

CS 4376 Project Presentation

By: Bhargav, Chase, Sergio,
Hayden, Ebrahim

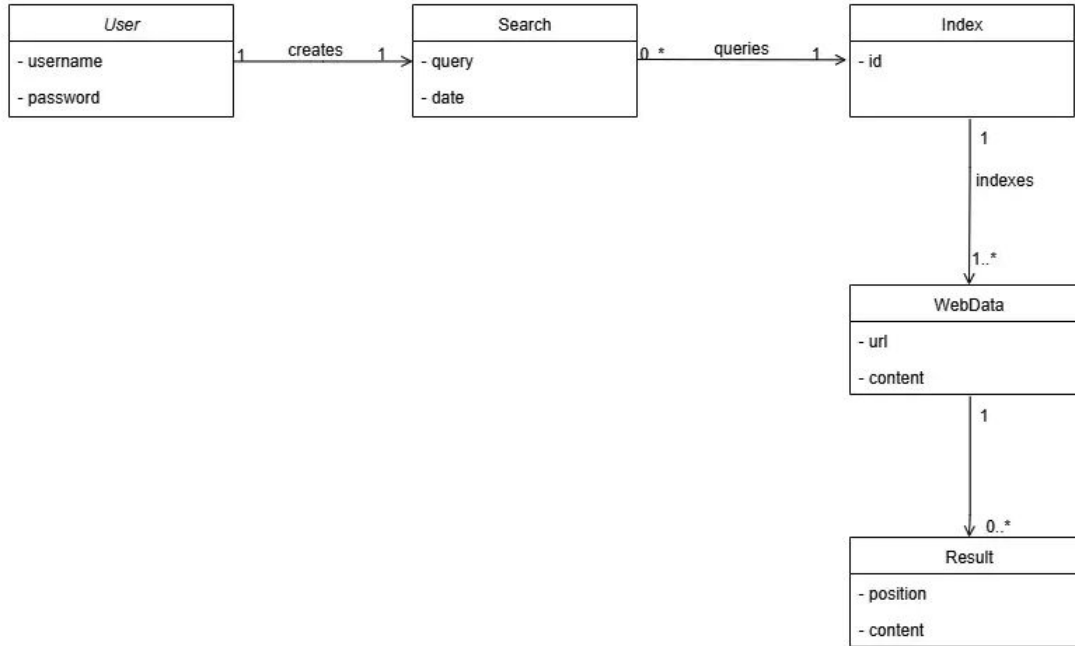


Project Overview

In this project, we will be making a search engine. Users will type what they want in a search bar and the site will show websites in order of how often your keywords appear.

Business Model

Domain Model



Shows basic structure and interconnectivity of the system.

Requirements Analysis

Functional Requirements

Cyberminer shall accept a list of keywords and return a list of URLs whose descriptions contain any of the given keywords.

Cyberminer shall allow for:

- Case sensitive search: The system shall store the input as given and retrieve the input also as such;
- Hyperlink enforcement: When the user clicks on the URL, which has been retrieved as the result of a query, the system shall take the user to the corresponding web site;
- Specifying OR/AND/NOT Search: A keyword-based search is usually an OR search, i.e., a search on any of the keywords given. The system shall allow the user to specify the mode of search, using “OR”, “AND” or “NOT”;
- Multiple search engines: to run concurrently;
- Deletion of out-of-date URL: and corresponding description from the database;
- Listing of the query result in ascending alphabetical order; most frequently accessed order, or per payment,
- Setting the number of results to show per page, and navigation between pages;
- Autofill, while correcting typographical errors,
- Filtering out symbols that are not meaningful, according to the user configuration.

Non-Functional Requirements

Cyberminer shall be easily understandable, portable, enhanceable and reusable

with good performance. The KWIC system shall also be user-friendly, responsive, and adaptable.

Stakeholders

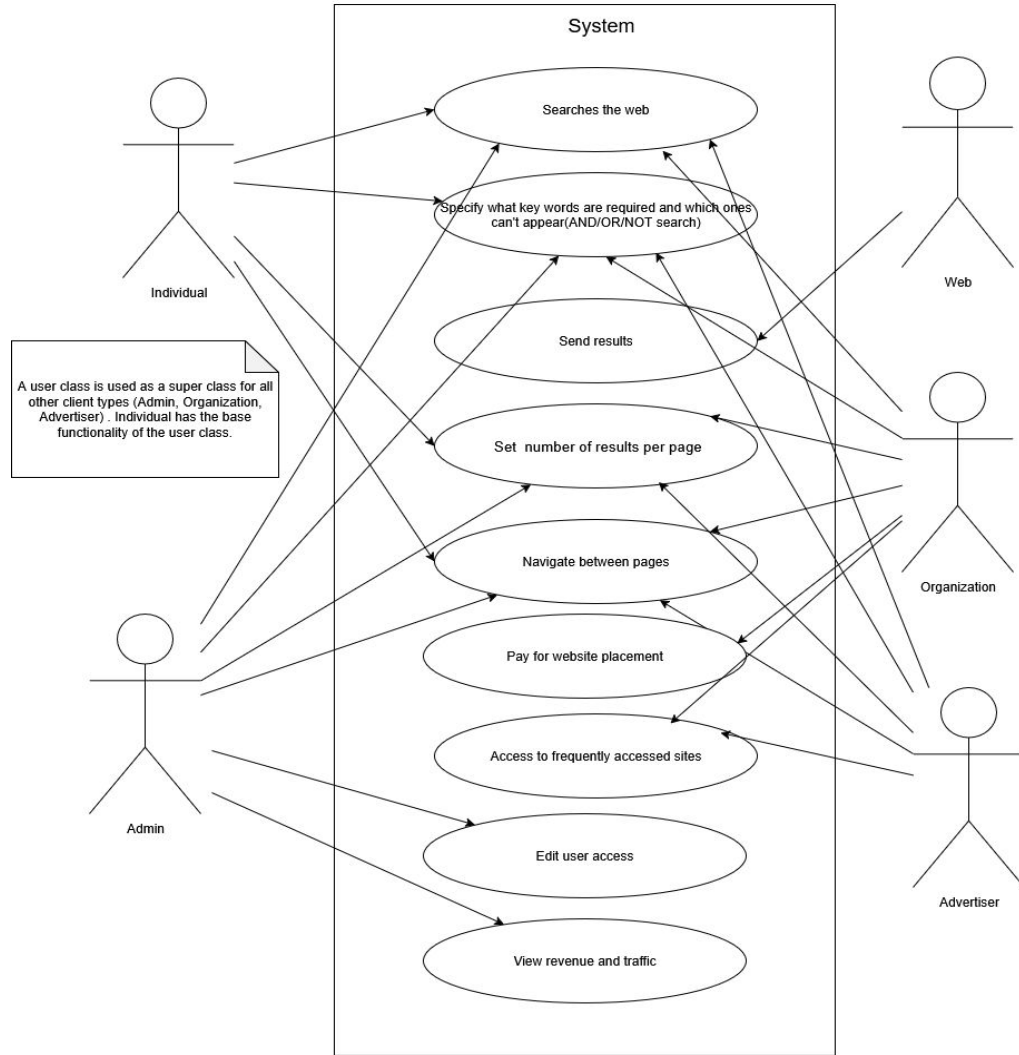
Organizational: Ability to pay for website placement

Individual: Basic features

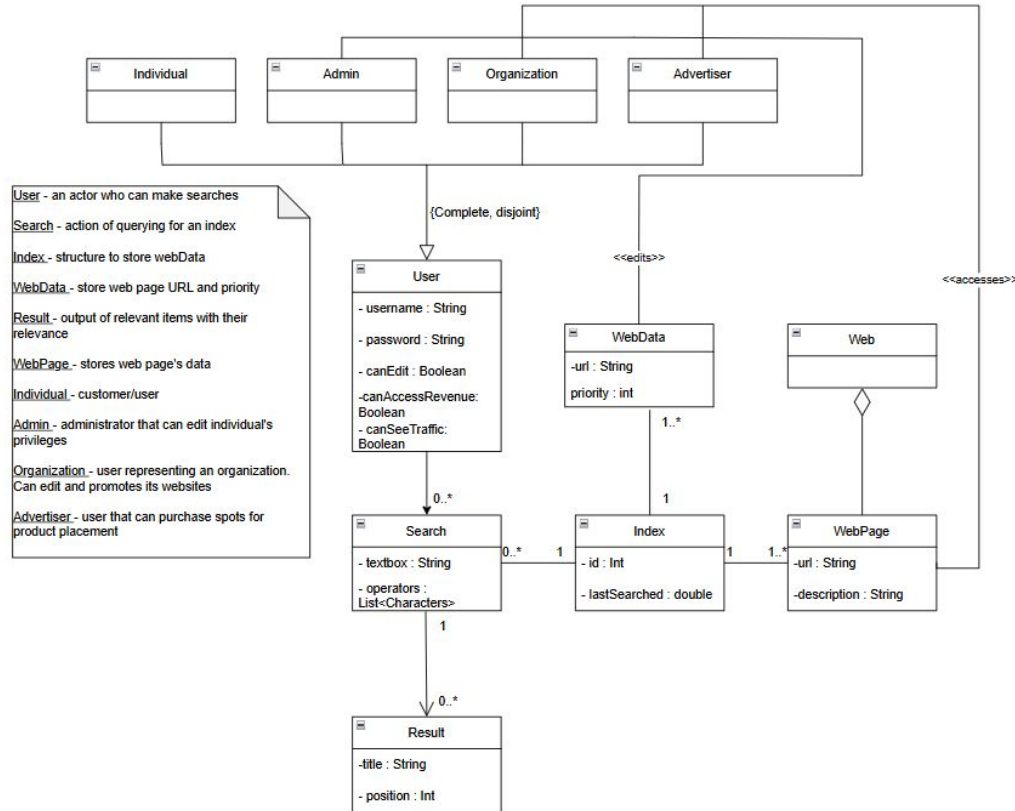
Advertisers: Access to frequently accessed sites

Admins: Edit access, ability to view revenue and traffic

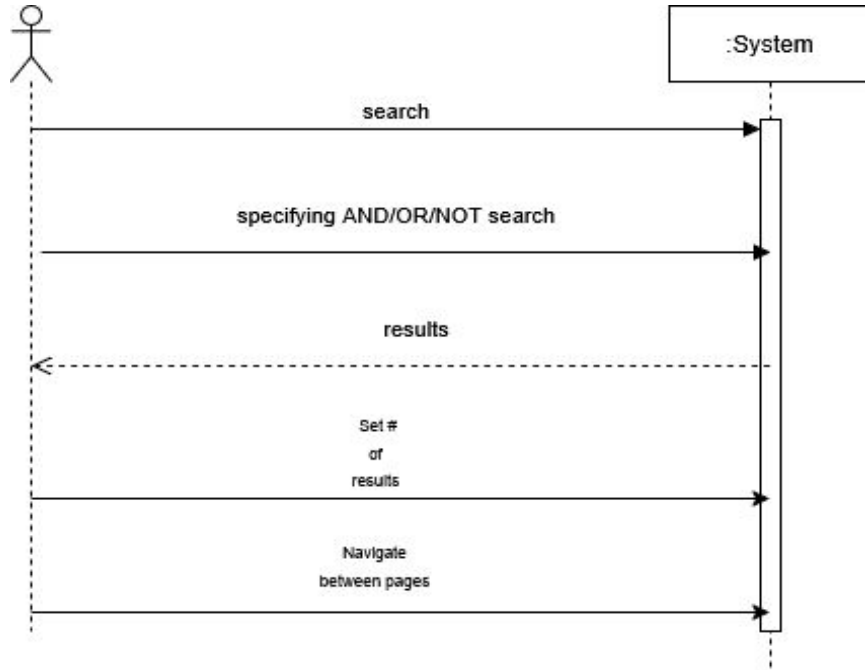
Use Case Diagram



Requirements Class Diagram



Individual Search



Use Case Text:

1. The user provides keywords and operators for their search.
2. System returns results.
3. User can adjust returned results by setting the number that appear on each page and navigating between results pages.

Operations Contract:

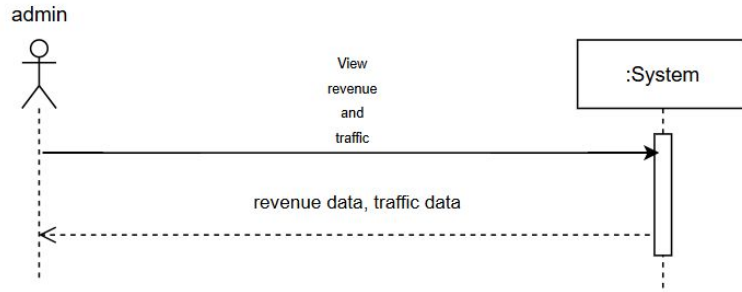
Operation: search()

Cross References: Use Cases: Searches the web

Preconditions: None

Postconditions: A results object is created

Admin Use Case



Use Case Text:

1. Includes all of the previous behaviors from Individual
2. Admin can edit another user's permissions
3. Has access to revenue and traffic data

Operations Contract:

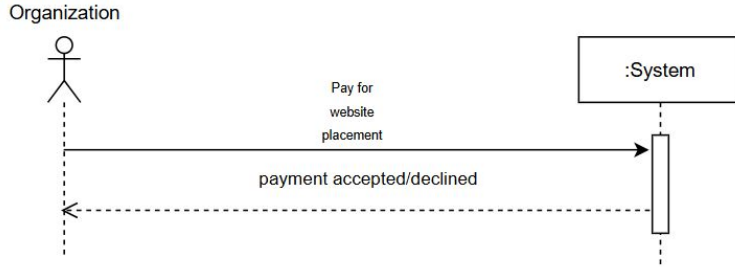
Operation: getTrafficAndRevenue()

Cross References: Use Cases: view revenue and traffic

Preconditions: None

Postconditions: None

Organization Use Case



Use Case Text:

1. Includes all of the previous behaviors from Individual
2. Organization can make payment to promote their URL

Operations Contract:

Operation: purchasePlacement()

Cross References: Use Case - pay for website placement

Preconditions: none

Postconditions: A payment instance is created and a payment status is returned

Advertiser Use Case



Use Case Text:

1. Includes all of the previous behaviors from Individual
2. Advertisers can request the most frequently accessed websites

Operations Contract:

Operation: viewTopSites()

Cross References: Use case - access to frequently accessed websites

Preconditions: none

Postconditions: A list of top websites is returned and displayed to advertiser

Project Deliverables

Class, Use Case, and sequence diagram - June 24th

Interim Project 1 (PPT) - June 26th

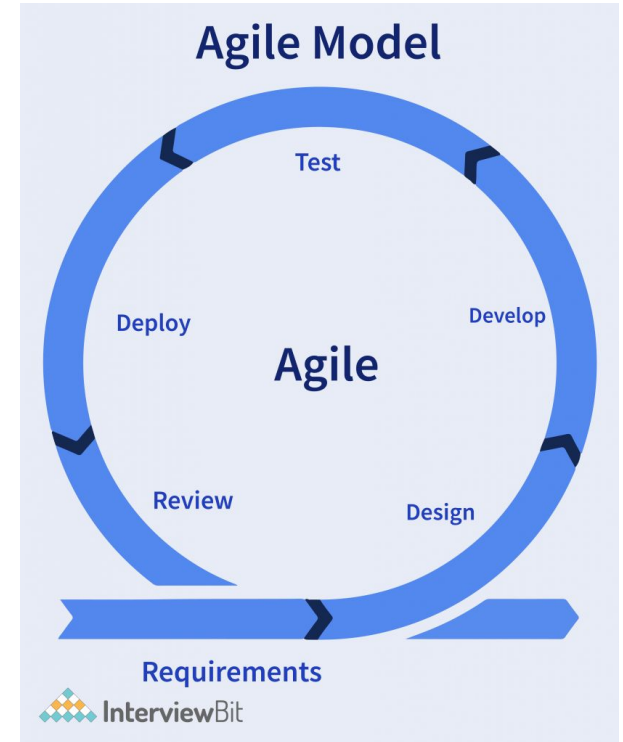
Final Project 1 - July 8th

Interim Project 2 - July 22nd

Final Project - August 7th

Process model

We will be using the agile model to do our project.



Important Terms

Search Engine: a website used to look up websites on the internet

HTML: Hypertext Markup Language. Standard markup language to be used for displaying in web browsers

JavaScript and CSS: Programming languages used with HTML

Agile Model: Process for developing software. Test software and get feedback to improve it

Bootstrap: Bootstrap is a free and open source CSS framework that utilizes Javascript for front end web dev.

Organization

We have a team of five and we'll be rotating leadership every 2 - 3 weeks. All suggestions will be directed towards the current team leader. All final decisions will be made by them.

The group will meet at least once before a due date to ensure that everyone knows what to be doing. All work will be divided evenly amongst the five group members.

Managerial Process

There will be four different phases of this project. Interim Project 1, Final Project 1, Interim Project 2, Final Project 2. For each phase a different group member who has not been leader will be the leader of that phase.

Assumptions - every group member will give an appropriate amount of time each week to contribute and fulfill their responsibilities. Dependencies will become apparent at a later point in the project and will be dealt with accordingly.

Constraints - the due dates mentioned before. Must complete before those deadlines.

Monitoring our progress will be done by our meetups. We will run the code consistently after major and minor changes to ensure that bugs are not created.

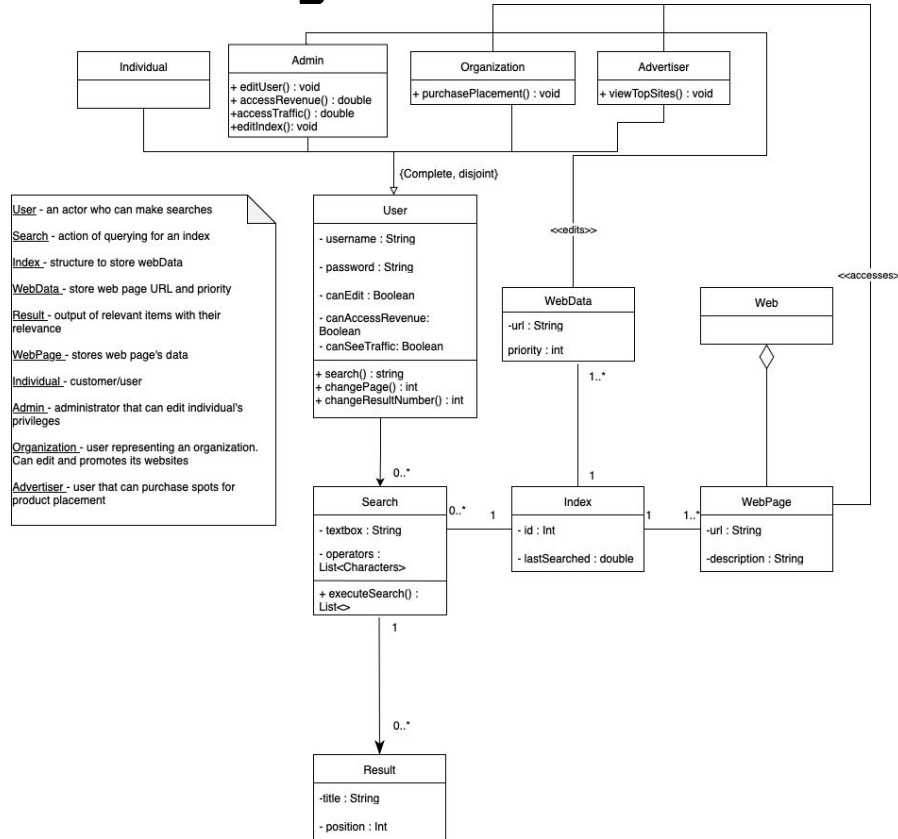
Technical Process

We will use existing IDEs and Bootstrap to help build our website.
Documentation will be done using UML practices.

Budget - \$0. All done on currently owned software.

High-Level Design

High-Level Class Diagram



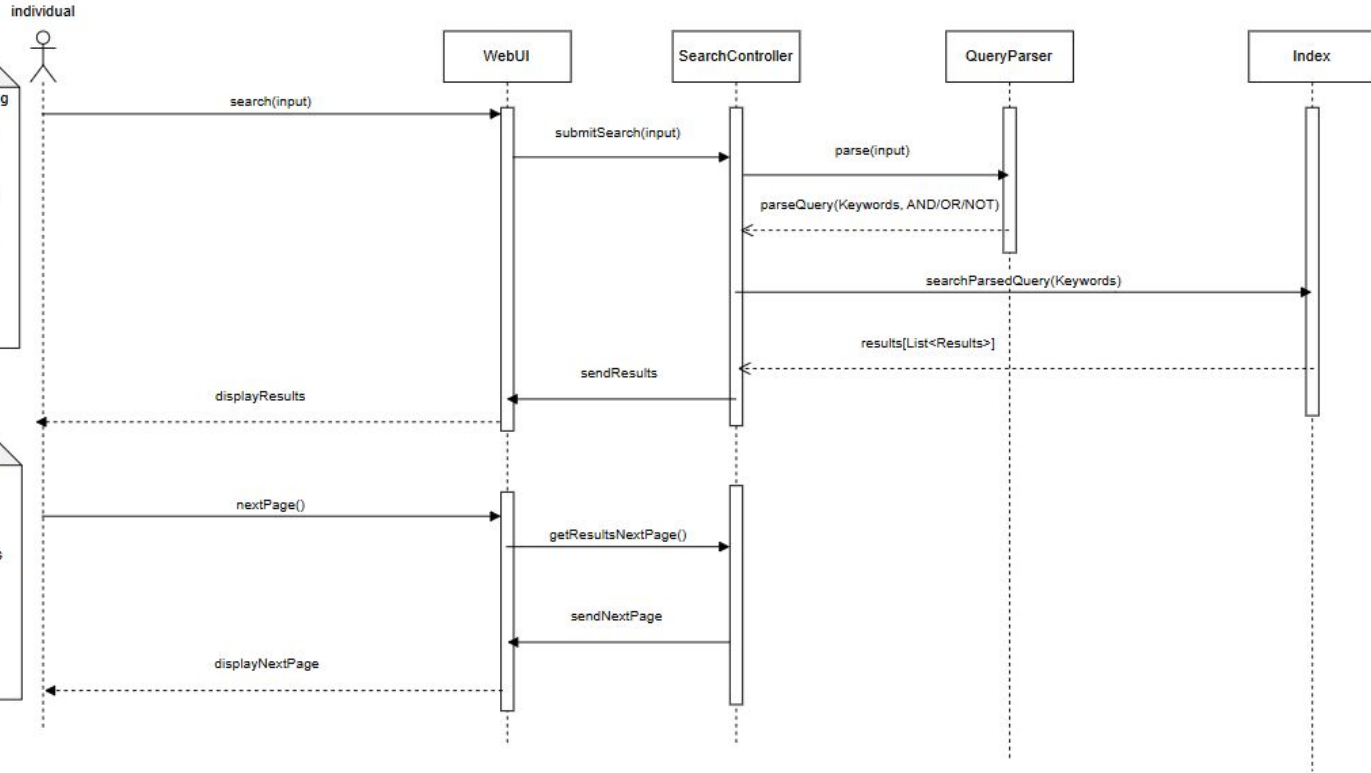
Sequence Diagram

Search Function:

- Accepts the user's raw query string
- Tokenizes and parses keywords, phrases, and Boolean operators
- Retrieves matching postings from the inverted index
- Computes relevance scores and sorts the results
- Returns a ranked list of search results

Navigate Pages:

- Shows pagination controls (page numbers, next/prev) below the results
- User can set the number of results per page
- Highlights the current page
- User clicks a control to load that page's results



Demo

<https://cs-4376-assignment.vercel.app/>

Cyberminer

Search

Enter a search term to get started. Try searching for "web", "JavaScript", or "CSS".

Cyberminer User Manual

Jean-LuckyPicard edited this page 19 hours ago · [11 revisions](#)

Edit New page

1. Summary

Cyberminer is a simple web search engine that shows results from a specific search inputted and how often its keywords appear.

2. Accessing the Engine

The user can access the web search engine by the URL ---

3. Interface

These will be present in the current model of our search engine:

Element	Description
Search box	Type your query here
Search button	Click to execute your search(or press enter)

4. Usage

1. Enter and Run Search

- Specify search mode (**AND**, **OR**, **NOT**).
- Type one or more keywords or exact phrases (use quotes) in the Search box.
- Press **Enter** or click **Search**. Cyberminer will parse your input, query the index, and rank matching documents.

2. Browse Results

Each page shows:

- Title
- URL
- Description

3. Navigate Results

- Use the pagination controls at the bottom of the results to move to the next or previous page, or jump directly to a specific page number.

Pages 2

Find a page...

Home

Cyberminer User Manual

- Summary
- Accessing the Engine
- Interface
- Usage

+ Add a custom sidebar

Clone this wiki locally

<https://github.com/s616627/CS-4376-a>



Link to github repository:

<https://github.com/s616627/CS-4376-assignment/wiki>

Questions?