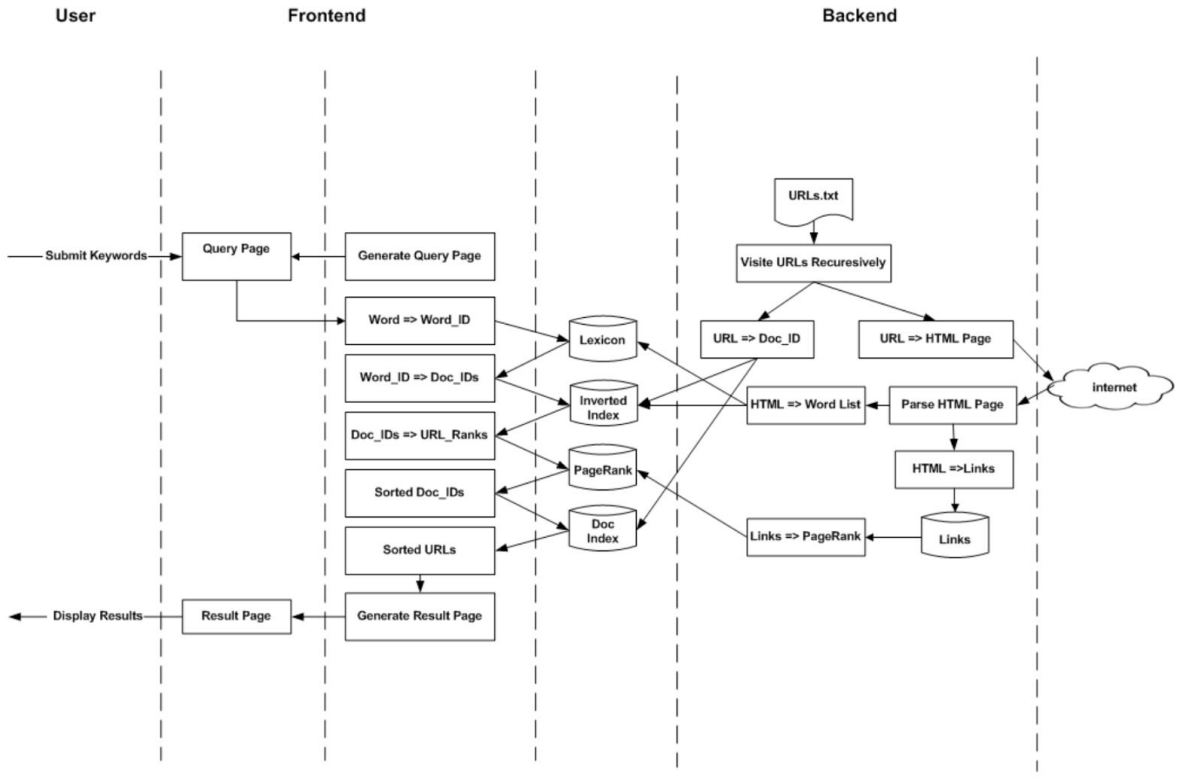
Our enhanced search engine mainly focuses on the user experience from frontend. For example, we implemented the responsive web design by detecting different devices and then responding the user with a more comfortable and delicate frontend. We designed our website in cell phone, ipad, and pc platform. Besides, we add two animations by jQuery for our homepage which one is the expansion of our logo and the other one is the stretch of our search bar. However, the two animations are restricted with PC only, because we consider the efficiency for cell phone and ipad platform. By using jQueryUI autocomplete library, we successfully finished the interaction between python and javascript by putting the data from python on HTML tag, and then javascript read the data from the tag to store in the autocomplete library. Moreover, except for the display of a website title, we also displayed the words for each crawled website under its URL, which offers the user with a generalized view for his or her targeted information. Some small enhancements such as when the search bar is clicked, the search word will be disappeared, which was implemented by simple javascript code to improve user search experience, and also spell check for input tag by HTML5 build-in attribute. Furthermore, in consideration of the languages of our website, we referenced Google Translator web plug-ins on the top left of our website for the user to read our website in his or her most comfortable languages. In addition, each sub domain website can detect whether the user logins or not, and correspondingly display sign in button or the user profile picture.

TESTING:

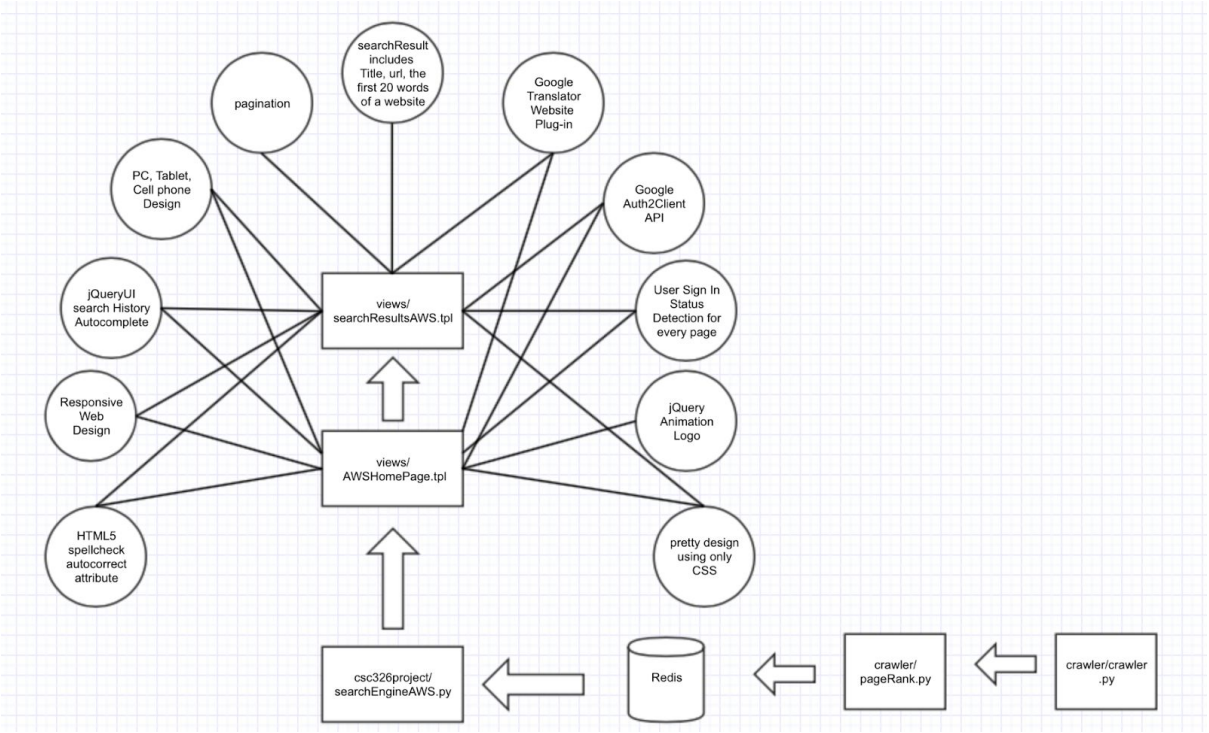
For front end, we tested our website frontend by google chrome “inspect” with cell phone view and pc view respectively. We tested every type of phones with our website, such as iphone 5, iphone 6, nexus 6p, ipad, ipad pro, iphone 6 plus and so on. We found that for the width of different devices there is indeed some impact on our website. We also tested our website as previous labs for concurrent connection, although for this time the concurrent

connection drops down to 150, because we implemented more on front end, and it did become slower.

HIGH LEVEL CODE UML:



The UML relation diagram above is provided from course lab handout, and we design our project based on this.



The diagram above shows the main feature that our search engine contains.

LESSON LEARNED:

This project really teaches me a lot about both the frontend and the backend. Especially, it is my first time creating a personal website and implement the website with python. I learned how to use python, bottle, crawler, redis, google api, boto api, aws deployment, jquery, javascript, and sqlite3. However, there is still a lot to pay attention in future development. For example, writing a scalable and robust program is crucial, because every phase of the project needs to add more features based on the previous program. If only consider every phase as an independent task to accomplish without attention to future development, the program is extremely hard to modify and also inefficient. Besides, the ability to read API documentation is also important. Although, at the beginning, it is not easy to follow and fully understand how to use the api, I start to test in other files to learn how to use them correctly. In addition, I encountered an issue with AWS account and password exposed on github in consequence of charging me 130 dollars. What I did at that time was immediately closed my account and followed the AWS case instruction to delete my root access key and password. Luckily, I got no charge for this accident, but this issue did raise my attention for computer security. Therefore, I bought a private github repository and also tried to use md5, RSA to encrypt the files. Comparatively, I found that development with only CSS is less efficient with Bootstrap or any other existing front end frameworks, because the frameworks provide you with more standard and efficient code than what you wrote in pure CSS. Hence, next time, I will try to use LESS, bootstrap, react.js, angular.js to develop higher performance and delicate website.

What I would do differently next time?

For front end part, I would like to use LESS, bootstrap, react.js, and angular.js to develop higher performance and delicate website. For back end part, I would consider higher efficient data structure to process the data.

If I had more time?

I would like to implement simultaneous auto-correction when the user types a word on input bar. I would also consider to use nginx to improve my network concurrent connection. Moreover, consider use amazon dynamoDB and load balancer to initiate many servers and given the user ip address to direct the user to the servers belonging to that region and also at the same time balance the load for user-located regions' servers. Besides, I would also implement multiple words search and even use python machine learning framework tensorflow to guess what the user might input and send his most favourite results given data collected before, which makes the search engine more intelligent.

The material from the course did give you a complete usage of python, it includes a variety of programming paradigms. However, this project takes more time dealing with web programming, which means knowing how to use python is still not enough to accomplish a supreme design. You also need knowledge for HTML5, javascript, CSS3, and any famous web framework.

I think that every part of the project is useful, because if you really want to learn a lot and you are also targeted to be a full stack developer. This project will be an introduction to web programming, which indeed offer you with a sense of current web development career. Learning more is necessary no matter back end or front end. However, I did recommend the project to encourage student to use Django and Flask to develop the website, because these two are the most famous web framework for python, which is going to familiarize students and also help students with future web development career.

Overall, I personally enjoy this project, because it is my first experience with web programming. I indeed learned a lot from this project. Because of this project, I accepted a website developer PEY job which will start in May 21th for one year. I will study more on web programming and aim to develop industrial standard website.

I hope the course note posted in time… Can we get more past exam to practice with?

CSC326_group_27 consists of two people. Boqian Fan is mainly in charge of implementing this project, and he developed both frontend and backend. Cheng Cao gives advisements on every design mainly acting as an UI designer.