Milestone 2

CS 6460

Team Awesome

Overview

Connect to a research paper database to allow an AI to read over the papers and return values that will be stored in a separate database.

Use AI to interpret a researcher’s search criteria and pull papers that are relevant.

Allow a user to upvote/downvote papers, which will help teach the AI to return better results over time for an individual's queries.

Allow researchers to have the ability to log in, store papers, and restore previous sessions.

What has been achieved since last milestone?

Updated the Loader service to not crash when encountering errors.

Incorporated the Loader service to use more of the tables in the database instead of just a single table.

Added trigger to database to generate concepts that were not currently in the database.

Developed REST service to query the database for concepts and returns JSON sorted by relevance of the papers to the search query.

Developed early frontend layout.

Demo

Please click on the link below to view the demo, it is still in development so we would appreciate feedback :

Link to site

Queries to try (“cancer”, “DNA”, “cancer DNA”, “protein”, “what protein causes cancer”)

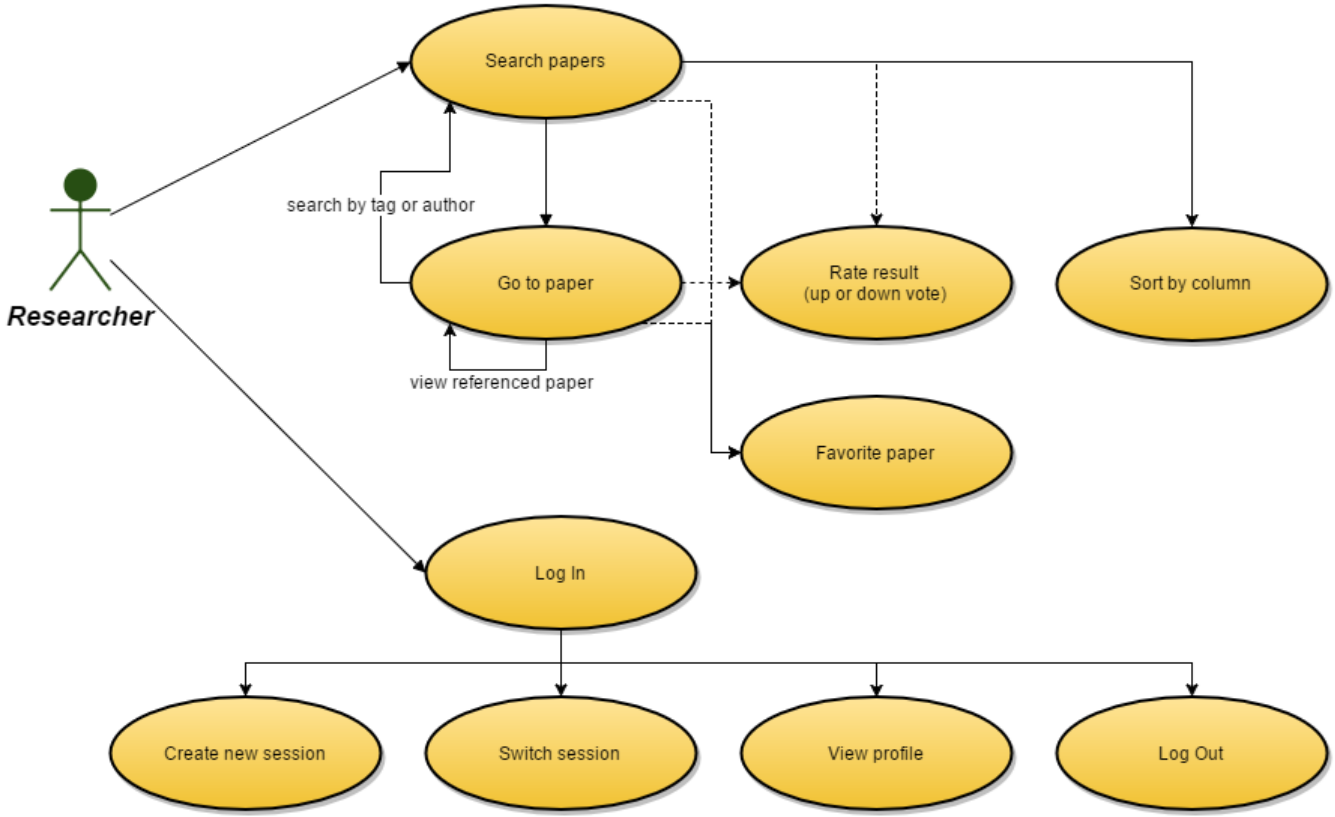
Queries not to try (“humpty dumpty”, “The other green power ranger”, Aaron”)

Use Case Diagram (from milestone 1 presentation)

This diagram displays the possible use-cases through a user interface.

When the user logs in, they have capabilities such as sessions and the ability favorite papers.

Whether logged in or not, the user can search, sort, and view papers over a variety of methods, as well as rate results up or down.



Details on the frameworks

- The backend was created using :
  - NodeJS (allows running javascript as server side code)
  - Express JS (a webserver framework for Node JS)
  - SQL connector for Node JS (connects to our SQL database)
  - Passport JS (allows using OAuth for various providers)
- The backend provides an api to the front end to :
  - Login (using google OAuth2.0 for now)
  - Search for research papers
  - Rank papers (upcoming)

Tools used

GATech Github to manage the project code : https://github.gatech.edu/dstevens34/6460/

Amazon Web Services for the backend

IBM Watson through IBM BlueMix for AI : https://console.ng.bluemix.net

Springer OpenAccess API for research papers : https://dev.springer.com/

MySQL Workbench to work with the DB

Python to quickly prototype the project's code, all the code can be found in the github repository.

What is left?

Service to let users log in

Allow users to save queries

Improve front end for the users