Florida International University
School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Project Title: Envo Scholar 1.0

Team Members: Bryan Bastida
Andrew Castillo

Product Owner(s): Mark Finlayson

Mentor(s): Masoud Sadjadi

Instructor: Masoud Sadjadi

Abstract

This document presents the information necessary to gain a good understanding of the current system of Envo Scholar. Envo Scholar is an academic search engine that allows users to search for environmental science peer reviewed articles. Users are able to create profiles so that they can save articles and gain access to view their saved articles, along with their search history and click history. Users can leave feedback on the overall user experience and can also browse the Envo ontology. This is version 1.0 of Envo Scholar, therefore this document will talk about how Envo Scholar was built from scratch. This document will contain the user stories that were completed, the user stories that were not completed, the project plan, the system design, and the system validation. With the information provided in the document, one will be able to understand how Envo Scholar works and know what the future plans are for Envo Scholar.

Table of Contents

INTRODUCTION

Environmental Scientists need to find papers in order to further their research. They rely on academic search engines to help them find those papers, but there isn't a search engine solely dedicated to environmental science papers. This is where Envo Scholar comes into play. Envo Scholar is a search engine, which allows users to search for peer reviewed articles on environmental science topics. Users can view the Envo ontology to get a better understanding of what they can search for. Users are also able to create a profile, which allows them to save articles and keep track of articles they have saved, clicked on, and searches they have made.

This is version 1.0 of Envo Scholar, therefore the introduction section will only discuss the current system. The current system section will discuss how the system functions, the implementation of the system, the shortcomings of the system, and what needs to be improved.

Current System

The current system of Envo Scholar is version 1.0. Envo Scholar was built using Angular 6 for the front end, NodeJS and Express as a server for the MongoDB which contains the user profiles and feedback space, and PostgreSQL which contains the articles. The current system allows any user, whether signed in or not, to enter a search. When a search is submitted, the user will be redirected to the "displayresults" page, which displays articles related to the search entry. The article information that is being displayed is coming from an API call to a JSON file, which contains the article information from the PostgreSQL database. On that page, the user is able to do a variety of things.

The user can click on any article title, which will redirect them to a page with more information on the article (This page will be talked about later). For any article, the user can click on View PDF, which will open a new tab with the Science Direct page that contains the PDF of that article. For any article, the user can click on Cite, which would ideally give them the citation for the article, but the citations have not been added to the database so at this moment there is only dummy text. For any article, if a user is logged into their account, the user can click on Save, which will save that article to their profile. For any article, the user can click on Share, which will prompt them with a variety of social media platforms that they can share the link to.

Towards the top of the page, the user is presented with six buttons, each with a different keyword. Clicking on one of these buttons will replace the search that the user has entered with the keyword they have clicked on. At the moment, this is being done with the keywords that are being returned from the database. This is a placeholder for something referred to as "Concepts", which have still not been fully added to the database. Eventually, the database will contain Concepts for each article.

The user is able to leave feedback on the overall user experience by clicking on the Feedback button towards the top of the page. The user is prompted with 2 questions, one that asks for their general opinion, and the other that asks for a rate from a scale of 1-10. More feedback options should be added in the next iteration of Envo Scholar such as allowing users to view the feedback, leaving feedback on each article, and more, which is still up for discussion.

The user can click on "View Results in Microsoft Academic" or "View Results in Semantic Scholar" to cross compare the search they have entered with results from those other search engines.

The last thing on this page is Date Range and Sort. The user can type a date from 1900-2018 to filter the articles by the year they were published. If a user leaves either box empty, it will show articles from the value specified in the From box upto the most recent, or from the earliest upto the value specified in the To box. The user can also sort the articles based on Relevance (default) or Recency (newest to oldest). What should be improved from here is sorting by Recency to also do oldest to newest and when the Clear Filters button is pressed, it should also clear whatever values are in the input boxes.

Earlier in this section, it was mentioned that when a user clicks on an article title they are redirected to a page that contains more information on that article. This is the "articleinfo" page. In this page, the user can see the full abstract of the article as well as all of the authors that wrote the article. From this page the user can also click on View PDF, Cite, Save, and Share, and they will follow the same functionality as in the displayresults page. The page also displays a list of Citations, which as previously mentioned, are not in the database yet so it's only dummy text, and the Concepts being listed are all the keywords that belong to that specific article.

The header bar on each page contains a link called Ontology, which will navigate you to the "ontology.html" page. In this page, there is a search bar with a vertical tree that you can either manually navigate through to browse the ontology, or type something into the search bar to open up the paths to that entry. The user can also use the interactive tree to browse through the ontology. These widgets were taken from the Bioportal website. Provided is the link to the NCBO Widget https://www.bioontology.org/wiki/NCBO_Widgets.

The last features that were added to version 1.0 have to do with users and their accounts. Users are able to create an account by clicking on the Account button on any page. They are redirected to the "login" page where they can either sign in if they already have an account or sign up if they do not. Once a user signs into their account, they can view their user profile. In the user profile, the user can view any articles that they have saved, view any articles they have clicked on (whether saved or not), and view their search history. What needs to be implemented in the next iteration is the ability to edit their own information such as name, email, and password.

User Stories

- NVOS-3 Query Entry
  - As a user, I want to enter a query, so that a result is obtained.

- NVOS-4 Display Results
  - As a user, I want to receive the results from a query input, so that they are able to view the articles related to my search query.

- NVOS-7 Sort and Filter Results
  - As a user, I want to click a button that links to the PDF result, so that I can view the PDF.

- NVOS-9 Data Structure Retrieved
  - As a user, I want to retrieve the data structure from the database which contains the concepts, so that I can view it in the UI to then enhance my search query.

- NVOS-10 Visualization of Text
  - As a developer, I want to create a visualization of the text, so the user can see the results.

- NVOS-13 View and Browse Concepts

  - As a user, I want to view the ontology, so that I can see the concepts that I can find articles on.

- NVOS-14 Ontology Search Function
  - As a user, I want access to a search function for the ontology, so that I can search for a specific concept.

- NVOS-16 Add Markups
  - As a developer, I want to allow the user to add markups to their queries, so that they can put multiple concepts together for a more detailed search.

- NVOS-17 Popular Concepts
  - As a developer, I want to provide the user with popular concepts, so that they are able to see what concepts are used more often for building a query entry.

- NVOS-18 User Specific Concepts
  - As a developer, I want to provide the user with concepts that are specific to their previous searches, so that they don't lose their previous searches.

- NVOS-19 Drag and Drop Concepts
  - As a developer, I want to allow the user to drag and drop the concepts, so that they can create their query entries.

- NVOS-21 Login
  - As a user, I want to login to my account, so that I can have access to my user specific information.

- NVOS-22 Create Profile
  - As a user, I want to create my own profile, so that I can manage my searches and results.

- NVOS-23 View Profile
  - As a user, I want to view my own profile, so that I can see my searches and results.

- NVOS-24 View Saved Articles (merged with View Profile)

  - As a user, I want to click on a saved articles button, so that I can view my saved articles.

- NVOS-25 Personalize Results
  - As a user,I want personalized results, so that my results pertain to my contextual intent based on previous searches.

- NVOS-28 Cross Compare with other Search Engines
  - As a user, I want to cross compare the results obtained with other search engines, so that I can have access to our articles, the other search engines articles, and determine which articles are better for me.

- NVOS-29 View Search History
  - As a user, I want view my search history, so that I can get to previous results faster.

- NVOS-30 View Query History
  - As a developer, I want to allow the user to view their query history, so that they can build queries faster and conveniently.

- NVOS-31 View Result History
  - As a developer, I want to allow the user to view their result history, so that they can go back to any articles they want without having to type in a new query.

- NVOS-32 View Click History
  - As a user, I want to view the articles I have clicked on, to be able to get to them faster if I never saved them.

- NVOS-33 View Feedback History
  - As a developer, I want to allow the user to view their feedback history, so they can view any articles they have left feedback on.

- NVOS-34 Feedback
  - As a user, I want to leave feedback on any article, so that the article can have a rating.

- NVOS-35 Reranking Articles/Results
    - As a user, I want to rerank the articles, so that I can organize my feedback history.

- NVOS-36 General Feedback
    - As a user, I want to provide general feedback, so that the user experience can be improved upon.

- NVOS-39 Article Features
    - As a user, I want to my article to have features such as viewing a PDF version of my article, viewing its citation, or viewing statistics about my article.

- NVOS-40 View PDF
    - As a user, I want to click a button that links to the PDF result, so that the I can view the PDF.

- NVOS-41 View Citations
    - As a user, I want to click a button that displays different citation formats for the article results, so that I can get the citations for the article.

- NVOS-42 View Statistics
    - As a user, I would like to view statistics on my published articles, so that I can know my article popularity etc.

The following section provides the detailed user stories that were implemented in this iteration of the Envo Scholar project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

- NVOS-3 Query Entry
    - **Description**
        - As a user, I want to enter a query, so that a result is obtained.

- **Acceptance Criteria**
    - Open web page
    - Type search query
    - Press search or hit enter
    - Query is obtained
- **Use Case**
    - **Name:**NVOS-3 Query Entry
    - **Actor:**User and Webpage
    - **Preconditions:**Web page is opened
    - **Description <Flow of events>:**
        - User opens EnvoScholar website
        - User types a search query
        - User presses the search button or hits enter on the keyboard
- **Use Case Diagram**
    - Use_Case_Diagram_NVOS-3 Query Entry
- **Sequence Diagram**
    - Sequence_Diagram_NVOS-3 Query Entry
- **Class Diagram**
    - Class_Diagram_NVOS-3 Query Entry
- **Testing**
    - NVOS-3 Query Entry

- NVOS-4 Display Results
    - **Description**
        - As a user, I want to receive the results from a query input, so that they are able to view the articles related to my search query.
    - **Acceptance Criteria**
        - User has written out search query and pressed the search button
        - A new page is opened to display results
        - Database has retrieved articles according to the search criteria and displayed them on the screen
    - **Use Case**
        - **Name**: Display Results

        - **Actor**: User and Database
        - **Preconditions**: User has written out search query and pressed the Search button
        - **Description** <Flow of events>:
            - User types a search query into search bar
            - User clicks the Search button
            - Database uses semantic search algorithm to obtain related articles
            - Each article is displayed to the screen along with buttons for View PDF, View Statistics, and Cite
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-4 Display Results
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-4 Display Results
    - **Class Diagram**
        - Class_Diagram_NVOS-4 Display Results
    - **Testing**
        - NVOS-4 Display Results

- NVOS-7 Sort and Filter Results
    - **Description**
        - As a user, I want to click a button that links to the PDF result, so that I can view the PDF.
    - **Acceptance Criteria**
        - User entered search query
        - Database successfully return articles in display results page
        - Display results page has buttons for sorting by recency
        - Display results page has inputs for filtering by date range
        - Correctly sorts and filters

    - **Use Case**
        - **Name:** Sort and Filter Results
        - **Actor:** User and webpage

        - **Preconditions:** Articles displayed on webpage as well as the filter inputs/ sort buttons
        - **Description <Flow of events>:**
            - User types search query and presses search
            - User navigated to display results page with database retrieved articles
            - User presses the Sort/filter inputs on the display results page
            - Either sorts by recency or by specified date range or both.
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-7 Sort and Filter Results
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-7 Sort and Filter Results
    - **Class Diagram**
        - Class_Diagram_NVOS-7 Sort and Filter Results
    - **Testing**
        - NVOS-7 Sort and Filter Results

- NVOS-9 Data Structure Retrieved
    - **Description**
        - As a user, I want to retrieve the data structure from the database which contains the concepts, so that I can view it in the UI to then enhance my search query.
    - **Acceptance Criteria**
        - User has typed a search query either in the homepage or the displayresults page
        - Http GET request will be made to retrieve information from the database
    - **Use Case**
        - **Name**: Data Structure Retrieved
        - **Actor**: User and Web page
        - **Preconditions**: User has accessed the Envo Scholar website
        - User has entered a search query
        - **Description** <Flow of events>:
            - User accesses the Envo Scholar website

        - Types a search query
        - Http GET request will be made to retrieve information from the database
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-9 Data Structure Retrieved
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-9 Data Structure Retrieved
    - **Class Diagram**
        - Class_Diagram_NVOS-9 Data Structure Retrieved
    - **Testing**
        - NVOS-9-Data-Structure-Retrieved

- NVOS-13 View and Browse Concepts
    - **Description**
        - As a user, I want to view the ontology, so that I can see the concepts that I can find articles on.
    - **Acceptance Criteria**
        - User has pressed on the Ontology tab from the homepage
        - The ontology page opens and the user can now browse through the concepts
    - **Use Case**
        - **Name**: View and Browse Concepts
        - **Actor**: User and Web page
        - **Preconditions**: User has accessed the Envo Scholar website
        - **Description** <Flow of events>:
            - User accesses the Envo Scholar website
            - User clicks on the Ontology tab
            - Page changes from homepage to ontology
            - Ontology shows the concepts and the user can navigate through it by expanding the tree
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-13 View and Browse Concepts
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-13 View and Browse Concepts

- **Class Diagram**
  - Class_Diagram_NVOS-13 View and Browse Concepts
- **Testing**
  - NVOS-13-View-and-Browse-Concepts

- NVOS-14 Ontology Search Function
  - **Description**
    - As a user, I want access to a search function for the ontology, so that I can search for a specific concept.
  - **Acceptance Criteria**
    - User has pressed on the Ontology tab from the homepage
    - The ontology page opens
    - The user types the concept he/she is looking for in the search bar and the tree expands to the concept they typed.
  - **Use Case**
    - **Name**: Ontology Search Function
    - **Actor**: User and Web page
    - **Preconditions**:
      - User has accessed the Envo Scholar website
      - User has clicked on the Ontology tab
    - **Description** <Flow of events>:
      - User accesses the Envo Scholar website
      - User clicks on the Ontology tab
      - Page changes from homepage to ontology
      - User types a concept into the search bar
      - The vertical tree will expand to the concept they searched for
  - **Use Case Diagram**
    - Use_Case_Diagram_NVOS-14 Ontology Search Function
  - **Sequence Diagram**
    - Sequence_Diagram_NVOS-14 Ontology Search Function
  - **Class Diagram**
    - Class_Diagram_NVOS-14 Ontology Search Function
  - **Testing**
    - NVOS-14-Ontology-Search-Function

- NVOS-21 Login
    - **Description**
        - As a user, I want to login to my account, so that I can have access to my user specific information.
    - **Acceptance Criteria**
        - User has accessed website
        - User has pressed the Account button
        - User is taken to a new page with login credentials
        - User fills out out their login credentials then presses the Sign In button
    - **Use Case**
        - **Name**: Login
        - **Actor**: User and Website
        - **Preconditions**:
            - User has accessed the website
            - User has pressed the Account button
        - **Description** <Flow of events>:
            - User accesses the Envo Scholar website
            - User presses the Account button
            - User types in e-mail and password
            - User presses Sign In
    - **Use Case Diagram**
        - **U**se_Case_Diagram_NVOS-21 Login
    - **Sequence Diagram**
        - Sequence_NVOS-21 Login
    - **Class Diagram**
        - Class_NVOS-21 Login
    - **Testing**
        - NVOS-21-Login

- NVOS-22 Create Profile
    - Description
        - As a user, I want to create my own profile, so that I can manage my searches and results.

- **Acceptance Criteria**
    - User clicks Account
    - User fills out page with their credentials
    - User clicks Sign Up
- **Use Case**
    - **Name**: Create Profile
    - **Actor**: User, webpage, user server, MongoDB
    - **Preconditions**:
        - User has clicked on Account and is now at the Sign Up page
        - User has entered credentials
    - **Description** <Flow of events>:
        - User clicks on Account and is now at the Sign Up page
        - User adds their full name, e-mail, and password in the input boxes
        - User clicks on sign up
        - User will be displayed a message saying their account has
        - been created
- **Use Case Diagram**
    - Use_Case_Diagram_NVOS-22 Create Profile
- **Sequence Diagram**
    - Sequence_NVOS-22 Create Profile
- **Class Diagram**
    - Class_Diagram_NVOS-22 Create Profile
- **Testing**
    - NVOS-22-Create-Profile1
    - NVOS-22-Create-Profile2

- NVOS-24 View Saved Articles
    - **Description**
        - As a user, I want to click on a saved articles button, so that I can view my saved articles.
    - **Acceptance Criteria**
        - User has logged into their account
        - User has searched for an article

        - User has clicked on the Save button
        - User goes into their profile
        - User profile displays the users saved articles
    - **Use Case**
        - **Name**: View Saved Articles
        - **Actor**: User, Webpage, MongoDB, User server
        - **Preconditions**:
            - User is logged into their account
            - User has searched for an article
            - User has clicked the Save button
        - **Description** <Flow of events>:
            - User logs into their account
            - User searches for an article
            - User clicks on the Save button
            - User clicks on the Account button
            - Saved articles are displayed to the user
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-24 View Saved Articles
    - **Sequence Diagram**
        - Sequence_NVOS-24 View Saved Articles
    - **Class Diagram**
        - Class_NVOS-24 View Saved Articles
    - **Testing**
        - NVOS-24-View-Saved-Articles

- NVOS-28 Cross Compare with other Search Engines
    - **Description**
        - As a user, I want to cross compare the results obtained with other search engines, so that I can have access to our articles, the other search engines articles, and determine which articles are better for me.
    - **Acceptance Criteria**
        - User entered search query
        - Database successfully return articles in display results page

        - Buttons for other search engines given the query and applied filtering
    - **Use Case**
        - **Name:** Cross Compare With Other Search Engines
        - **Actor:** User and webpage
        - **Preconditions:** Articles displayed on webpage as well as the links to other search engine via buttons
        - **Description <Flow of events>:**
            - User types search query and presses search
            - User navigated to display results page with database retrieved articles
            - User presses the links to other search engines on the display results page
            - Sends user to the other search engines site with the applied filter if any.
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-28 Cross Compare With Other Search Engines
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-28 Cross Compare With Other Search Engines
    - **Class Diagram**
        - Class_Diagram_NVOS-28 Cross Compare With Other Search Engines
    - **Testing**
        - NVOS-28 Cross Compare With Other Search Engines

- NVOS-29 View Search History
    - **Description**
        - As a user, I want view my search history, so that I can get to previous results faster.
    - **Acceptance Criteria**
        - User is logged in
        - User has made at least 1 search
        - User logs into their profile

        - User clicks on the Search History button
    - **Use Case**
        - **Name**: View Search History
        - **Actor**: User, Website, MongoDB, User server
        - **Preconditions**:
            - User is logged in
            - User has made at least 1 search
        - **Description** <Flow of events>:
            - User logs into their account
            - User types a search
            - User goes to their profile
            - User clicks on the Search History button
            - Search History page opens and displays the users search
            - history
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-29 View Search History
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-29 View Search History
    - **Class Diagram**
        - Class_Diagram_NVOS-29 View Search History
    - **Testing**
        - NVOS-29-View-Search-History

- NVOS-32 View Click History
    - **Description**
        - As a user, I want to view the articles I have clicked on, to be able to get to them faster if I never saved them.
    - **Acceptance Criteria**
        - User is logged in
        - User has clicked on the Click History button
    - **Use Case**
        - **Name**: View Click History
        - **Actor**: User, Webpage, User Server, MongoDB

        - **Preconditions**:
            - User is logged in
            - User has searched for an article
            - User has clicked on an article
            - User clicked on the Account button
            - User clicked on the Click History button
        - **Description** <Flow of events>:
            - User clicks on the Account button
            - User logs into their account
            - User searches for an article
            - User clicks on the article
            - User goes to their profile
            - User clicks on the Click History button
            - Website routes to the Click History page displaying the article information of the article the user has clicked on
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-32 View Click History
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-32 View Click History
    - **Class Diagram**
        - Class_Diagram_NVOS-32 View Click History
    - **Testing**
        - NVOS-32-View-Click-History

- NVOS-36 General Feedback
    - **Description**
        - As a user, I want to provide general feedback, so that the user experience can be improved upon.
    - **Acceptance Criteria**
        - User presses feedback button
        - Two windows pop up collection user feedback
        - Backend receives the response
    - **Use Case**

        - **Name:** General Feedback
        - **Actor:** User, webpage, Database
        - **Preconditions:** Articles displayed on webpage as well as the Feedback button
        - **Description <Flow of events>:**
            - User types search query and presses search
            - User navigated to display results page with database retrieved articles
            - User presses Feedback buttons
            - User Responds
            - Sends response to backend.
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-36 General Feedback
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-36 General Feedback
    - **Class Diagram**
        - Class_Diagram_NVOS-36 General Feedback
    - **Testing**
        - NVOS-36 General Feedback

- NVOS-40 View PDF
    - **Description**
        - As a user, I want to click a button that links to the PDF result, so that the I can view the PDF.
    - **Acceptance Criteria**
        - User entered a search query
        - Database successfully returned articles related to the search
        - Article links are displayed on the screen
        - User pressed button to view PDF
        - PDF file opens on a new page
    - **Use Case**
        - **Name**: View PDF
        - **Actor**: User and web page

        - **Preconditions**: Article links are displayed on web page as well is link for View PDF
        - **Description** <Flow of events>:
            - User types a search query and hits the Search button
            - Search query is sent to the database
            - Database returns articles
            - Web page displays article links and link for View PDF
            - User hits the View PDF link
            - The PDF opens on a new window
    - **Use Case Diagram**
        - Use_Case_Diagram_NVOS-40 View PDF
    - **Sequence Diagram**
        - Sequence_Diagram_NVOS-40 View PDF
    - **Class Diagram**
        - Class_Diagram_NVOS-40 View PDF
    - **Testing**
        - NVOS-40-View-PDF

Pending User Stories

- NVOS-10 Visualization of Text
    - As a developer, I want to create a visualization of the text for the concepts, so the user can see the results.

- NVOS-16 Add Markups
    - As a developer, I want to allow the user to add markups to their queries, so that they can put multiple concepts together for a more detailed search.

- NVOS-17 Popular Concepts
    - As a developer, I want to provide the user with popular concepts, so that they are able to see what concepts are used more often for building a query entry.

- NVOS-18 User Specific Concepts
    - As a developer, I want to provide the user with concepts that are specific to their previous searches, so that they don't lose their previous searches.

- NVOS-19 Drag and Drop Concepts
    - As a developer, I want to allow the user to drag and drop the concepts, so that they can create their query entries.

- NVOS-23 View Profile
    - As a user, I want to view my own profile, so that I can see my searches and results.

- NVOS-25 Personalize Results
    - As a user,I want personalized results, so that my results pertain to my contextual intent based on previous searches.

- NVOS-30 View Query History
    - As a developer, I want to allow the user to view their query history, so that they can build queries faster and conveniently.

- NVOS-31 View Result History
    - As a developer, I want to allow the user to view their result history, so that they can go back to any articles they want without having to type in a new query.

- NVOS-33 View Feedback History
    - As a developer, I want to allow the user to view their feedback history, so they can view any articles they have left feedback on.

- NVOS-34 Feedback
    - As a user, I want to leave feedback on any article, so that the article can have a rating.

- NVOS-36 General Feedback
    - As a user, I want to provide general feedback, so that the user experience can be improved upon.

- NVOS-39 Article Features
    - As a user, I want to my article to have features such as viewing a PDF version of my article, viewing its citation, or viewing statistics about my article.

- NVOS-41 View Citations
    - As a user, I want to click a button that displays different citation formats for the article results, so that I can get the citations for the article.

- NVOS-42 View Statistics
    - As a user, I would like to view statistics on my published articles, so that I can know my article popularity etc.

Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

- **Angular 6**
  - Angular 6 was used to build the front end functionality.

- **NodeJS/Express**
  - NodeJS/Express was used to build the server that connects to the Mongo Database.

- **MongoDB**
  - MongoDB was used to create the EnvoScholar database which contains a collection of users (user profile information) and a collection of feedback (user experience feedback).

- **PostgreSQL/Elasticsearch**
  - PostgreSQL was used to create the database that contains the articles.
  - Elasticsearch is used in the HTTP GET request to the database to retrieve the articles.
  - (Maria Presa, a graduate student at FIU, is working on this part.)

- **HTML5/CSS/Bootstrap Studio**
  - HTML, CSS, and Bootstrap Studio were used to style the front end. We first started with Bootstrap Studio to create the basic layout, then exported the files and made any edits directly to the HTML/CSS itself in Visual Studio Code.

- **Passport**

  - Passport was used as the authentication middleware for NodeJS for users logging into their profile.

- **JWT**
  - JWT was used to create the token for login authentication.

- **Visual Studio Code**
  - Visual Studio Code was the IDE that was used.

- **Bitbucket**
  - Bitbucket was used for code management.

- **Mac OS**
  - Both computers that were used to create Envo Scholar were Macbooks. (This should not be a problem for future developers if they do not own a Mac)

Sprints Plan

Sprint 1

Attendees: Bryan Bastida, Andrew Castillo

Start time: 5:00PM

End time: 6:00PM

After discussion, the velocity of the team were estimated to be 80.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results

The team members indicated their willingness to work on the following user stories.

- Andrew Castillo
  - NVOS-3 - Query Entry
- Bryan Bastida
  - NVOS-4 - Display Results
- Andrew Castillo
  - NVOS-7 - Sort and Filter Results

Sprint 2

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team were estimated to be 80.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-21 Login NVOS-21 - Login
- User Story NVOS-22 Create Profile NVOS-22 - Create Profile
- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results

The team members indicated their willingness to work on the following user stories.

- Andrew Castillo
  - NVOS-3 - Query Entry
- Bryan Bastida
  - NVOS-4 - Display Results
- Andrew Castillo
  - NVOS-7 - Sort and Filter Results
- Bryan Bastida
  - NVOS-21 - Login
- Andrew Castillo
  - NVOS-22 - Create Profile

Sprint 3

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team were estimated to be 80.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-13 View and Browse Concepts NVOS-13 - View and Browse Concepts
- User Story NVOS-14 Ontology Search Function NVOS-14 - Ontology Search Function

The team members indicated their willingness to work on the following user stories.

- Bryan Bastida, Andrew Castillo
  - NVOS-13 - View and Browse Concepts
- Bryan Bastida, Andrew Castillo
  - NVOS-14 - Ontology Search Function

Sprint 4

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team were estimated to be 80.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-9 Data Structure Retrieved NVOS-9 - Data Structure Retrieved
- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results

The team members indicated their willingness to work on the following user stories.

- Andrew Castillo
  - NVOS-3 - Query Entry
  - NVOS-7 - Sort and Filter Results

- Bryan Bastida
  - NVOS-4 - Display Results
  - NVOS-9 - Data Structure Retrieved

Sprint 5

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team were estimated to be 80.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results
- User Story NVOS-21 Login NVOS-21 - Login
- User Story NVOS-23 View Profile NVOS-23 - View Profile
- User Story NVOS-25 Personalize Results NVOS-25 - Personalize Results

The team members indicated their willingness to work on the following user stories.

- Bryan Bastida
  - NVOS-4 - Display Results
  - NVOS-21 - Login
  - NVOS-25 - Personalize Results
- Andrew Castillo
  - NVOS-7 - Sort and Filter Results
  - NVOS-23 - View Profile

Sprint 6

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team were estimated to be 66.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results
- User Story NVOS-23 View Profile NVOS-23 - View Profile
- User Story NVOS-28 Cross Compare with other Search Engines NVOS-28 - Cross Compare with other Search Engines
- User Story NVOS-29 View Search History NVOS-29 - View Search History

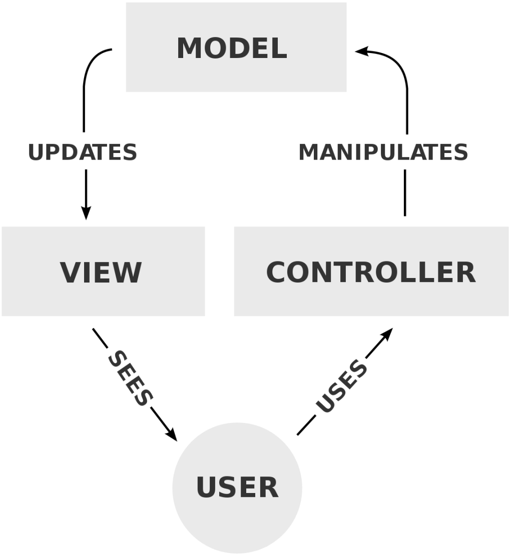
The team members indicated their willingness to work on the following user stories.

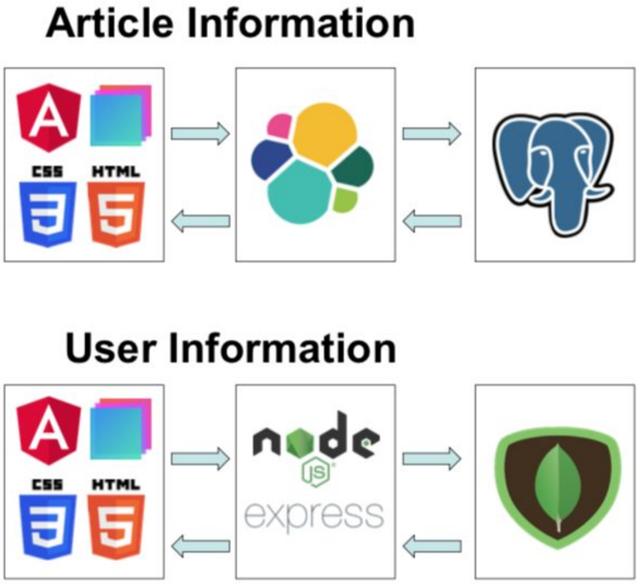
- Bryan Bastida
  - NVOS-4 - Display Results
  - NVOS-29 - View Search History
- Andrew Castillo
  - NVOS-7 - Sort and Filter Results
  - NVOS-23 - View Profile
  - NVOS-28 - Cross Compare with other Search Engines

System Design

This section contains information on the design decisions that went into Envo Scholar. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

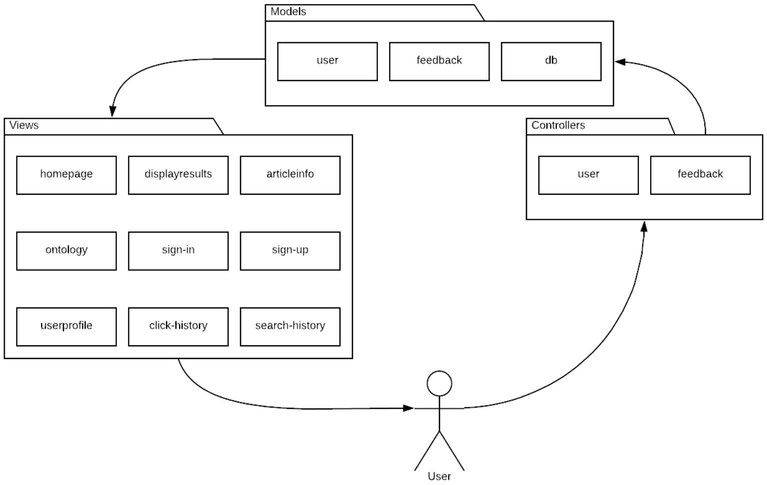
Architectural Patterns



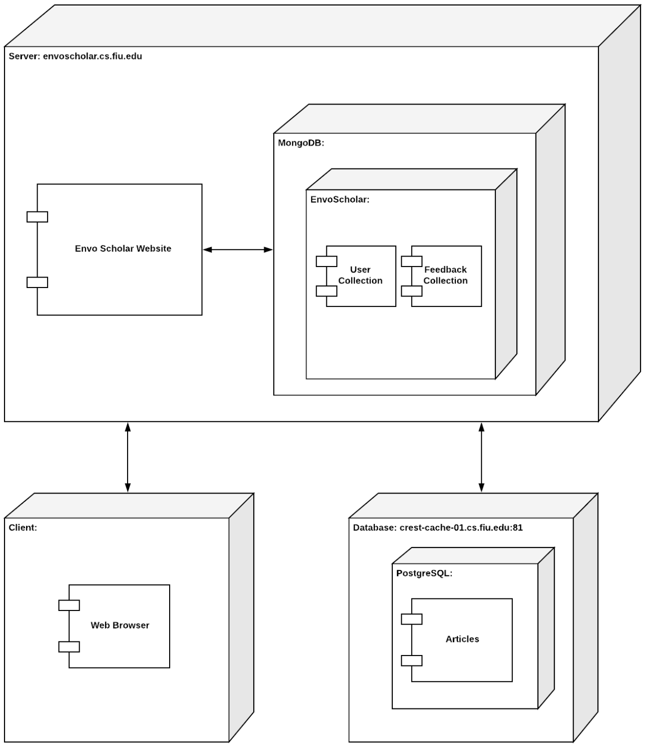


System and Subsystem Decomposition

- **Models:** The models manage the data and the logic of the program. It contains the methods that interact with the data store.
- **Views:** The views display the information to the page so that the user is able to interact with.
- **Controllers:** The controllers accept the input data from the users to be able to send to the models to then update the view.



Deployment Diagram



System Validation

NVOS-3 Query Entry

- Test case ID: NVOS-3-Query-Entry
- Description/Summary of Test: This test was made to check if input is accepted by the search bar
- Pre-condition: A search is entered into the search bar
- Expected Results: Search is accepted
- Actual Result: Search is accepted
- Status (Fail/Pass): Pass

NVOS-4 Display Results

- Test case ID: NVOS-3-Display-Results
- Description/Summary of Test: User has entered a search into the search box and the display page loads the related articles
- Pre-condition: User is either on the home page or in the display results page. User has entered a search query. User has submitted the search.
- Expected Results: Articles related to the search will appear on the screen.
- Actual Result: Articles related to the search will appear on the screen.
- Status (Fail/Pass): Pass

NVOS-9 Data Structure Retrieved

- Test case ID: NVOS-7 Sort and Filter
- Description/Summary of Test: Sort by recency and filter given date ranges
- Pre-condition: Displayed results list given a query
- Expected Results: Successful sorting and filtering
- Actual Result: Successful sorting and filtering

- Status (Fail/Pass): Pass

NVOS-9 Data Structure Retrieved

- Test case ID: NVOS-9-Data-Structure-Retrieved
- Description/Summary of Test: This test was to see if the JSON file was being retrieved from the HTTP GET request
- Pre-condition: A search query has been entered
- Expected Results: JSON file is retrieved
- Actual Result: JSON file is retrieved
- Status (Fail/Pass): Pass

NVOS-13 View and Browse Concepts

- Test case ID: NVOS-13-Browse-and-View-Concepts
- Description/Summary of Test: Ontology is displayed as a tree. User is able to expand nodes to traverse the tree.
- Pre-condition: User has opened the Ontology tab.
- Expected Results: : Ontology is displayed as a tree. User is able to expand nodes to traverse the tree.
- Actual Result: : Ontology is displayed as a tree. User is able to expand nodes to traverse the tree.
- Status (Fail/Pass): Pass

NVOS-14 Ontology Search Function

- Test case ID: NVOS-14-Ontology-Search-Function
- Description/Summary of Test: User types a concept. The tree expands showing the path to that concept.
- Pre-condition: User types a concept.
- Expected Results: The tree expands showing the path to that concept.
- Actual Result: The tree expands showing the path to that concept.

- Status (Fail/Pass): Pass

NVOS-21 Login

- Test case ID: NVOS-21-Login
- Description/Summary of Test: User has clicked on the Sign In button. Login page opens. User enters his/her login credentials then clicks login. User is now logged in.
- Pre-condition: User has clicked on the Account button. Login page opens. User enters his/her login credentials then clicks login.
- Expected Results: User is logged in.
- Actual Result: User is logged in.
- Status (Fail/Pass): Pass

NVOS-22 Create Profile

- Test case ID: NVOS-22-Create-Profile1
- Description/Summary of Test: This test was made to see if the a user profile is created
- Pre-condition: User has filled out name, e-mail, and password input boxes
- Expected Results: User will be prompted with message that says their account has been created and a new user will be added to the Mongo database
- Actual Result: User has been prompted with message that says their account has been created and a new user has been added to the Mongo database
- Status (Fail/Pass): Pass

- Test case ID: NVOS-22-Create-Profile2
- Description/Summary of Test: This test was made to see if the a user is prompted with message saying that another account has the same e-mail
- Pre-condition: User has filled out e-mail and password input boxes
- Expected Results: User will be prompted with message that says an account already exists with that e-mail address
- Actual Result: User has been prompted with message that says an account already exists with that e-mail address
- Status (Fail/Pass): Pass

NVOS-24 View Saved Articles

- Test case ID: NVOS-24-View-Saved-Articles
- Description/Summary of Test: This test was done to check if a user can view their saved articles
- Pre-condition: User is logged in. User has entered a search. User has clicked on the save button. User accesses their profile page
- Expected Results: Saved articles are displayed in the user profile
- Actual Result: Saved articles are displayed in the user profile
- Status (Fail/Pass): Pass

NVOS-29 View Search History

- Test case ID: NVOS-28 Cross Compare with Other Search Engines
- Description/Summary of Test: Opens new window to desired website
- Pre-condition: Displayed results list given, filter applied
- Expected Results: Successful navigation to alternate search engine with applied filter
- Actual Result: Successful navigation to alternate search engine with applied filter
- Status (Fail/Pass): Pass

NVOS-29 View Search History

- Test case ID: NVOS-29-View-Search-History
- Description/Summary of Test: This test was made to check that a user is able to view their search history
- Pre-condition: User is logged into their account. User has made at least 1 search. User goes to their profile page. User clicks on Search History
- Expected Results: Search history is displayed
- Actual Result: Search history is displayed

- Status (Fail/Pass): Pass

NVOS-32 View Click History

- Test case ID: NVOS-32-View-Click-History
- Description/Summary of Test: This test was made to check that a user is able to view their click history
- Pre-condition: User is logged into their account. User has searched for an article. User has clicked on the article. User goes to their profile page. User clicks on Click History
- Expected Results: Click history is displayed
- Actual Result: Click history is displayed
- Status (Fail/Pass): Pass

NVOS-36 General Feedback

- Test case ID: NVOS-36 General Feedback
- Description/Summary of Test: Opens 2 prompts
- Pre-condition: Displayed results list given
- Expected Results: Successful storage of user feedback
- Actual Result: Successful storage of user feedback
- Status (Fail/Pass): Pass

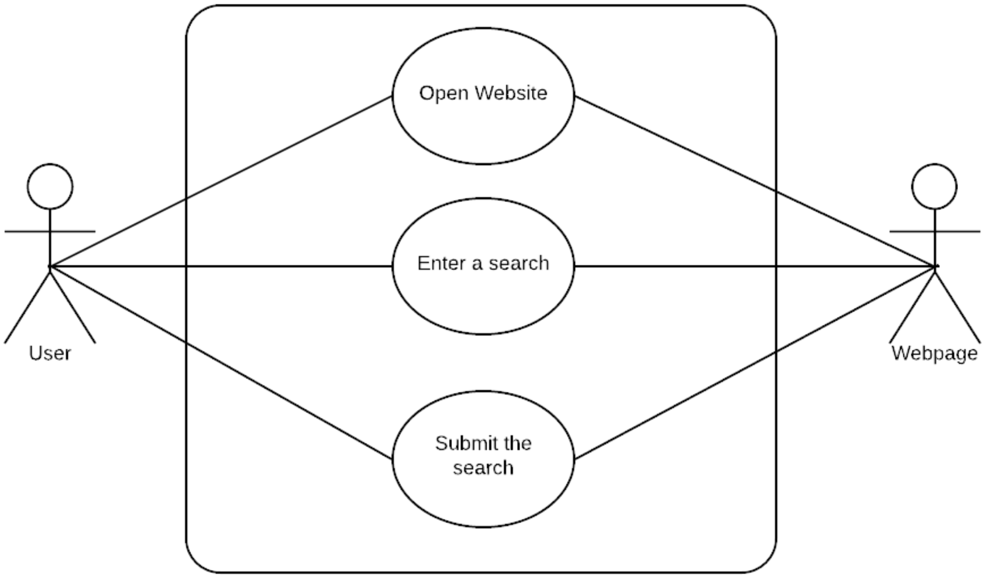
NVOS-40 View PDF

- Test case ID: NVOS-40-View-PDF
- Description/Summary of Test: This test was to check that a PDF page will open when the user clicks on View PDF.
- Pre-condition: User has searched for an article. User is now on the displayresults page and is hovering over the View PDF link on an article. User clicks on the View PDF.
- Expected Results: Science Direct page with PDF of that article opens on a new tab.
- Actual Result: Science Direct page with PDF of that article opened on a new tab.
- Status (Fail/Pass): Pass

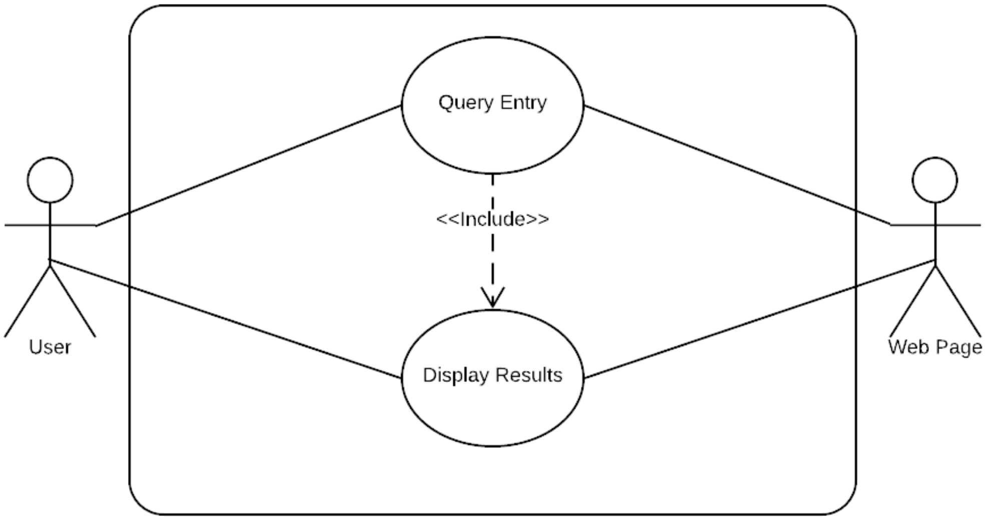
APPENDIX

Appendix A - Use Case Diagrams

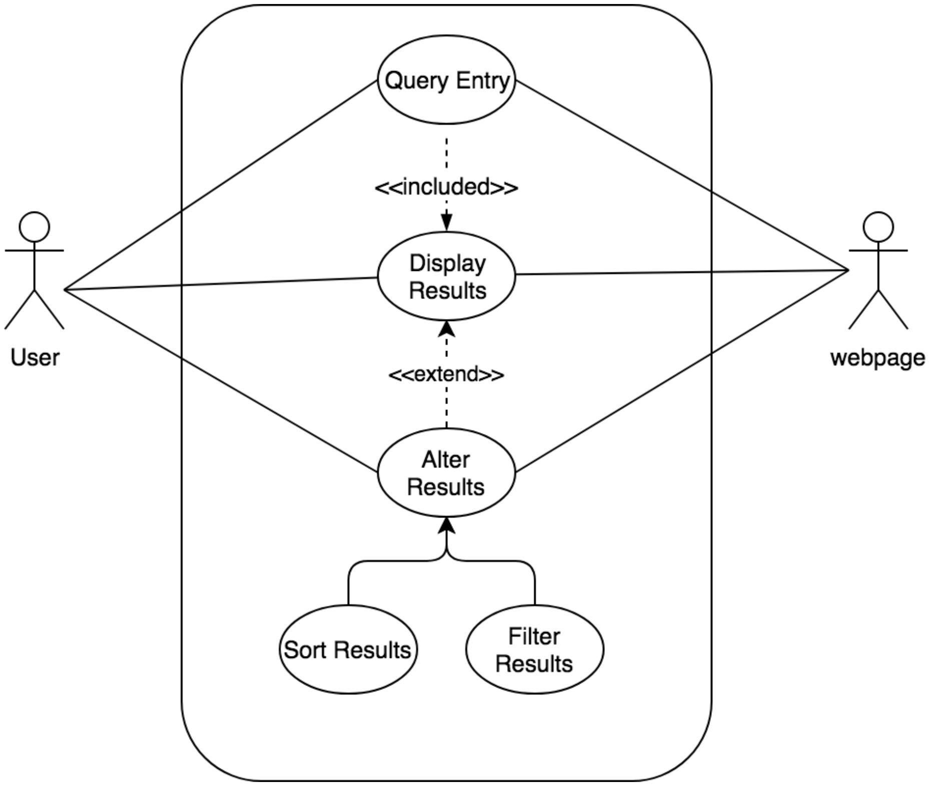
Use_Case_Diagram_NVOS-3 Query Entry



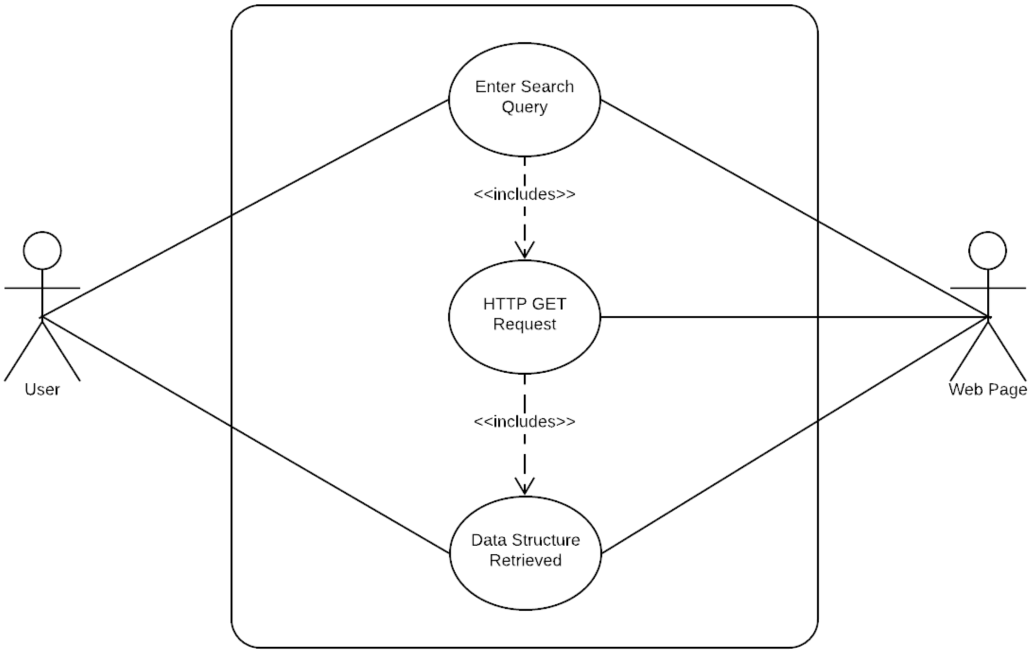
Use_Case_Diagram_NVOS-4 Display Results



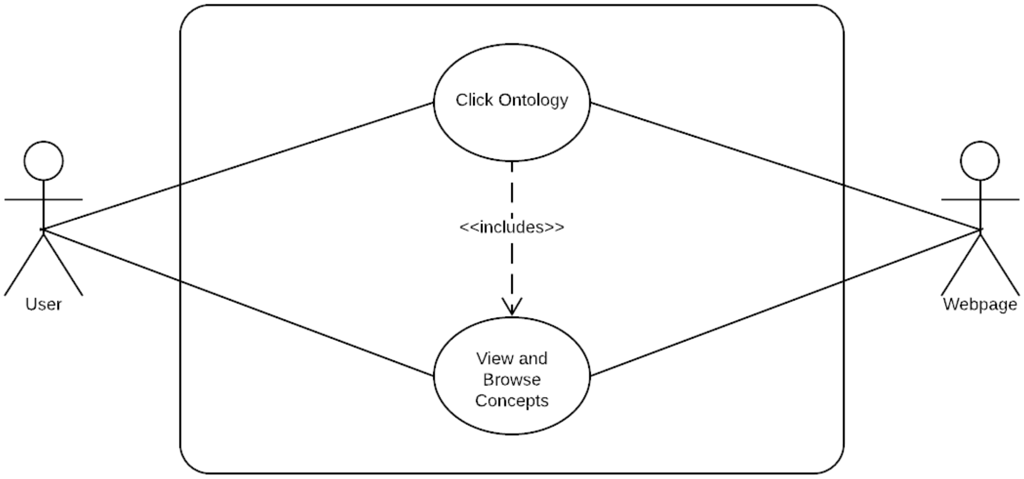
Use_Case_Diagram_NVOS-7 Sort and Filter Results



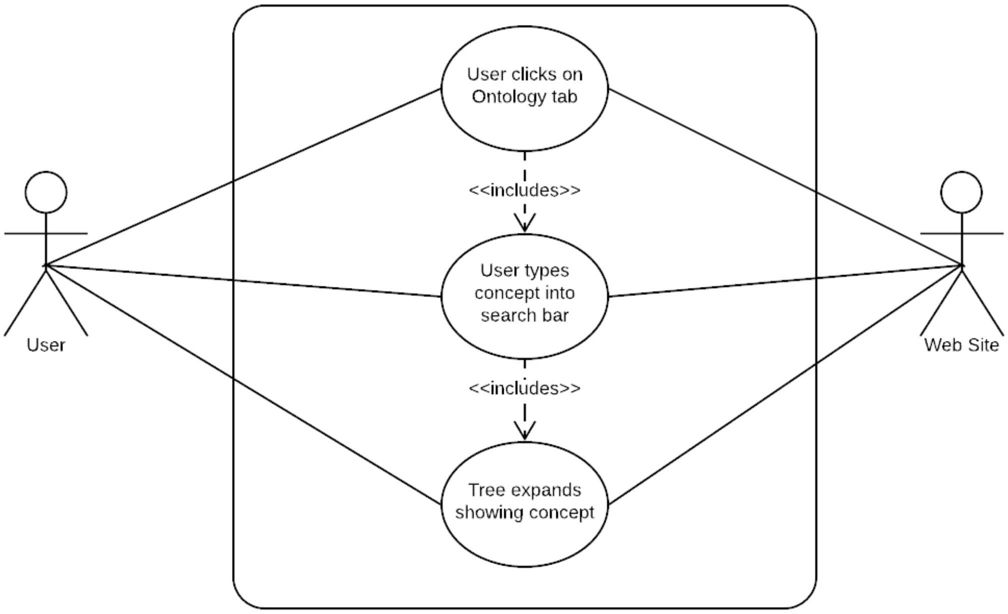
Use_Case_Diagram_NVOS-9 Data Structure Retrieved



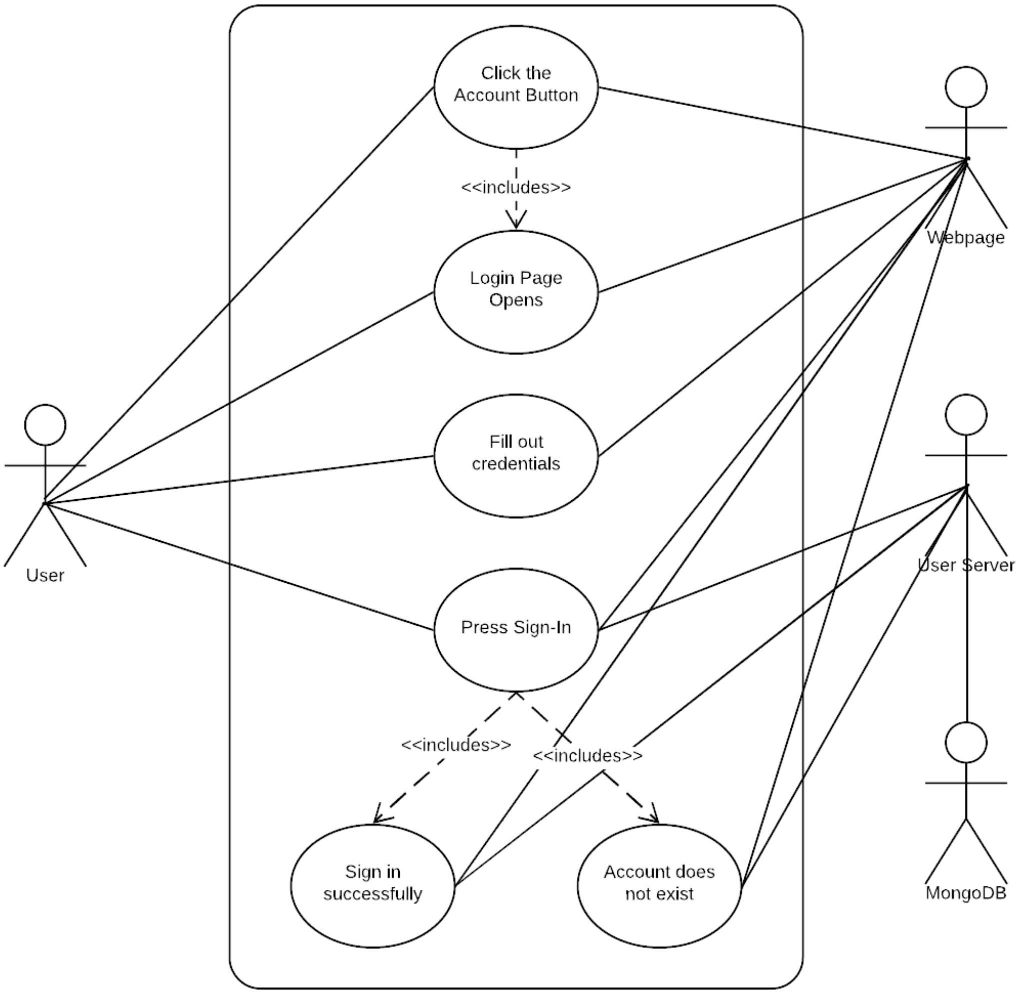
Use_Case_Diagram_NVOS-13 View and Browse Concepts



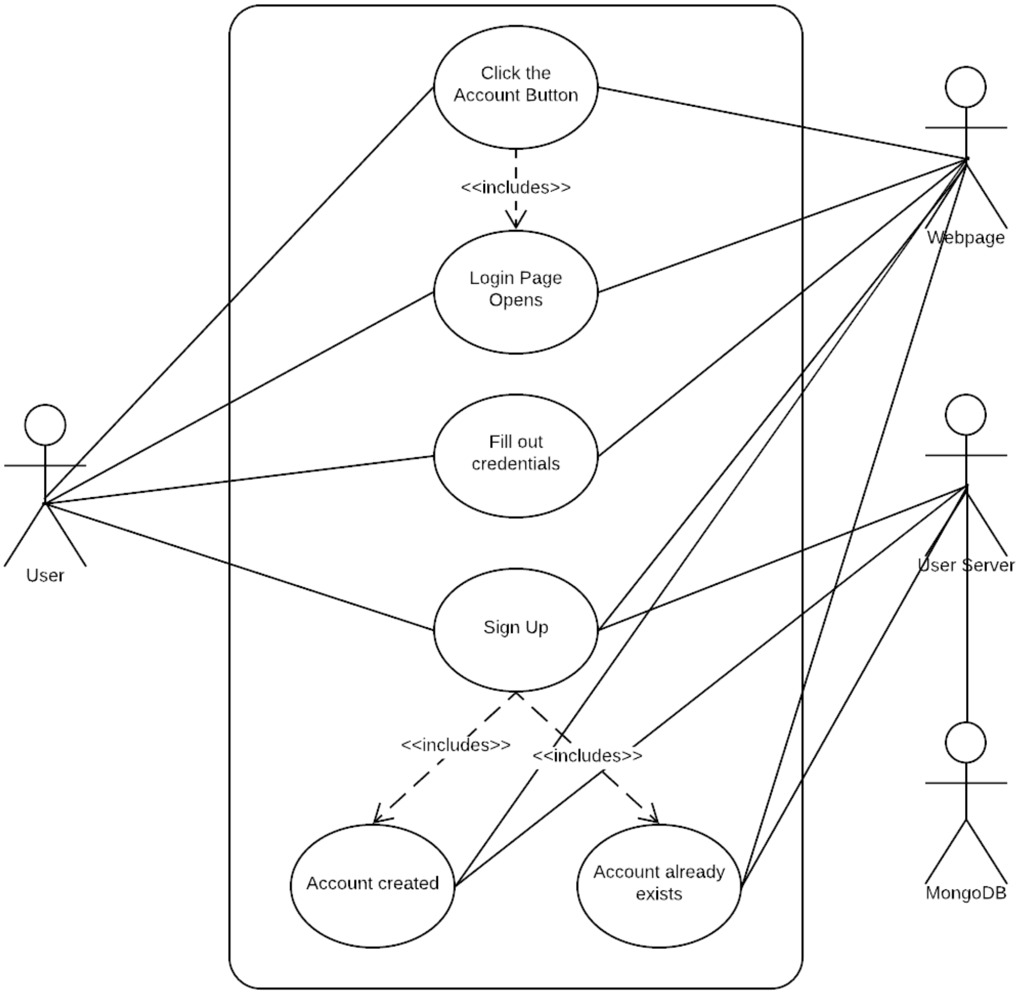
Use_Case_Diagram_NVOS-14 Ontology Search Function


User clicks on Ontology tab
<<includes>>
User types concept into search bar
<<includes>>
Tree expands showing concept
User
Web Site

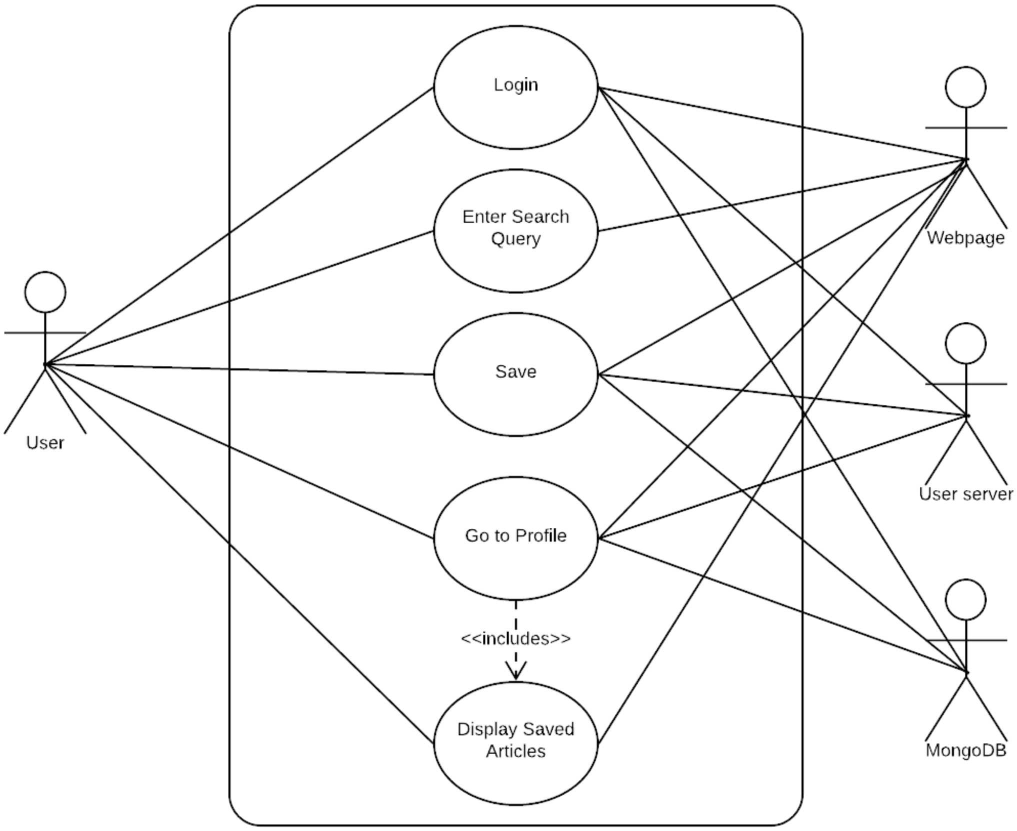
Use_Case_Diagram_NVOS-21 Login



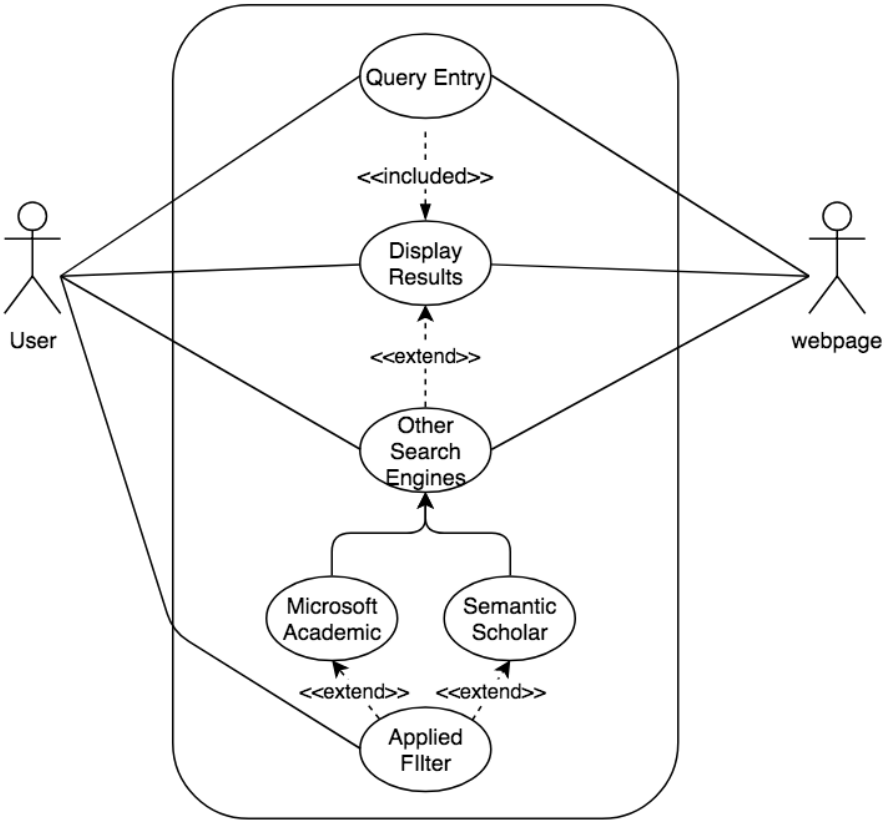
Use_Case_Diagram_NVOS-22 Create Profile



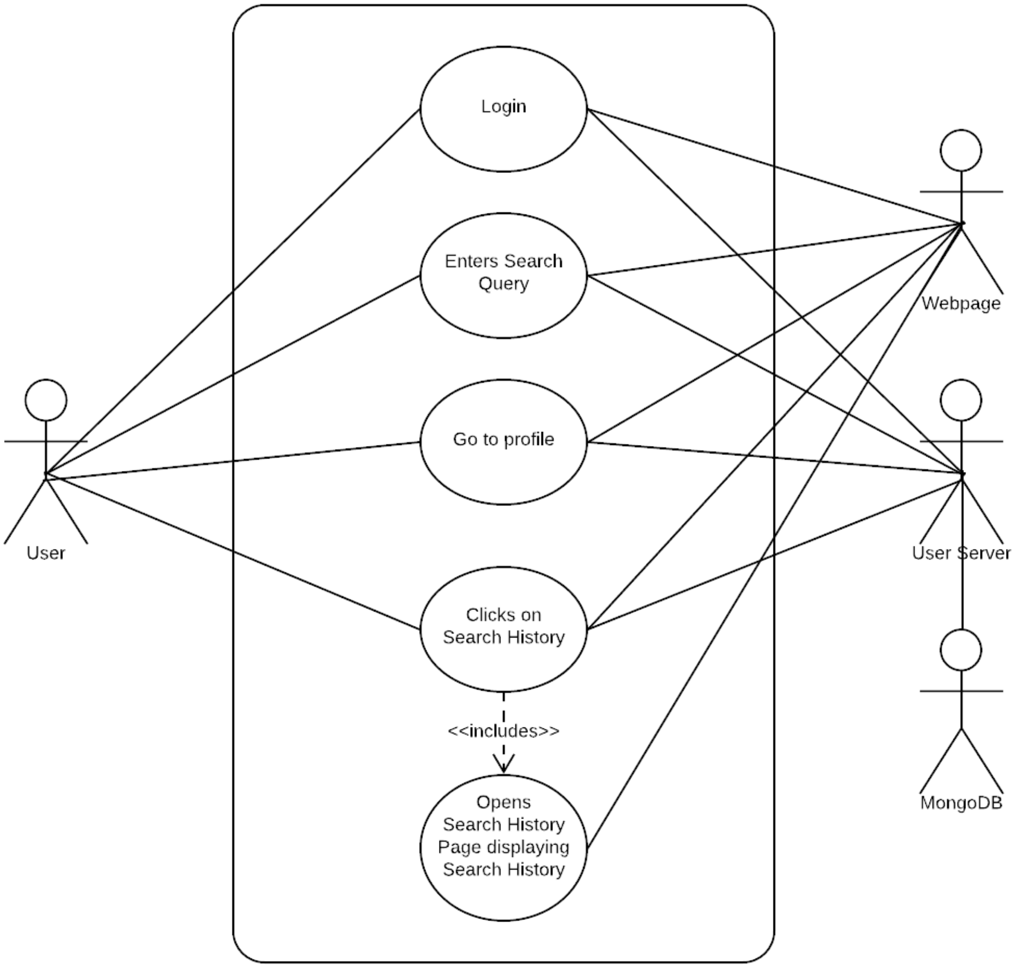
Use_Case_Diagram_NVOS-24 View Saved Articles



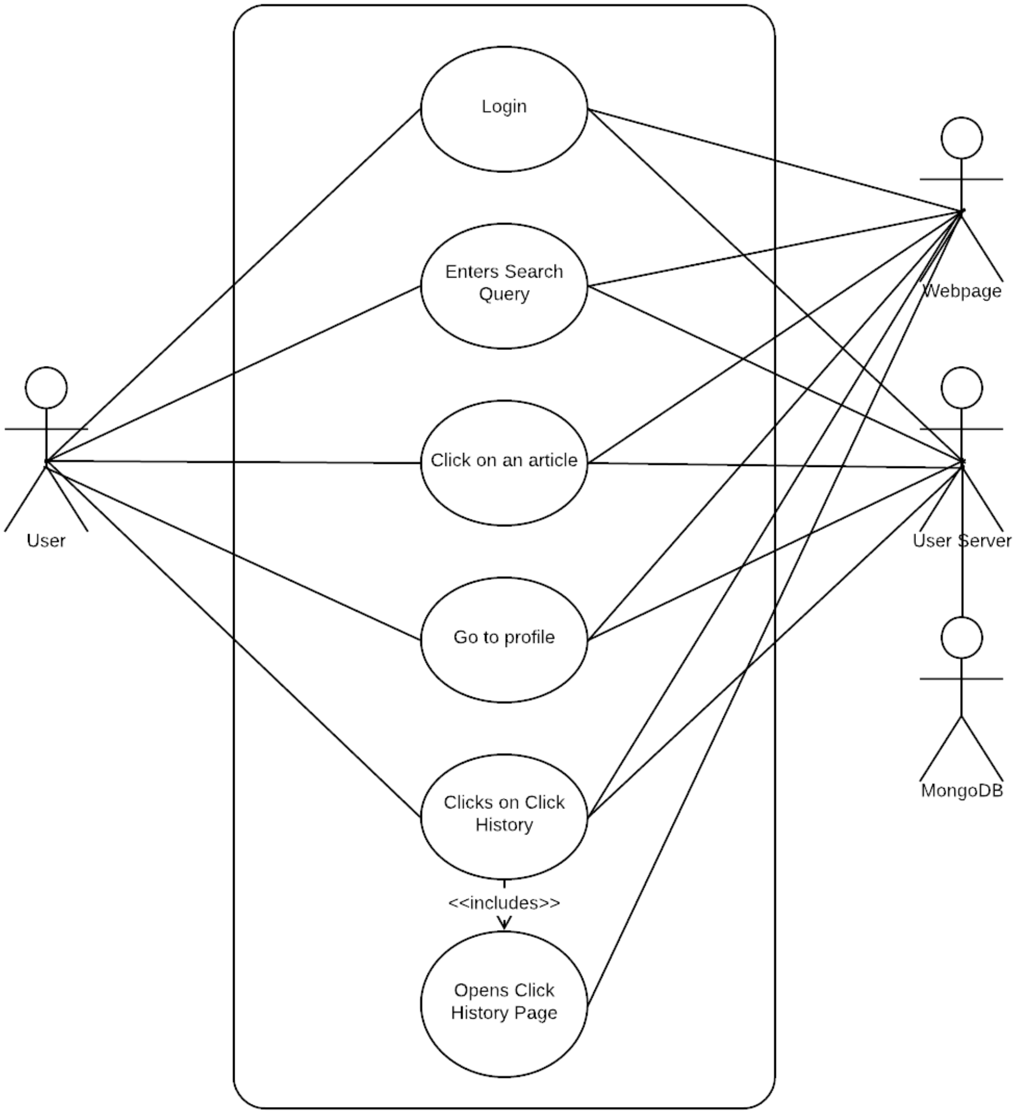
Use_Case_Diagram_NVOS-28 Cross Compare With Other Search Engines



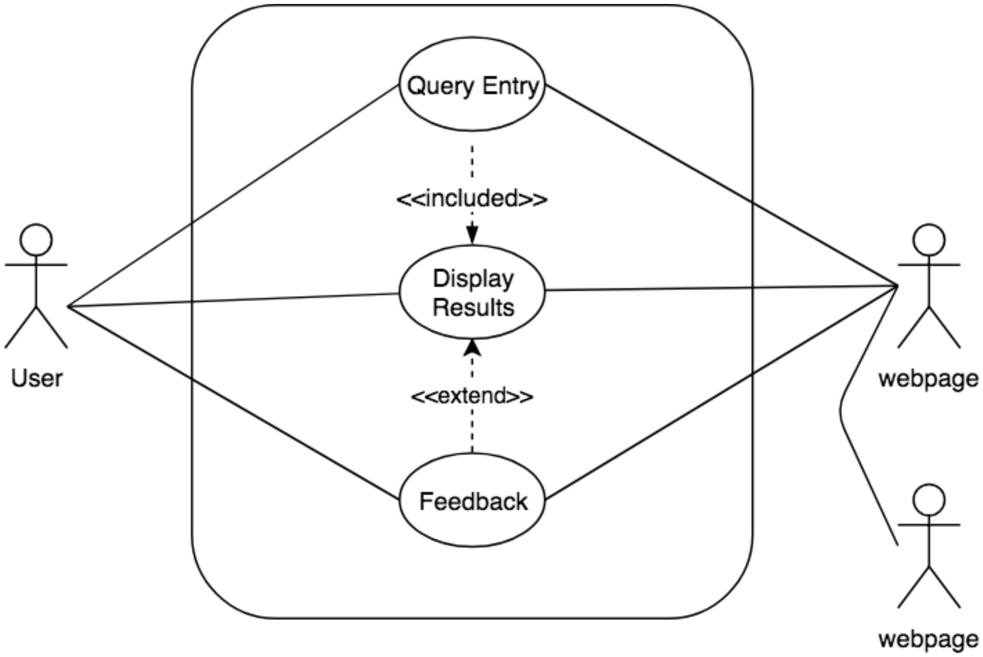
Use_Case_Diagram_NVOS-29 View Search History



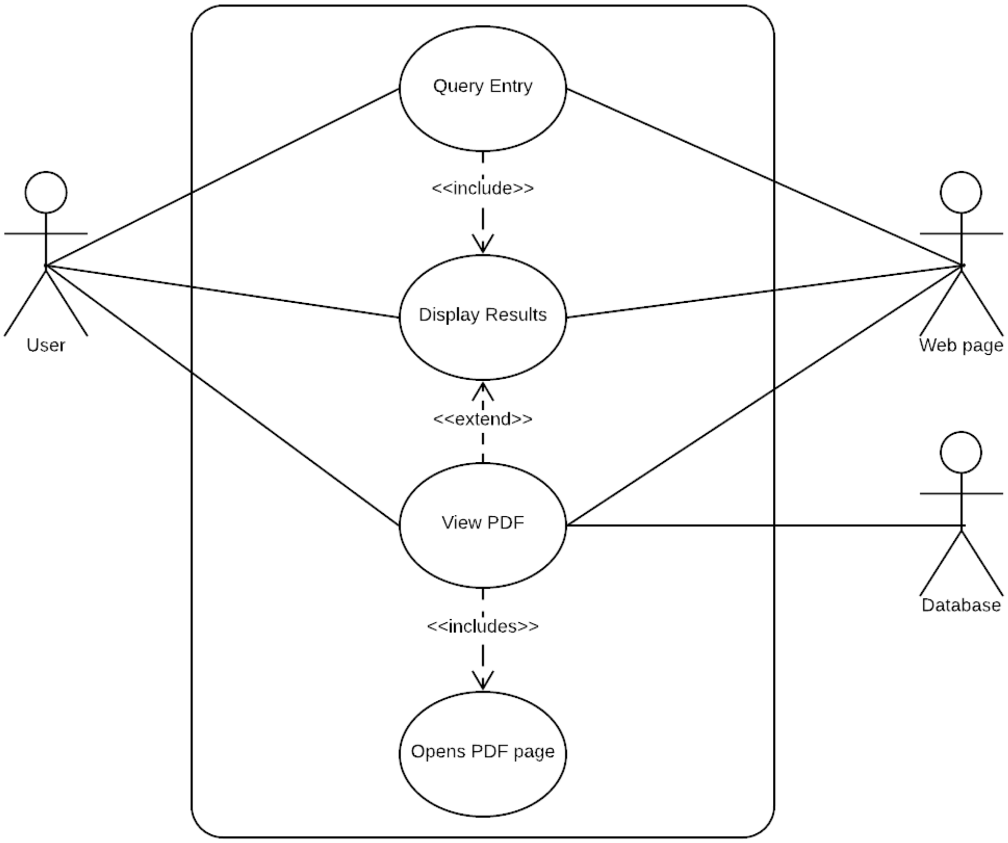
Use_Case_Diagram_NVOS-32 View Click History



Use_Case_Diagram_NVOS-36 General Feedback

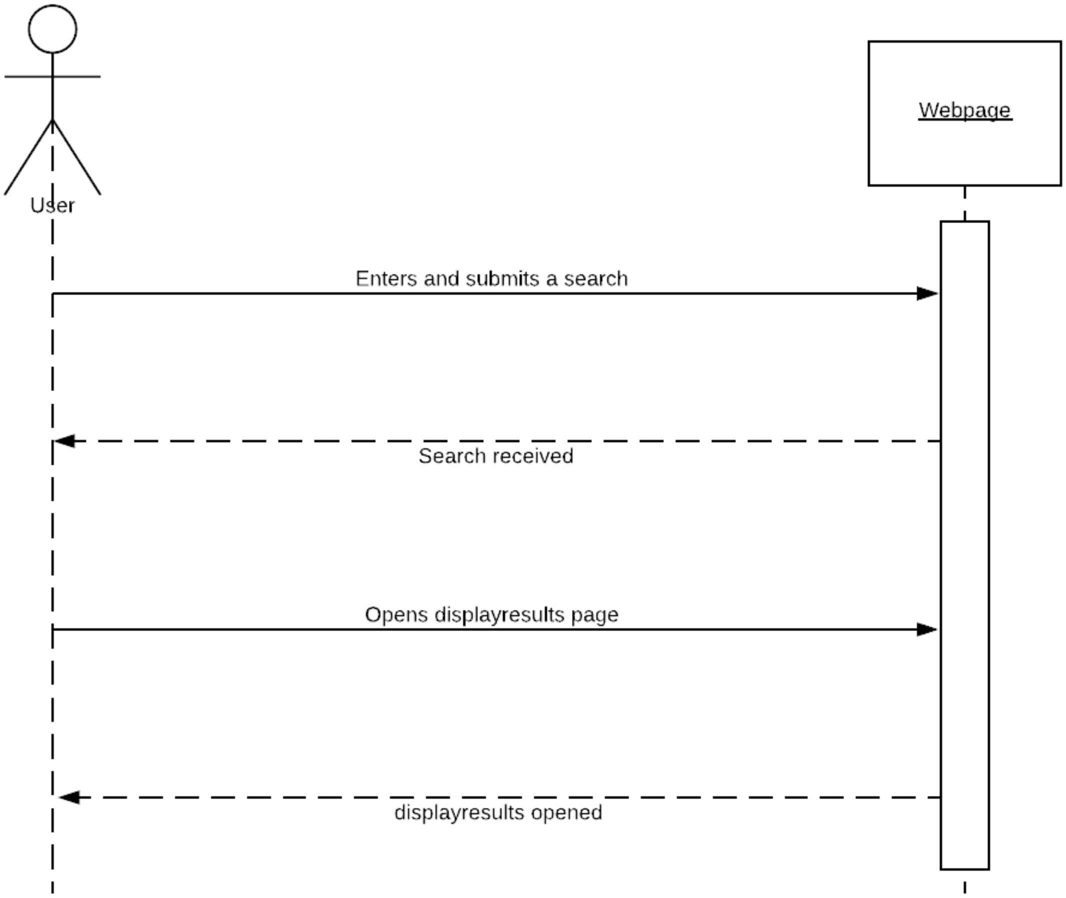


Use_Case_Diagram_NVOS-40 View PDF

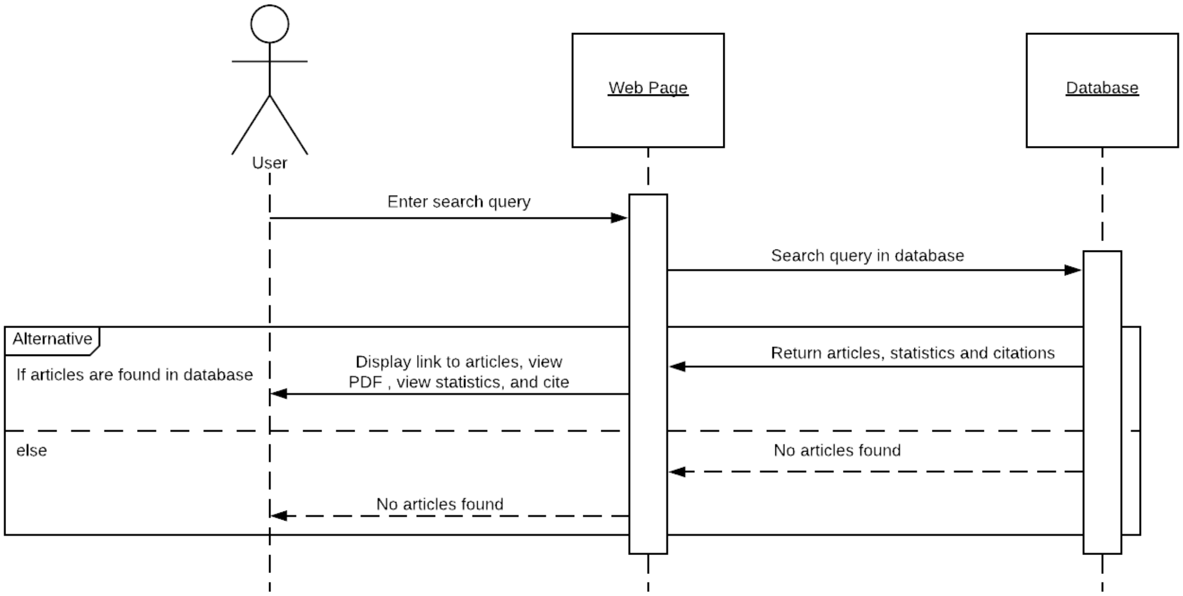


Appendix A - Sequence Diagrams

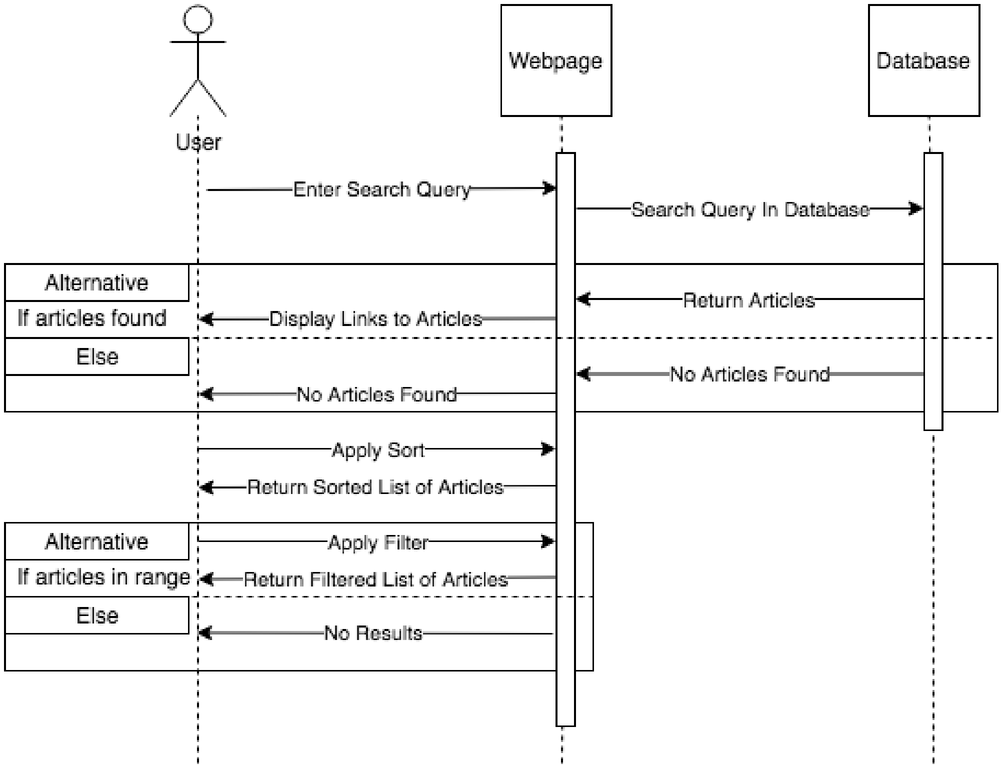
Sequence_Diagram_NVOS-3 Query Entry



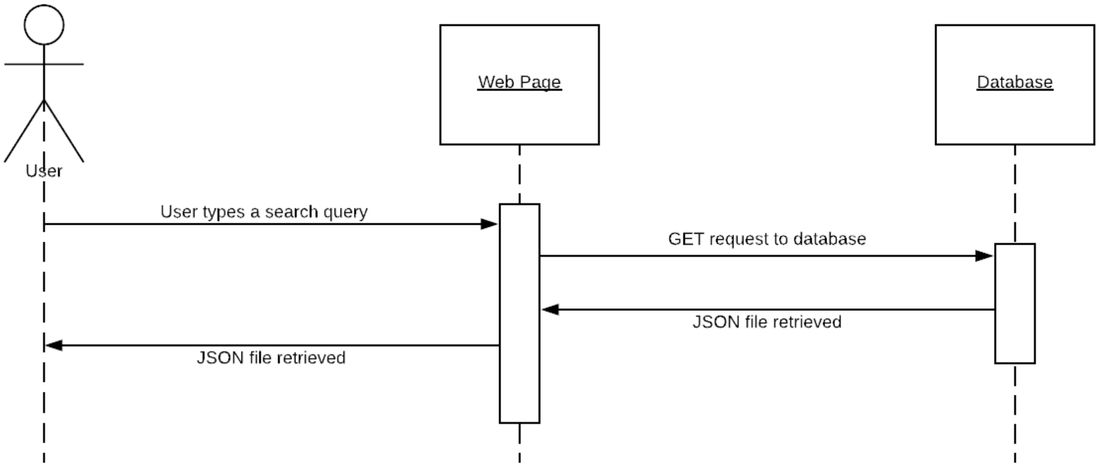
Sequence_Diagram_NVOS-4 Display Results


Web Page
Database
User
Enter search query
Search query in database
Alternative
If articles are found in database
Return articles, statistics and citations
Display link to articles, view PDF , view statistics, and cite
else
No articles found
No articles found

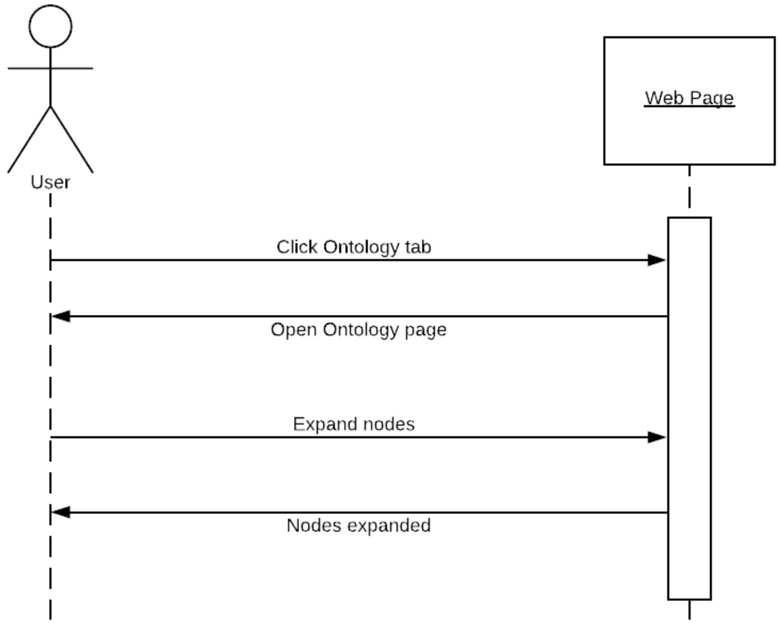
Sequence_Diagram_NVOS-7 Sort and Filter Results



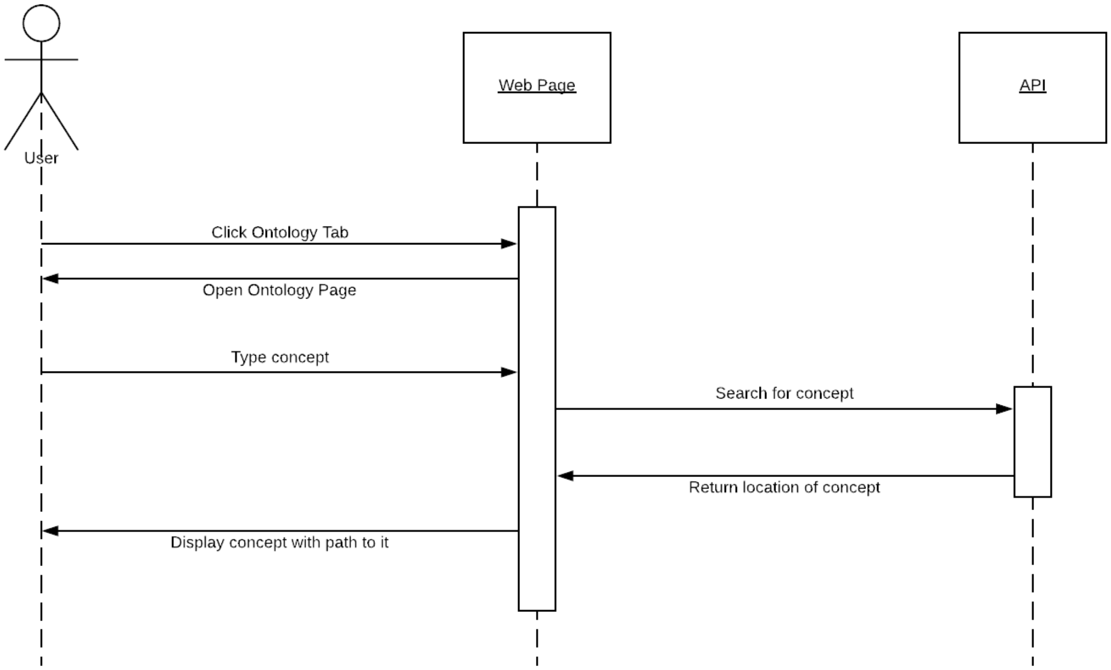
Sequence_Diagram_NVOS-9 Data Structure Retrieved



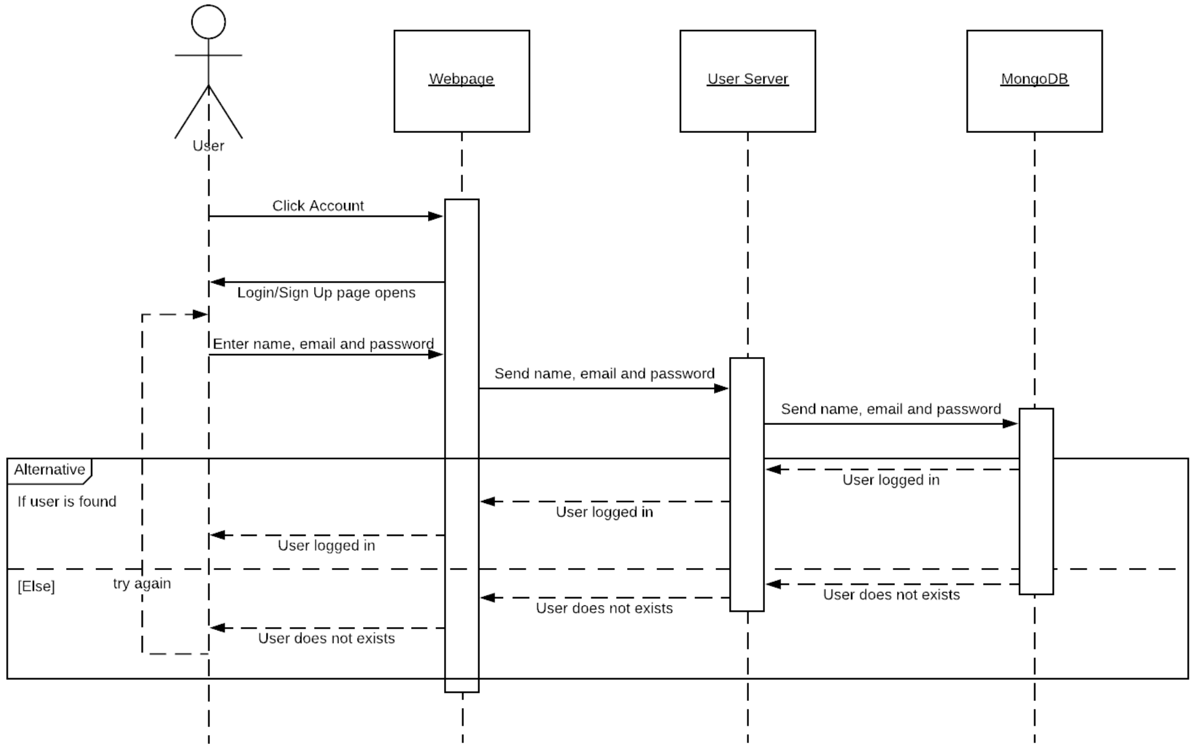
Sequence_Diagram_NVOS-13 View and Browse Concepts



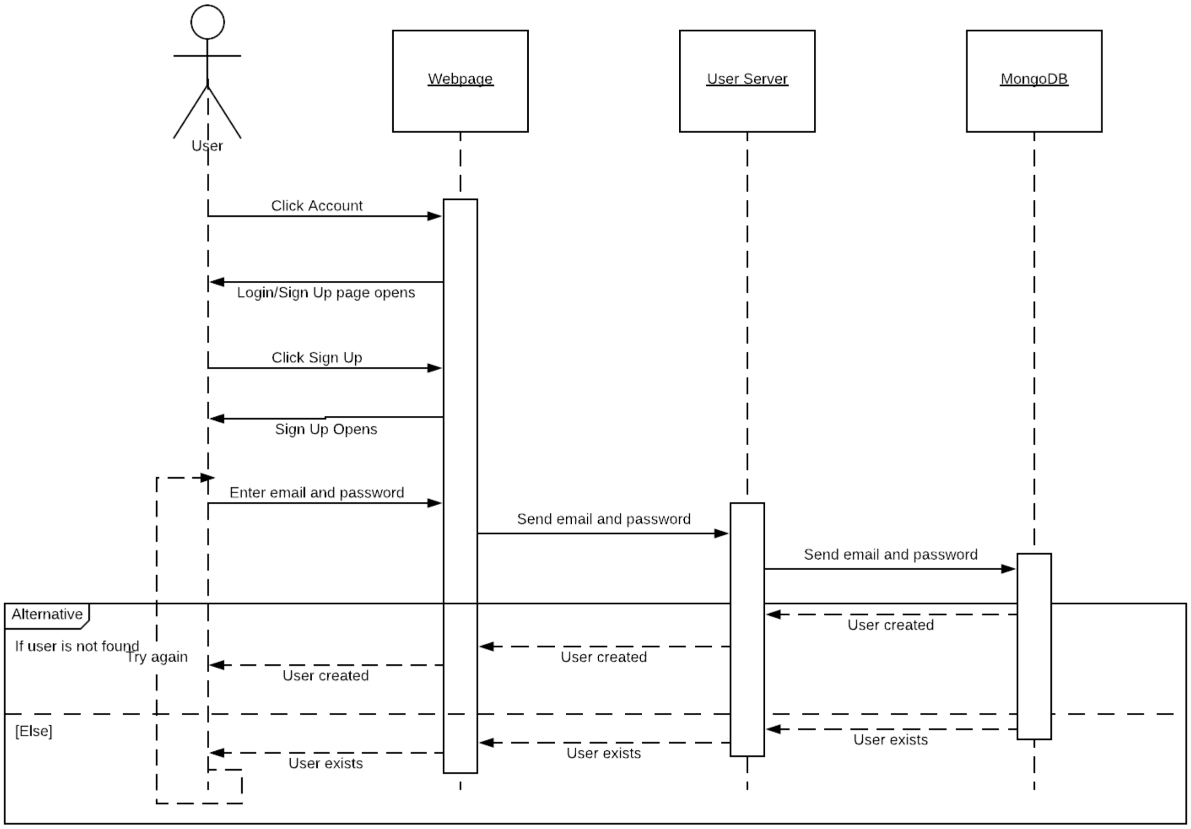
Sequence_Diagram_NVOS-14 Ontology Search Function



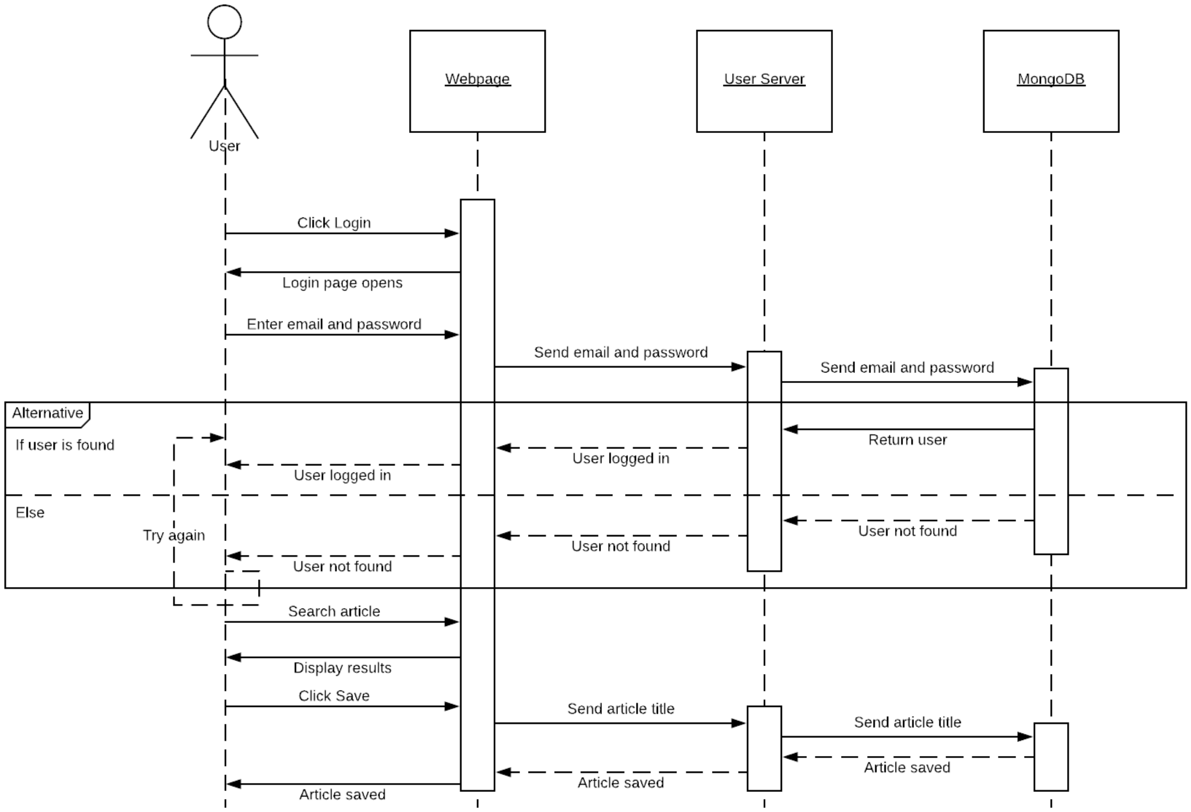
Sequence_Diagram_NVOS-21 Login



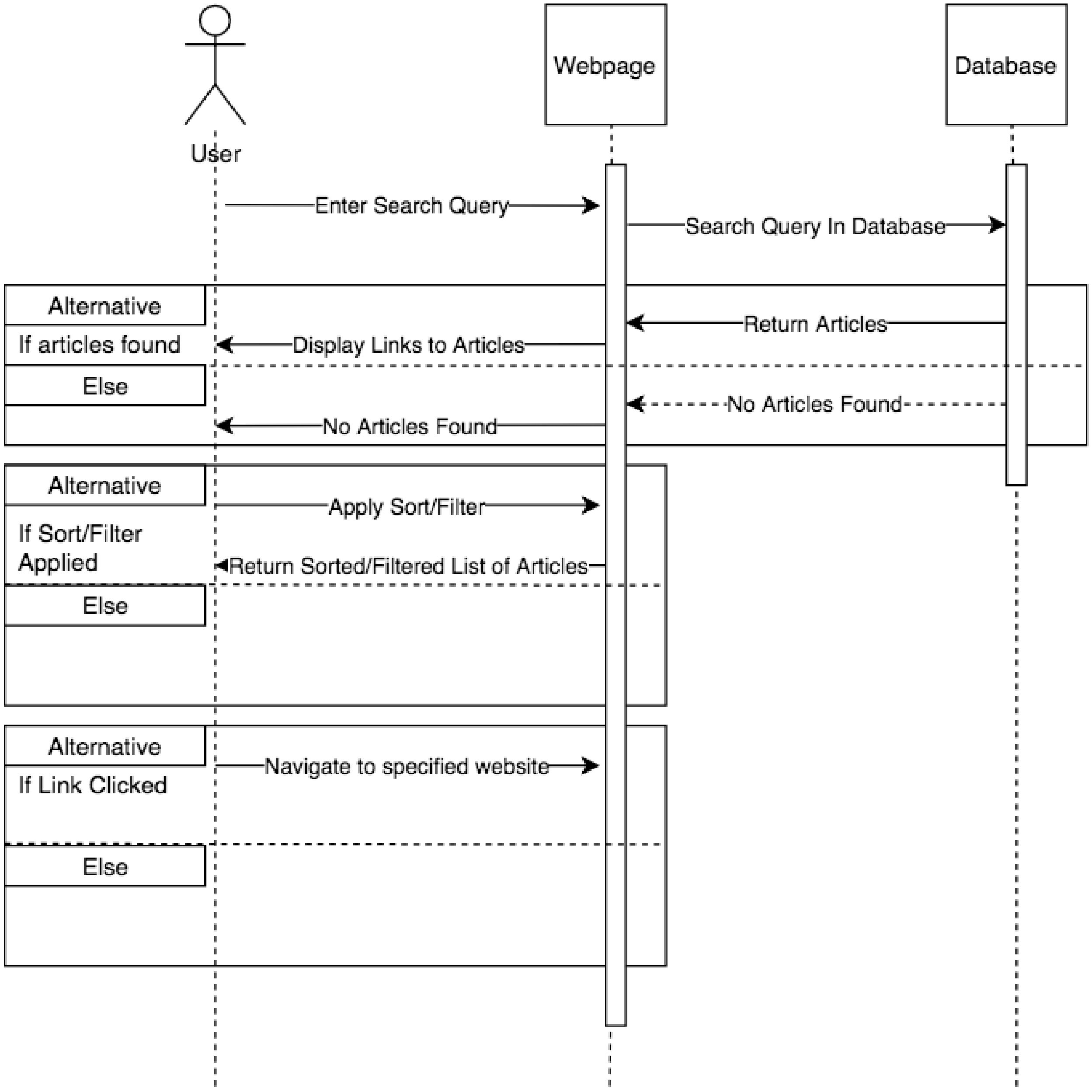
Sequence_Diagram_NVOS-22 Create Profile



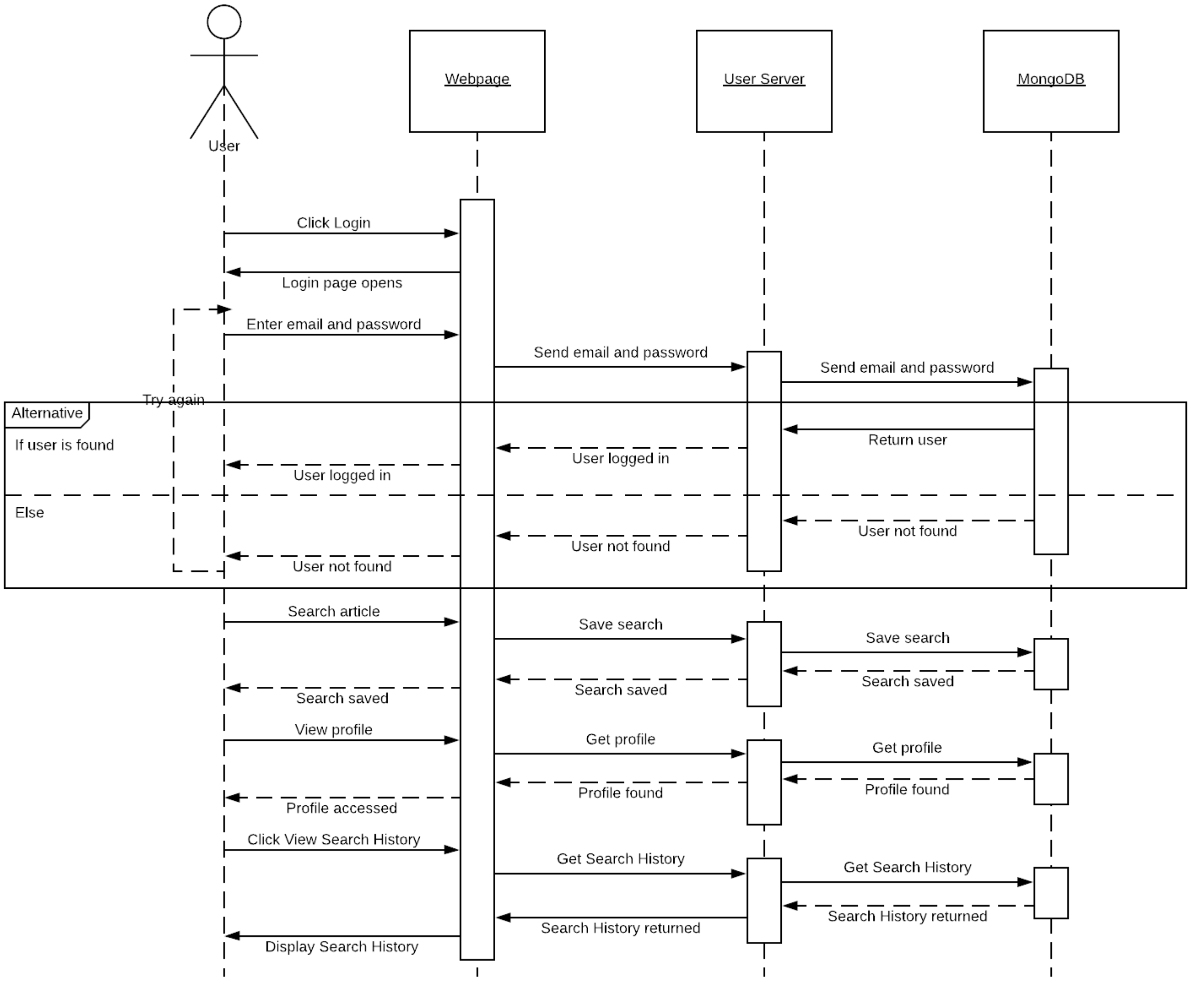
Sequence_Diagram_NVOS-24 View Saved Articles



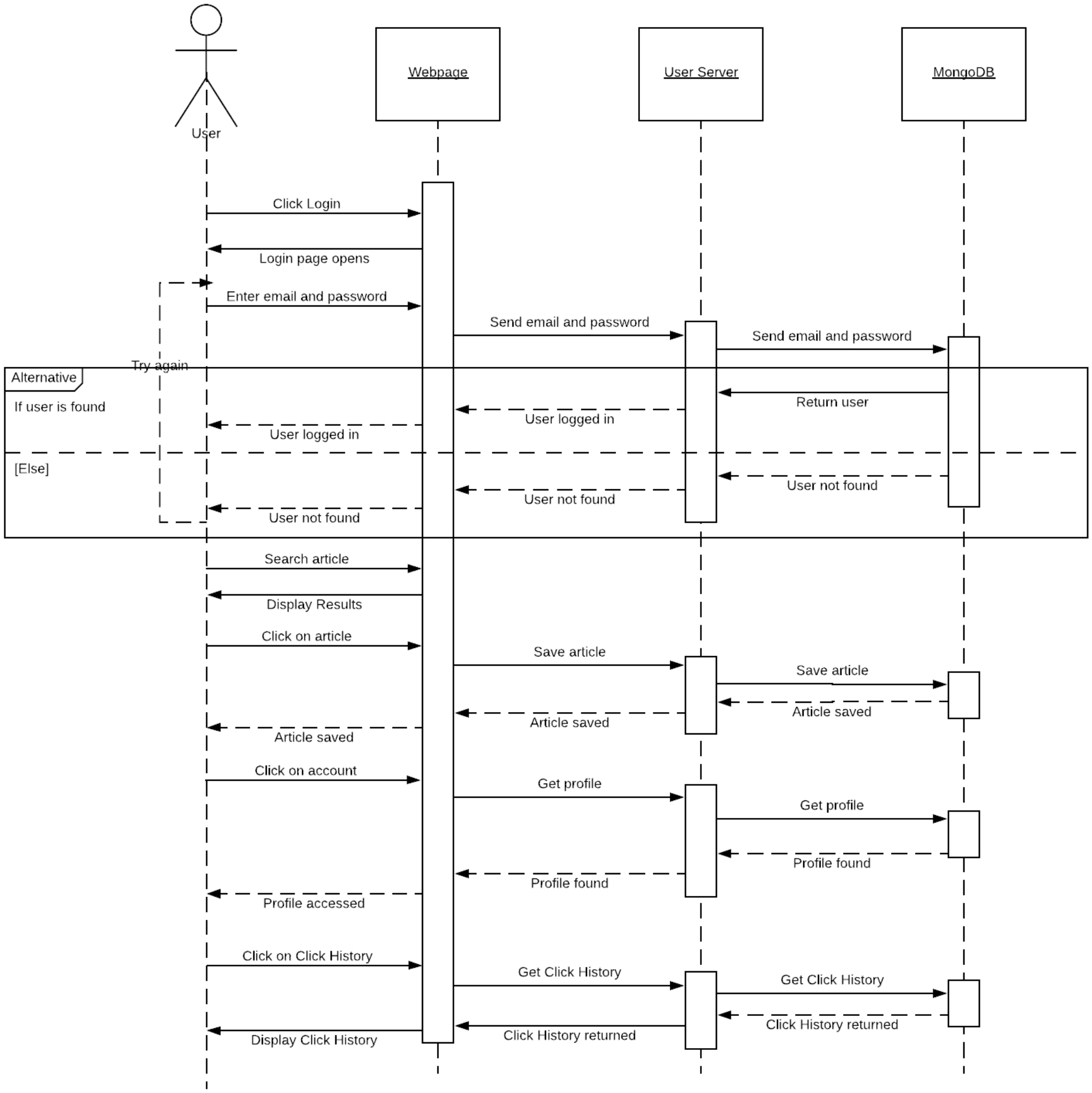
Sequence_Diagram_NVOS-28 Cross Compare With Other Search Engines



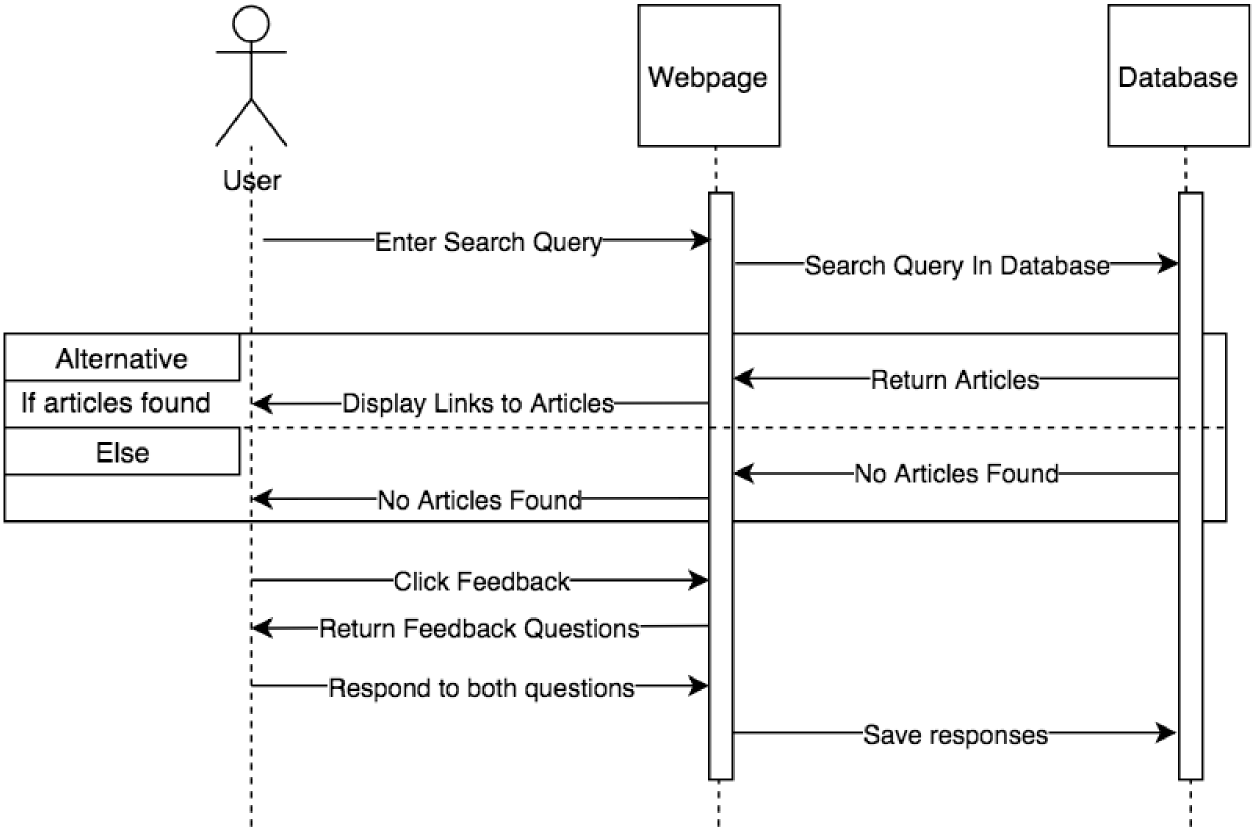
Sequence_Diagram_NVOS-29 View Search History



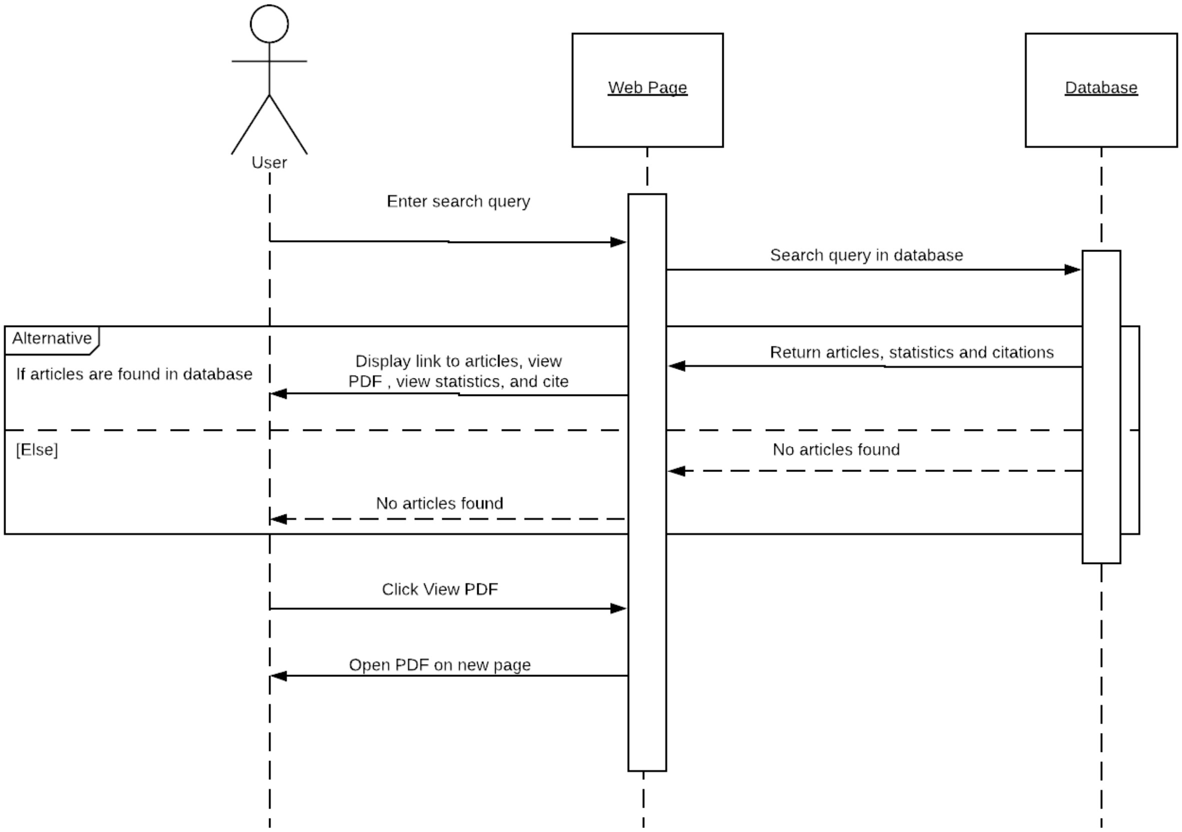
Sequence_Diagram_NVOS-32 View Click History



Sequence_Diagram_NVOS-36 General Feedback

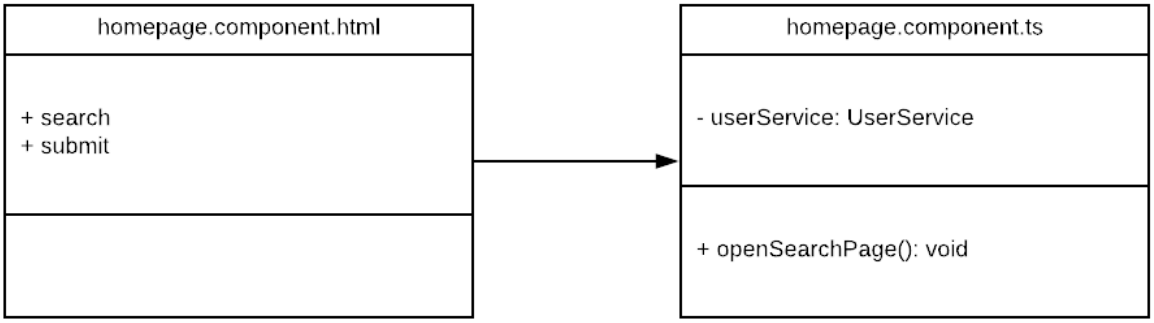


Sequence_Diagram_NVOS-40 View PDF

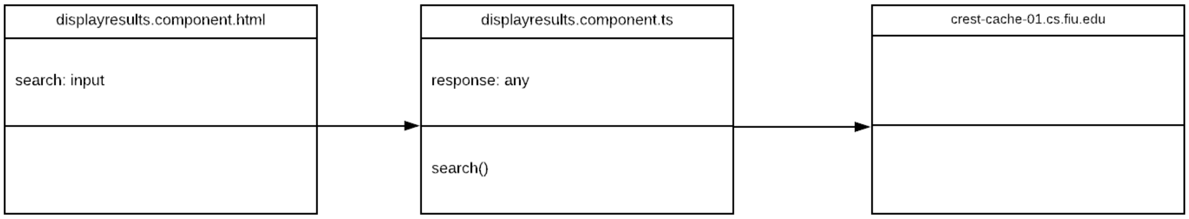


Appendix A - Class Diagrams

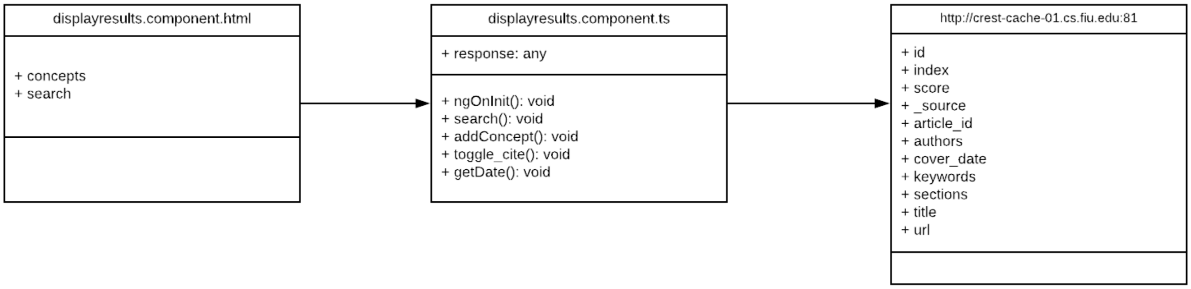
Class_Diagram_NVOS-3 Query Entry



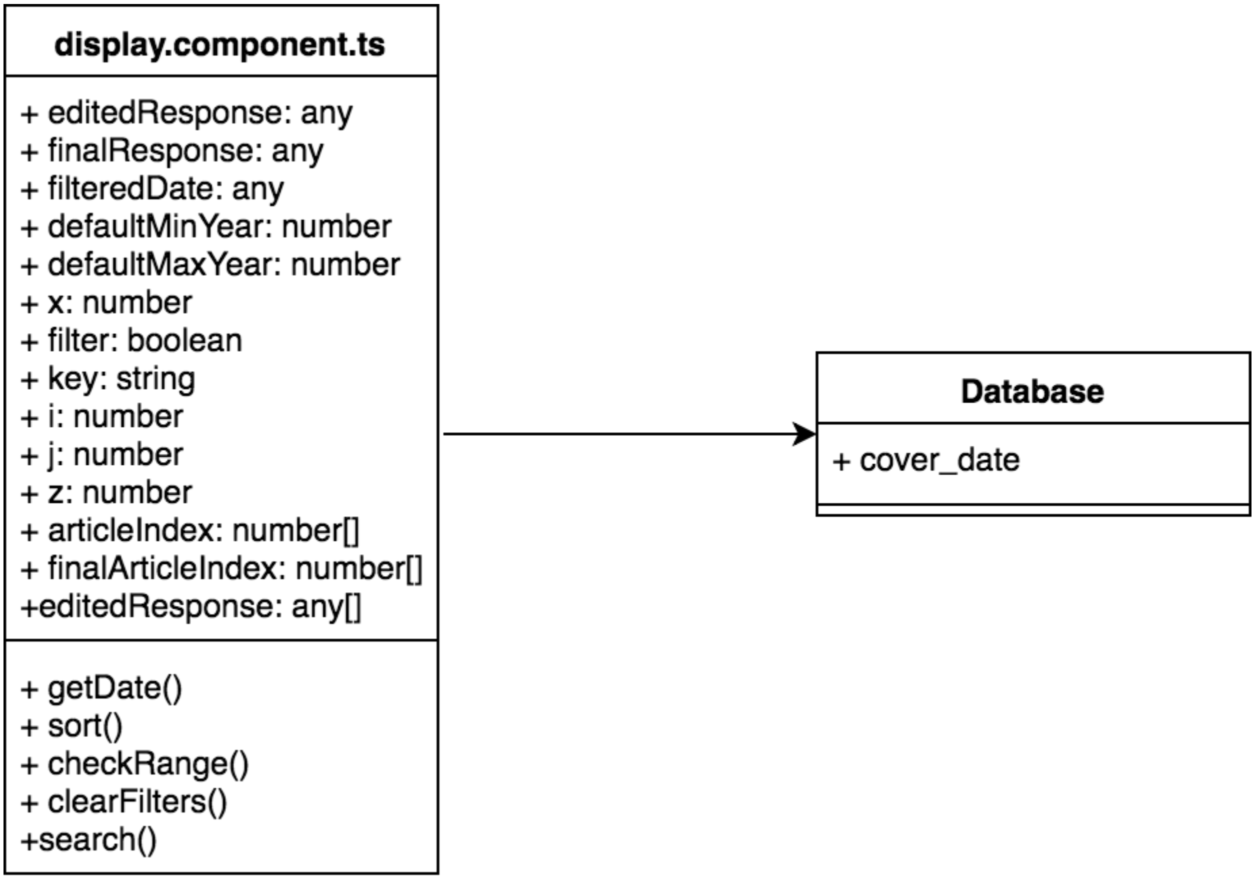
Class_Diagram_NVOS-4 Display Results


displayresults.component.html
search: input
displayresults.component.ts
response: any
search()
crest-cache-01.cs.fiu.edu

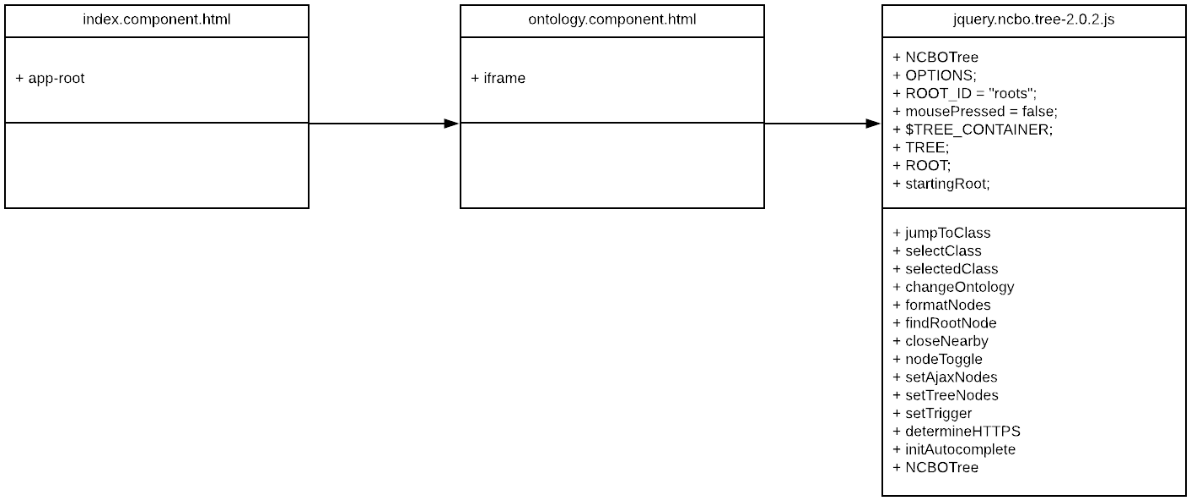
Class_Diagram_NVOS-9 Data Structure Retrieved



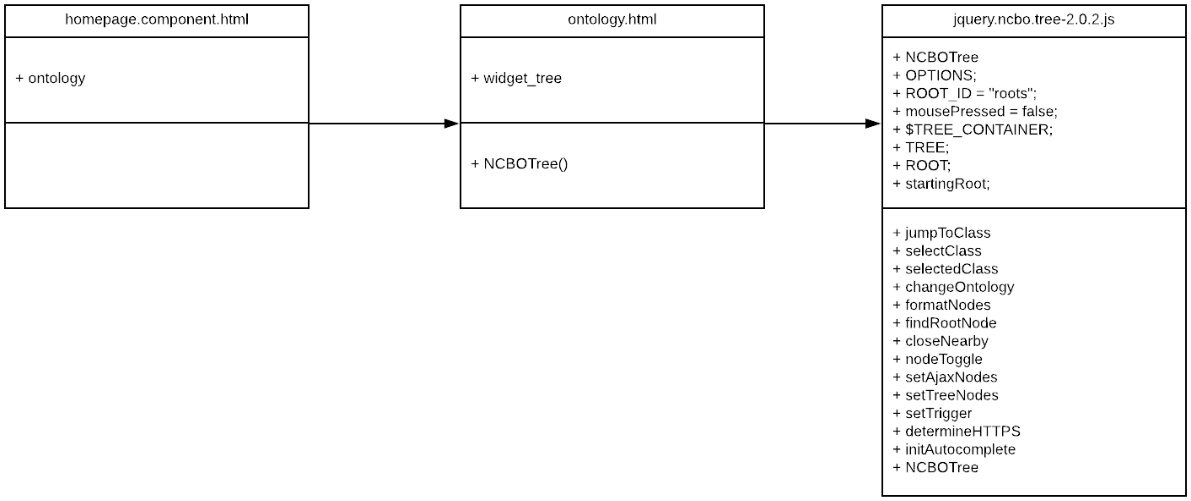
Class_Diagram_NVOS-7 Sort and Filter Results



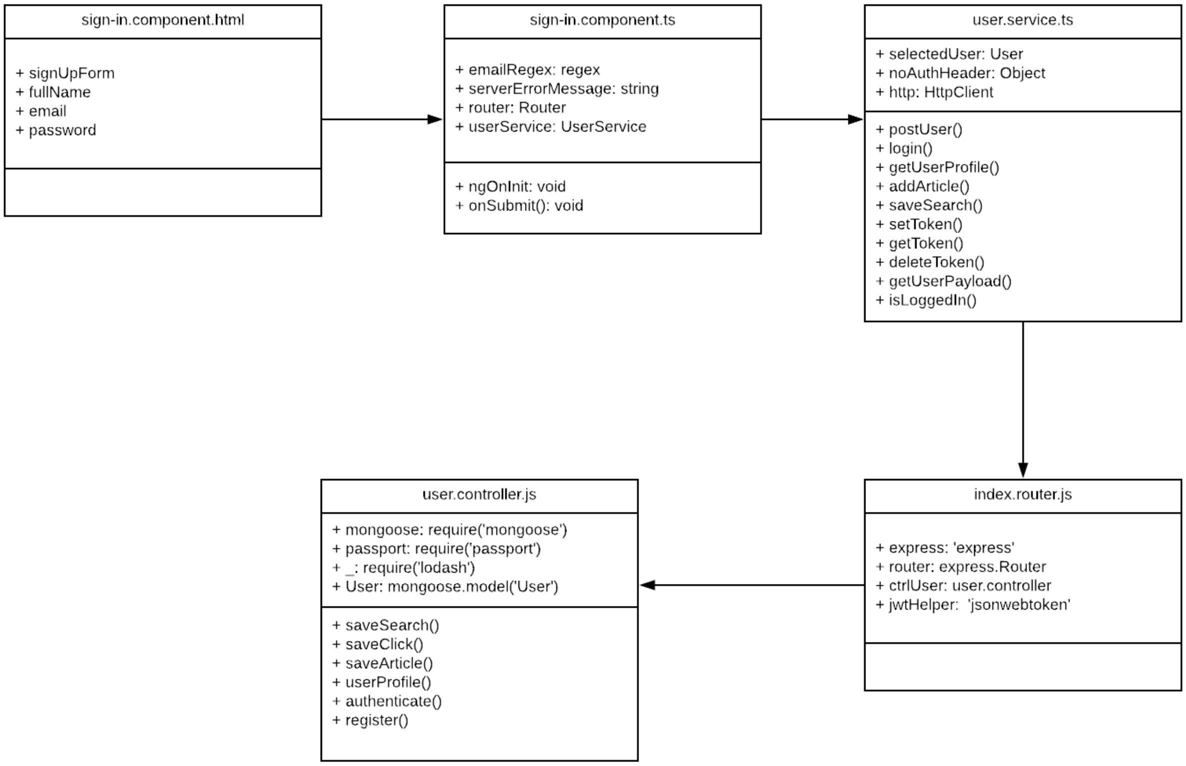
Class_Diagram_NVOS-13 View and Browse Concepts



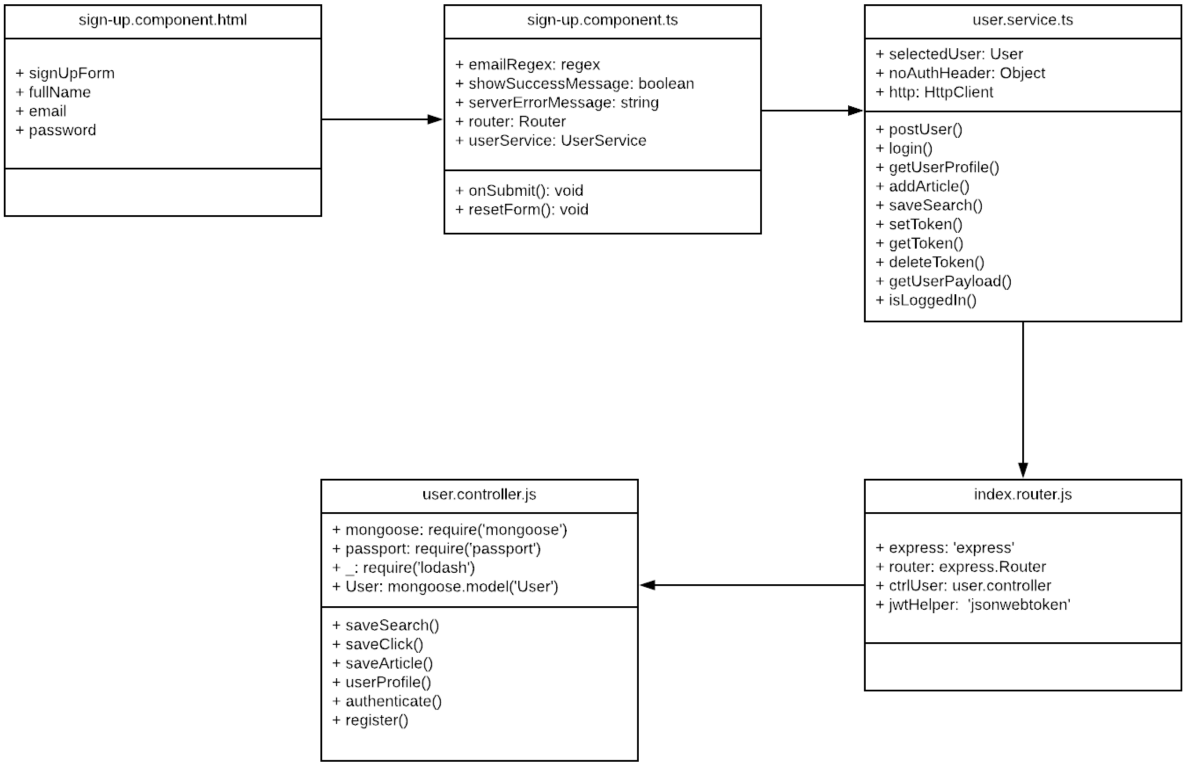
Class_Diagram_NVOS-14 Ontology Search Function



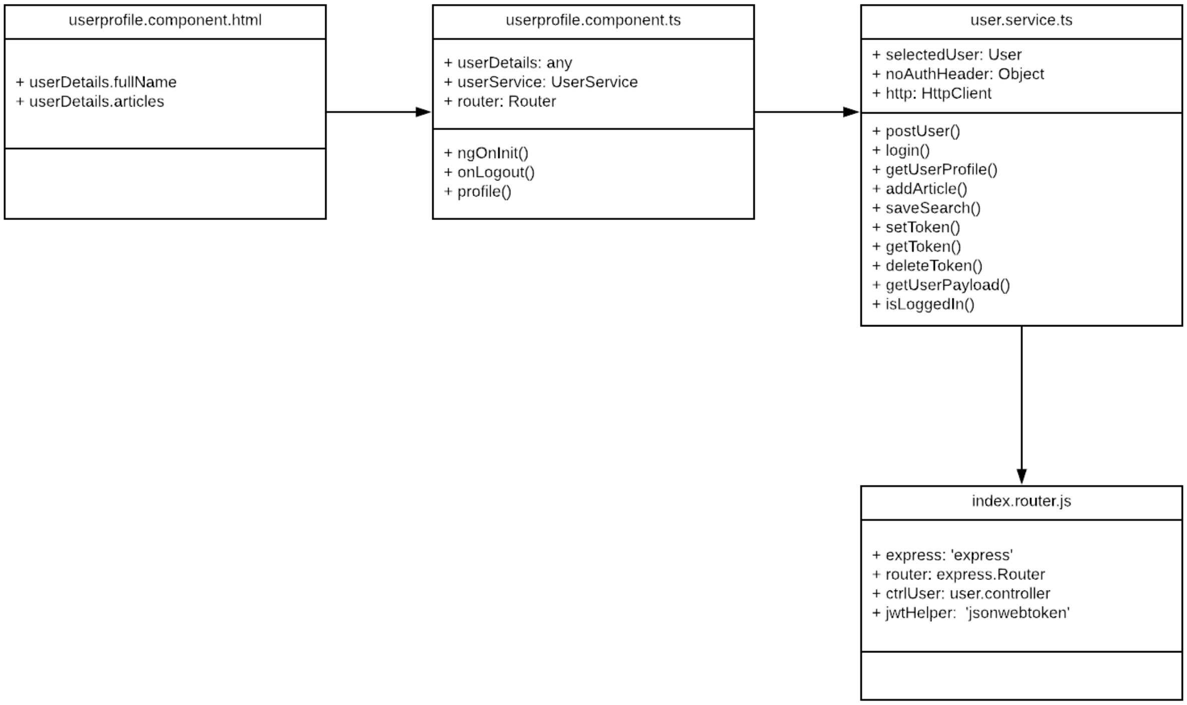
Class_Diagram_NVOS-21 Login



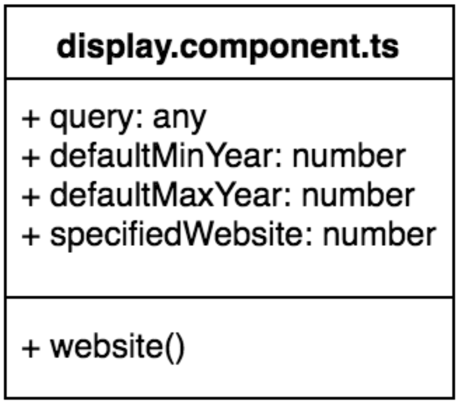
Class_Diagram_NVOS-22 Create Profile



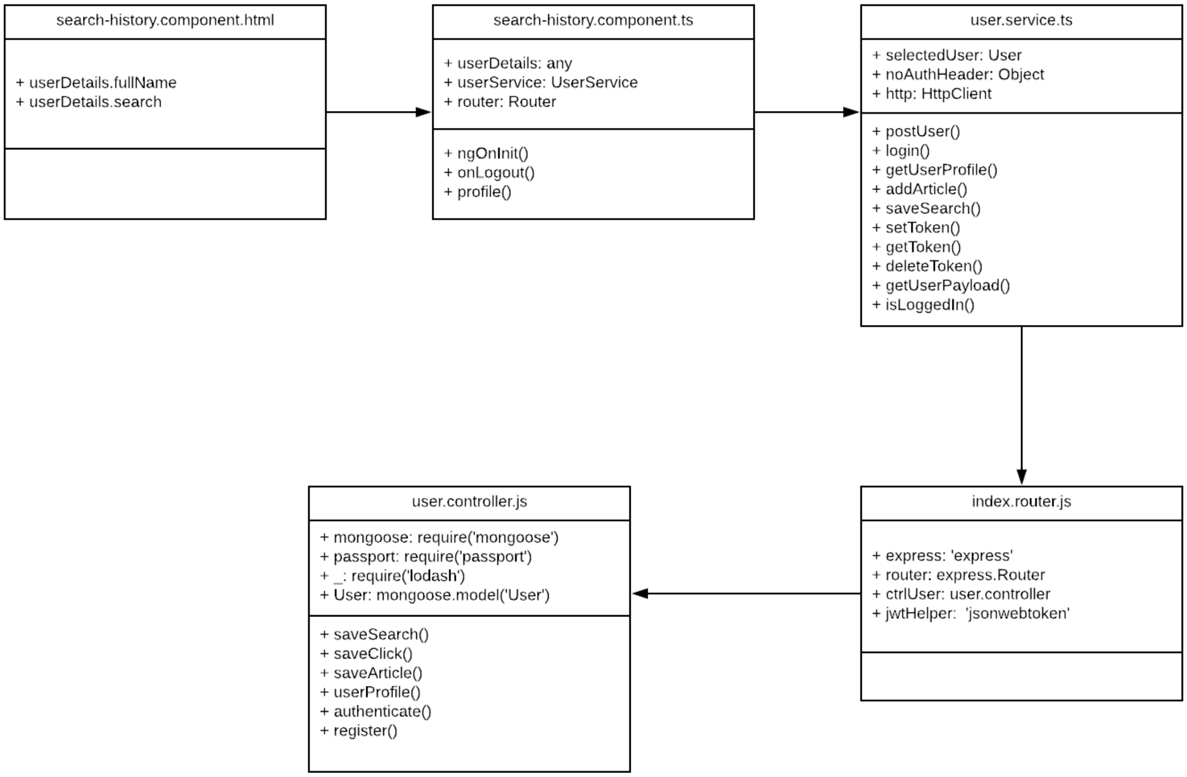
Class_Diagram_NVOS-24 View Saved Articles



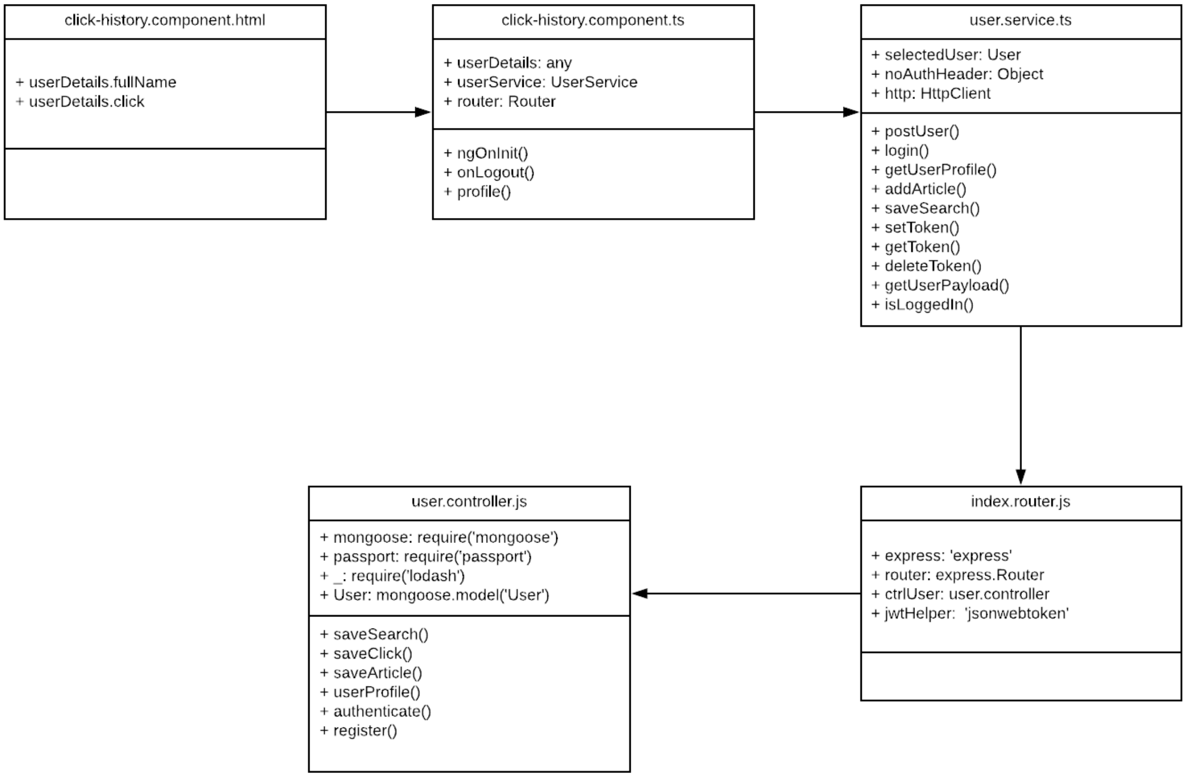
Class_Diagram_NVOS-28 Cross Compare With Other Search Engines



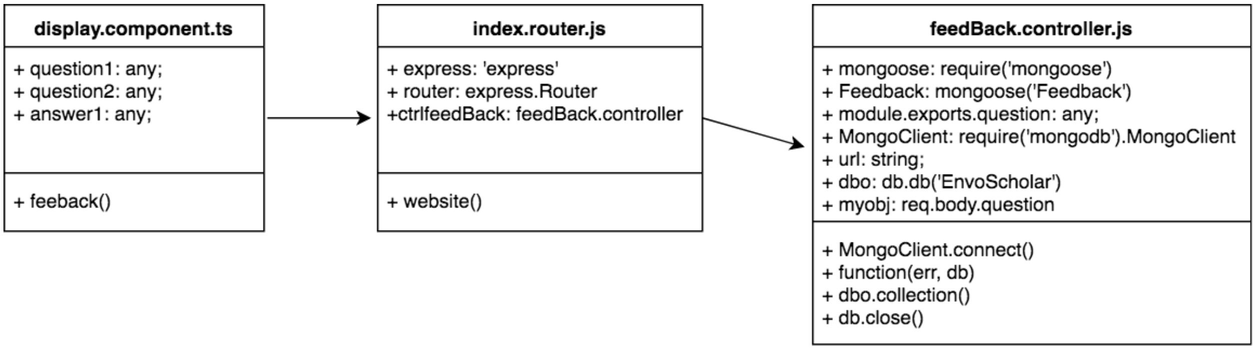
Class_Diagram_NVOS-29 View Search History



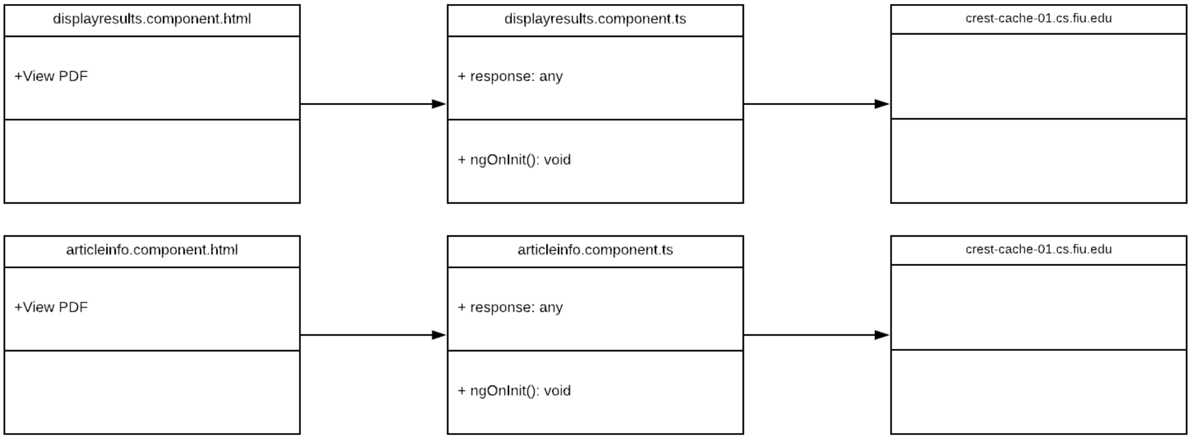
Class_Diagram_NVOS-32 View Click History



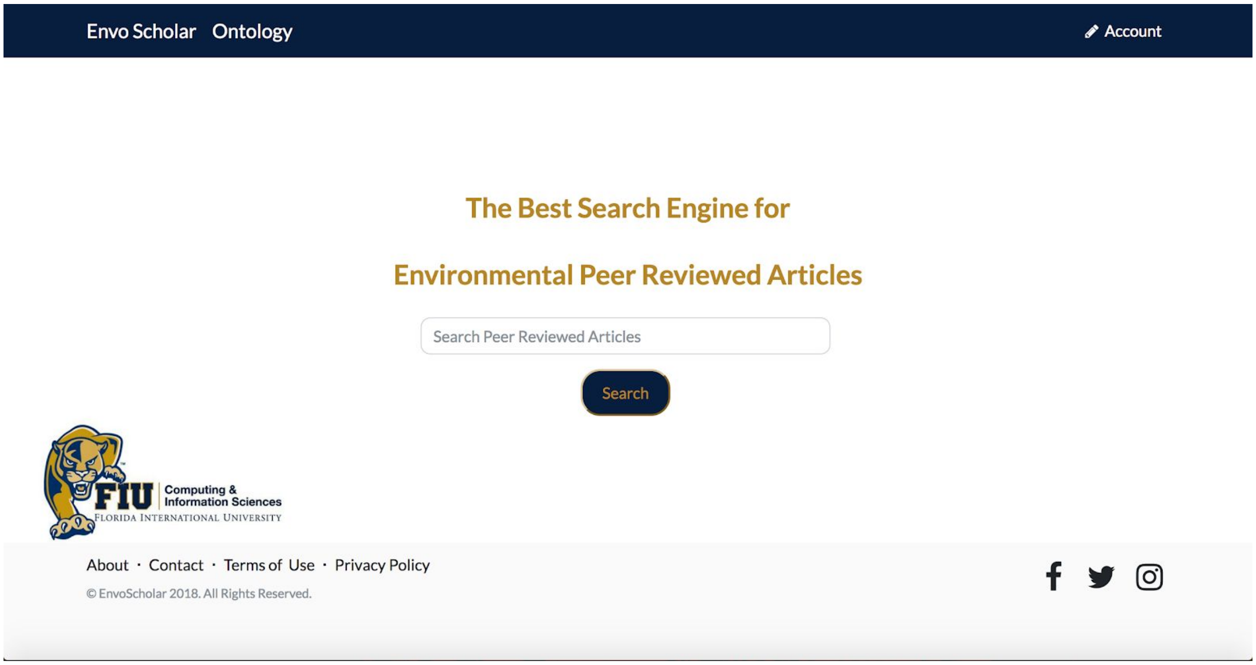
Class_Diagram_NVOS-36 General Feedback



Class_Diagram_NVOS-40 View PDF


displayresults.component.html
+View PDF
displayresults.component.ts
+ response: any
+ ngOnInit(): void
crest-cache-01.cs.fiu.edu
articleinfo.component.html
+View PDF
articleinfo.component.ts
+ response: any
+ ngOnInit(): void
crest-cache-01.cs.fiu.edu

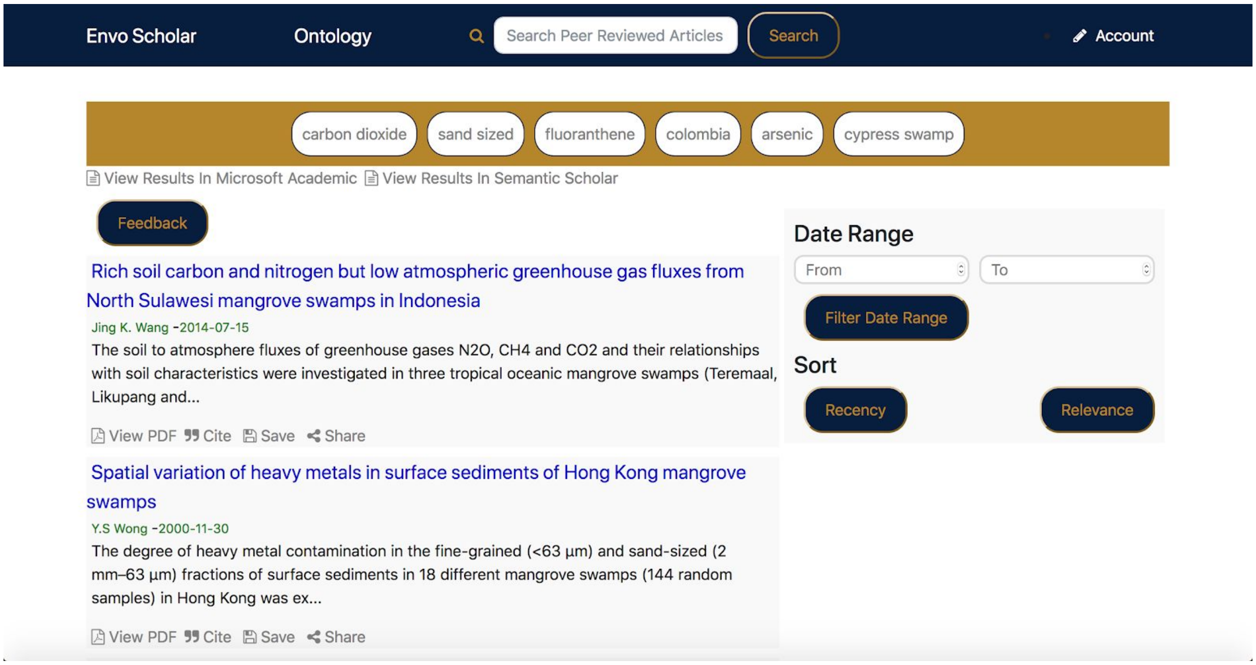
Appendix B - User Interface Design


Envo Scholar
Ontology
Account
Search Peer Reviewed Articles
Search
FIU
Computing & Information Sciences
Florida International University
About · Contact · Terms of Use · Privacy Policy

The Best Search Engine for

Environmental Peer Reviewed Articles

© EnvoScholar 2018. All Rights Reserved.



Rich soil carbon and nitrogen but low atmospheric greenhouse gas fluxes from North Sulawesi mangrove swamps in Indonesia

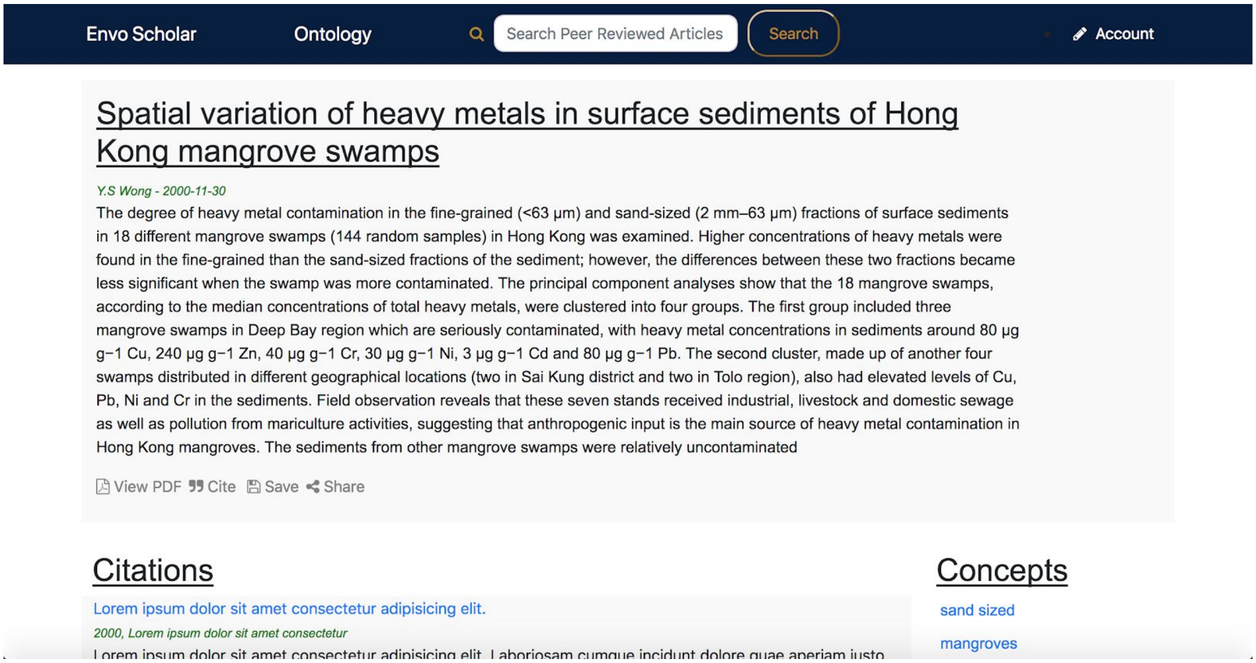
Jing K. Wang -2014-07-15

The soil to atmosphere fluxes of greenhouse gases N2O, CH4 and CO2 and their relationships with soil characteristics were investigated in three tropical oceanic mangrove swamps (Teremaal, Likupang and...

Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

Y.S Wong -2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 µm) and sand-sized (2 mm–63 µm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was ex...



Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

Y.S Wong - 2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 µm) and sand-sized (2 mm–63 µm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was examined. Higher concentrations of heavy metals were found in the fine-grained than the sand-sized fractions of the sediment; however, the differences between these two fractions became less significant when the swamp was more contaminated. The principal component analyses show that the 18 mangrove swamps, according to the median concentrations of total heavy metals, were clustered into four groups. The first group included three mangrove swamps in Deep Bay region which are seriously contaminated, with heavy metal concentrations in sediments around 80 µg g−1 Cu, 240 µg g−1 Zn, 40 µg g−1 Cr, 30 µg g−1 Ni, 3 µg g−1 Cd and 80 µg g−1 Pb. The second cluster, made up of another four swamps distributed in different geographical locations (two in Sai Kung district and two in Tolo region), also had elevated levels of Cu, Pb, Ni and Cr in the sediments. Field observation reveals that these seven stands received industrial, livestock and domestic sewage as well as pollution from mariculture activities, suggesting that anthropogenic input is the main source of heavy metal contamination in Hong Kong mangroves. The sediments from other mangrove swamps were relatively uncontaminated

Citations

Lorem ipsum dolor sit amet consectetur adipisicing elit.

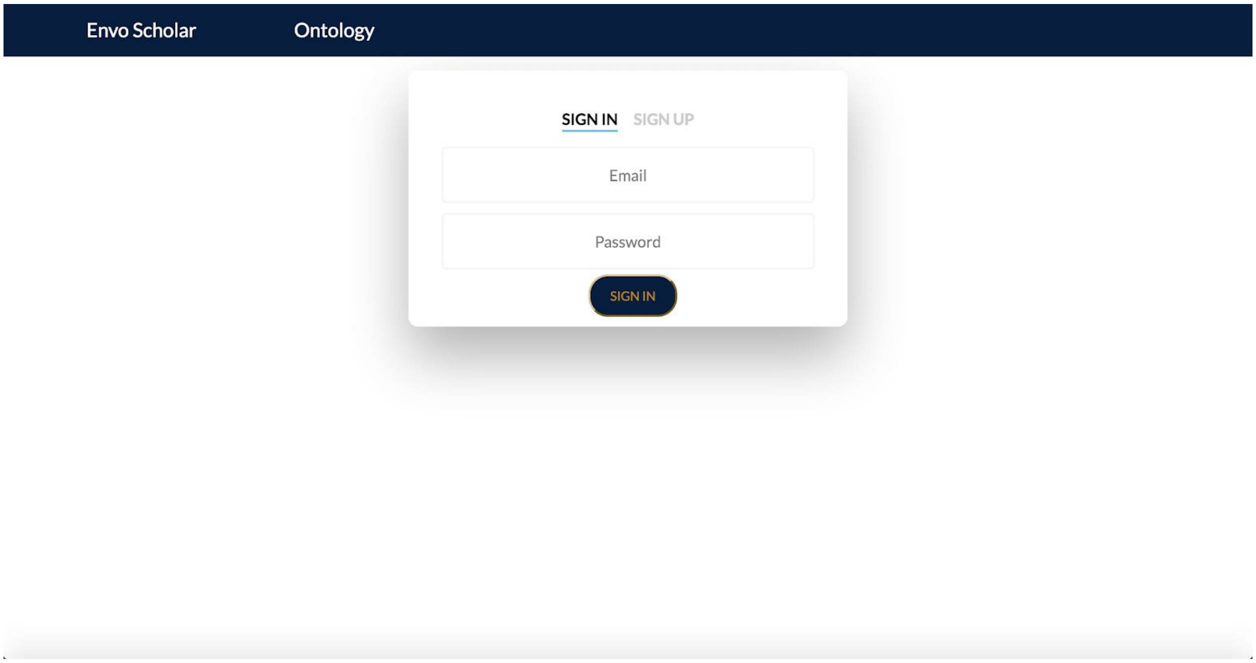
2000, Lorem ipsum dolor sit amet consectetur

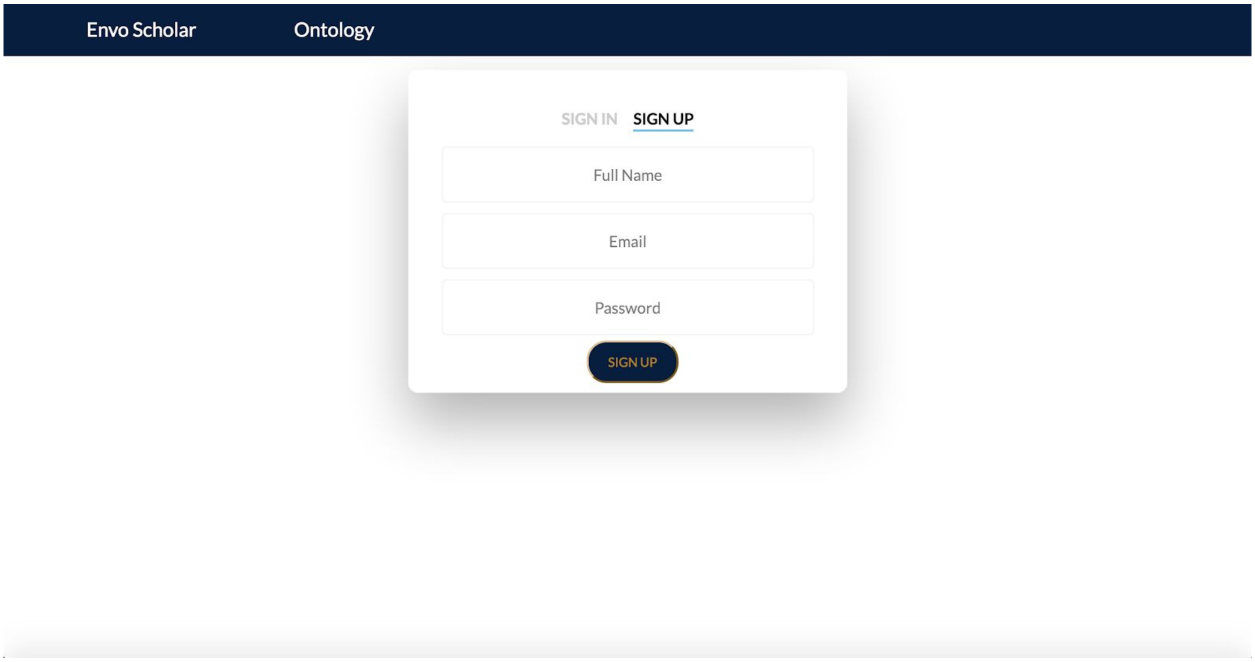
Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam cumque incidunt dolore quae aperiam iusto

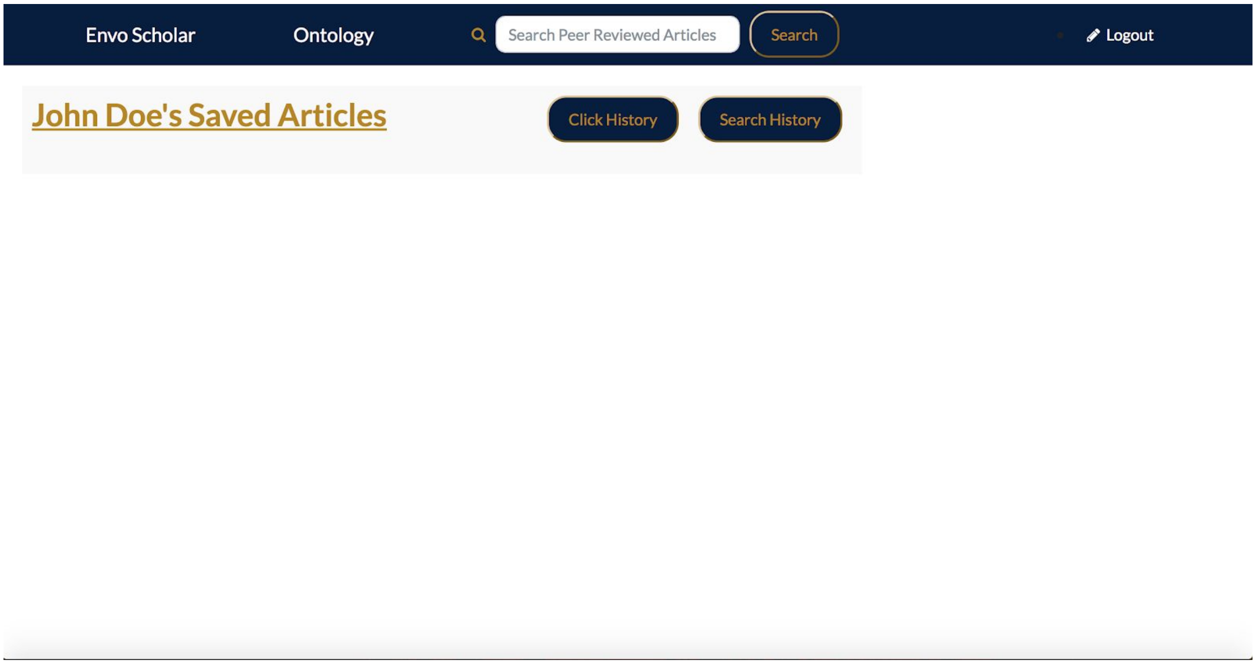
Concepts

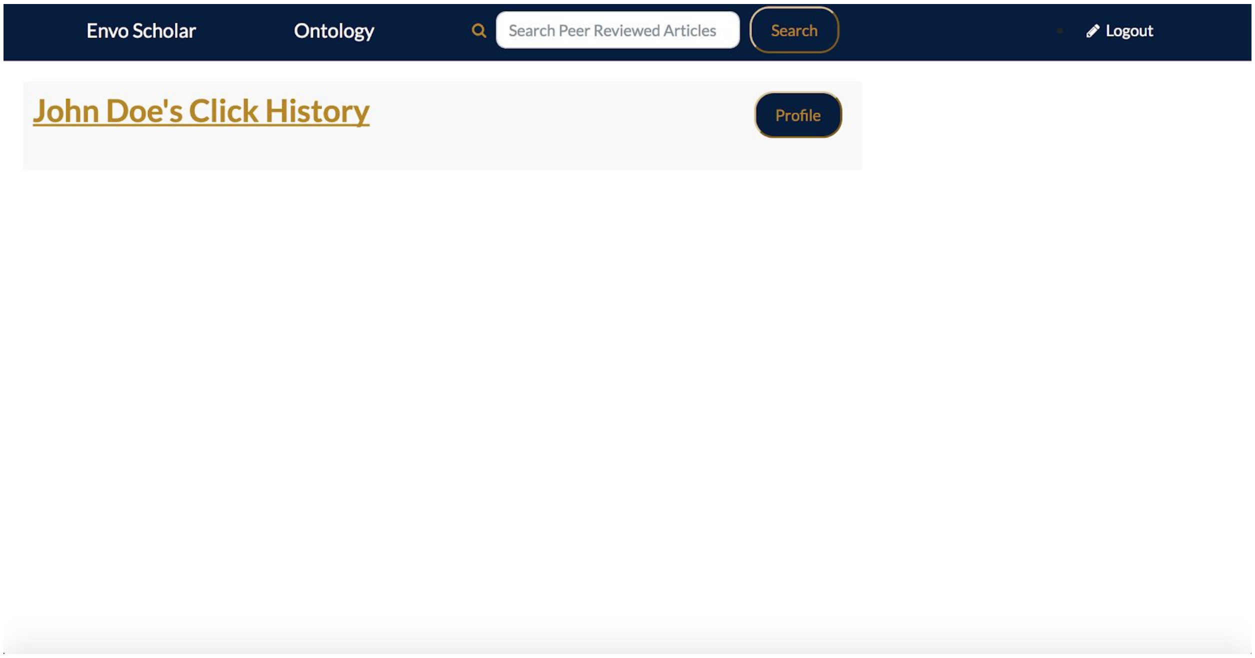
sand sized

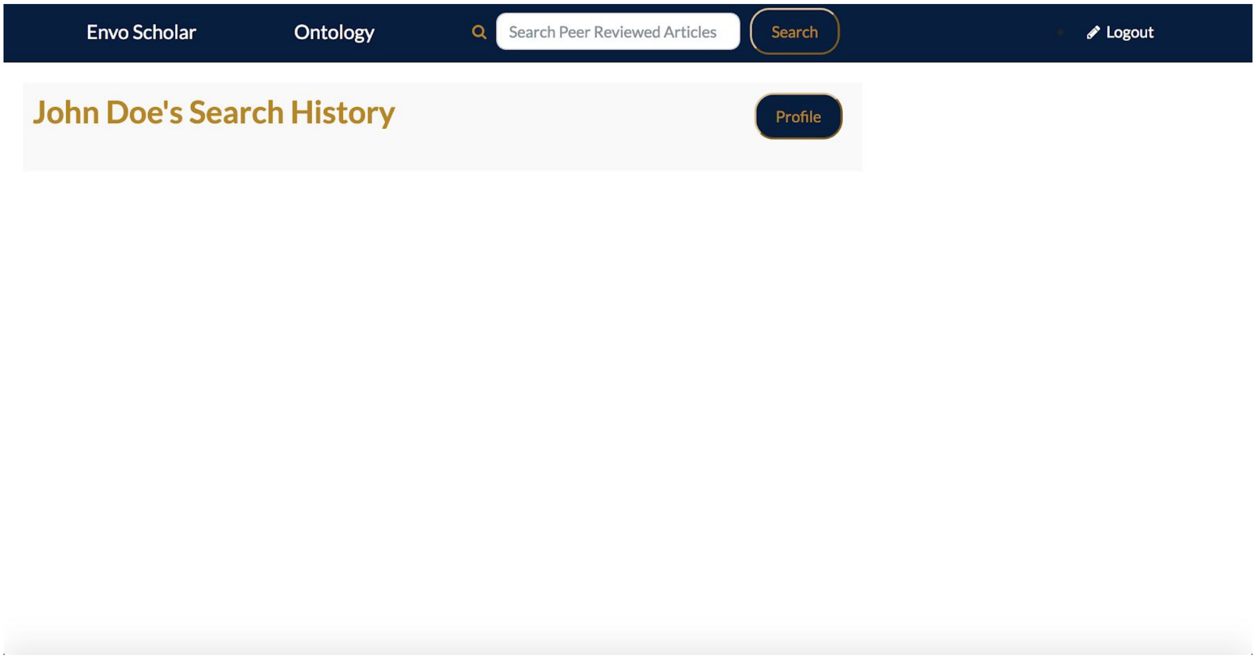
mangroves


Envo Scholar
Ontology
SIGN IN
SIGN UP
Email
Password
SIGN IN

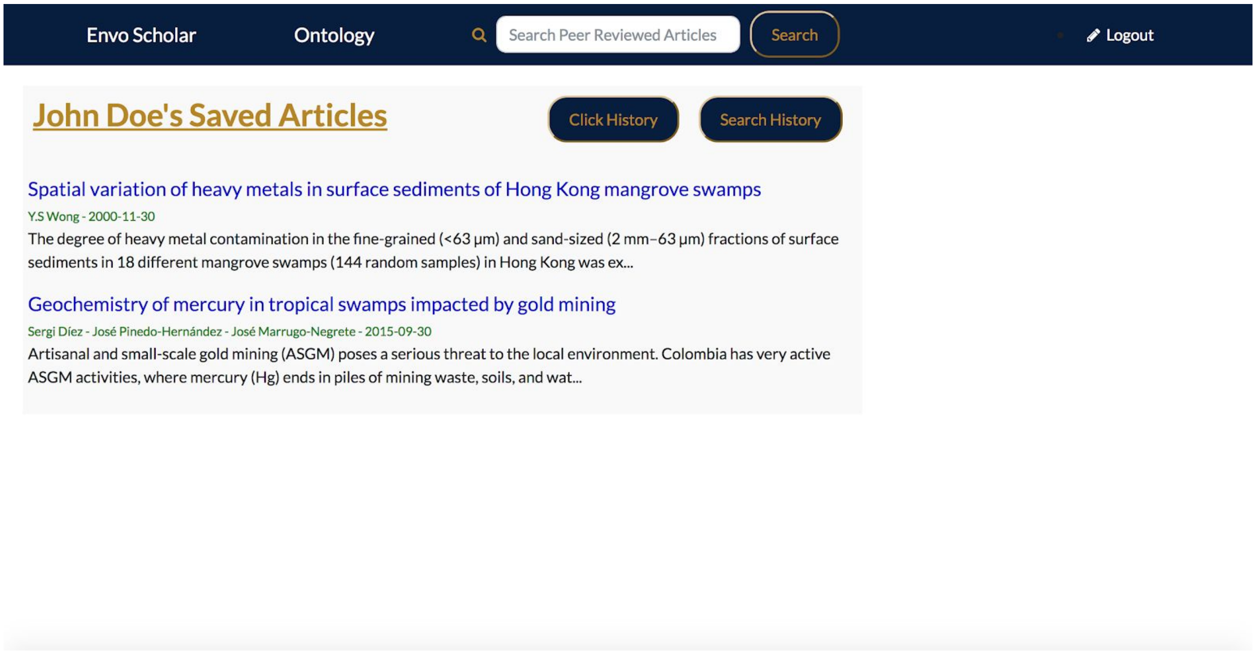

Envo Scholar
Ontology
SIGN IN
SIGN UP
Full Name
Email
Password
SIGN UP


Envo Scholar
Ontology
Search Peer Reviewed Articles
Search
Logout
John Doe's Saved Articles
Click History
Search History


Envo Scholar
Ontology
Search Peer Reviewed Articles
Search
Logout
John Doe's Click History
Profile



John Doe's Search History



John Doe's Saved Articles

Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

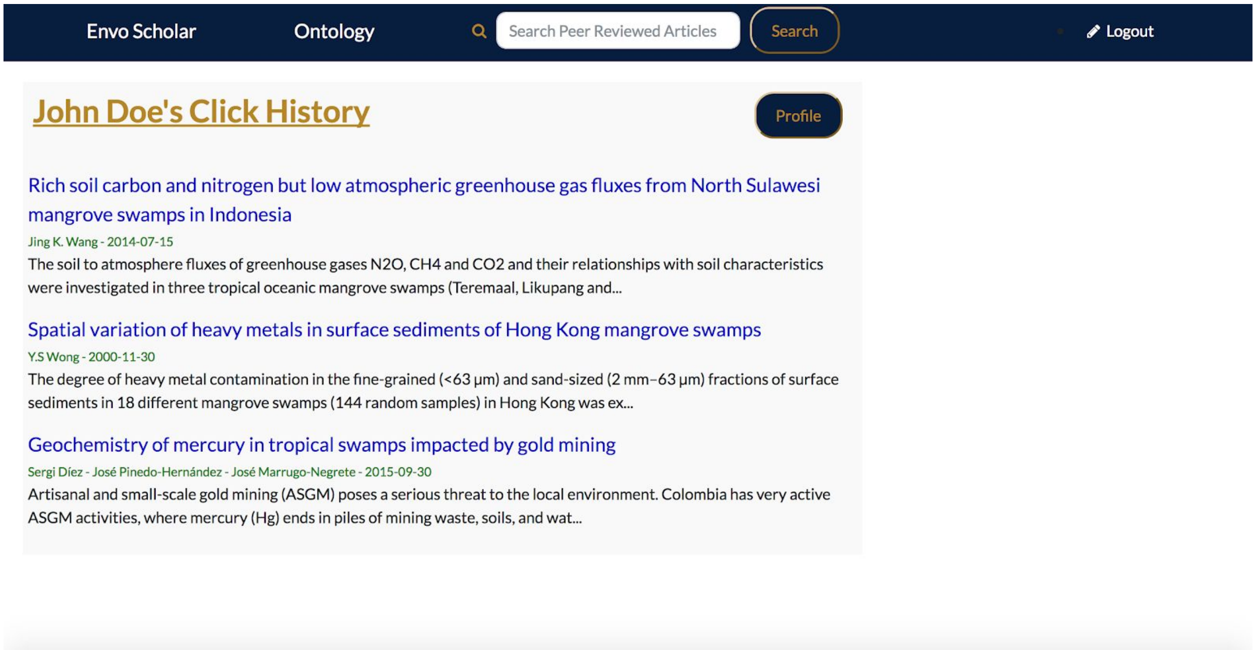
Y.S Wong - 2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 μm) and sand-sized (2 mm–63 μm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was ex...

Geochemistry of mercury in tropical swamps impacted by gold mining

Sergi Díez - José Pinedo-Hernández - José Marrugo-Negrete - 2015-09-30

Artisanal and small-scale gold mining (ASGM) poses a serious threat to the local environment. Colombia has very active ASGM activities, where mercury (Hg) ends in piles of mining waste, soils, and wat...



John Doe's Click History

Rich soil carbon and nitrogen but low atmospheric greenhouse gas fluxes from North Sulawesi mangrove swamps in Indonesia

Jing K. Wang - 2014-07-15

The soil to atmosphere fluxes of greenhouse gases N2O, CH4 and CO2 and their relationships with soil characteristics were investigated in three tropical oceanic mangrove swamps (Teremaal, Likupang and...

Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

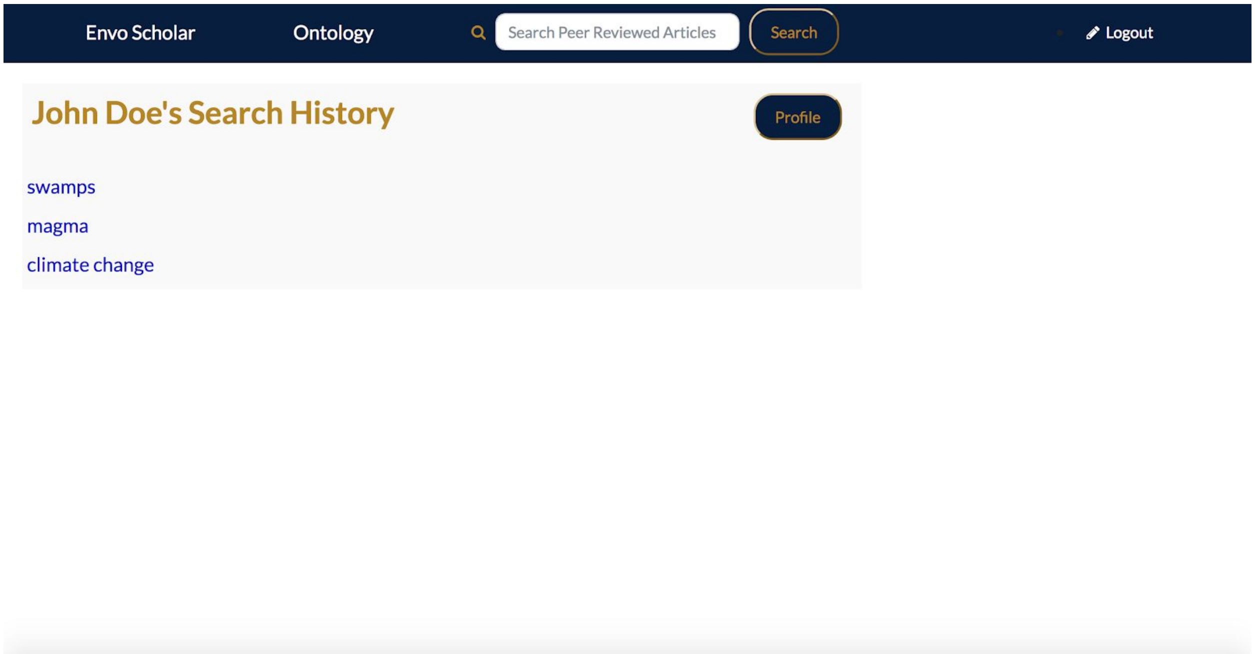
Y.S Wong - 2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 μm) and sand-sized (2 mm–63 μm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was ex...

Geochemistry of mercury in tropical swamps impacted by gold mining

Sergi Díez - José Pinedo-Hernández - José Marrugo-Negrete - 2015-09-30

Artisanal and small-scale gold mining (ASGM) poses a serious threat to the local environment. Colombia has very active ASGM activities, where mercury (Hg) ends in piles of mining waste, soils, and wat...

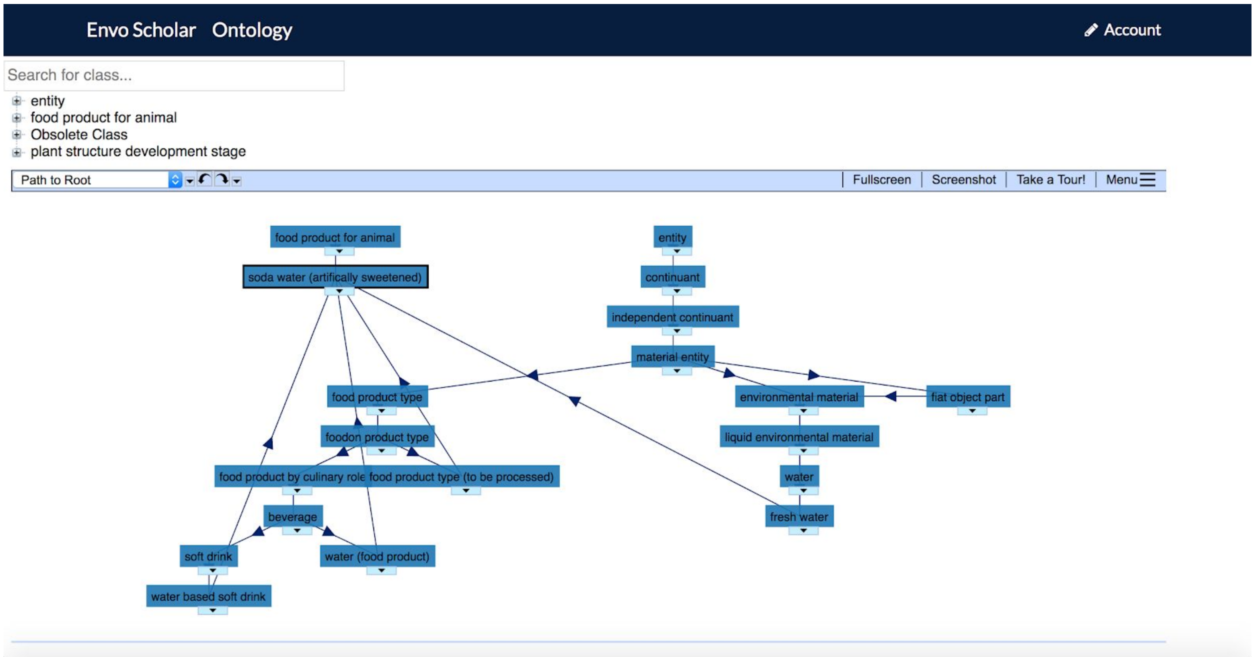


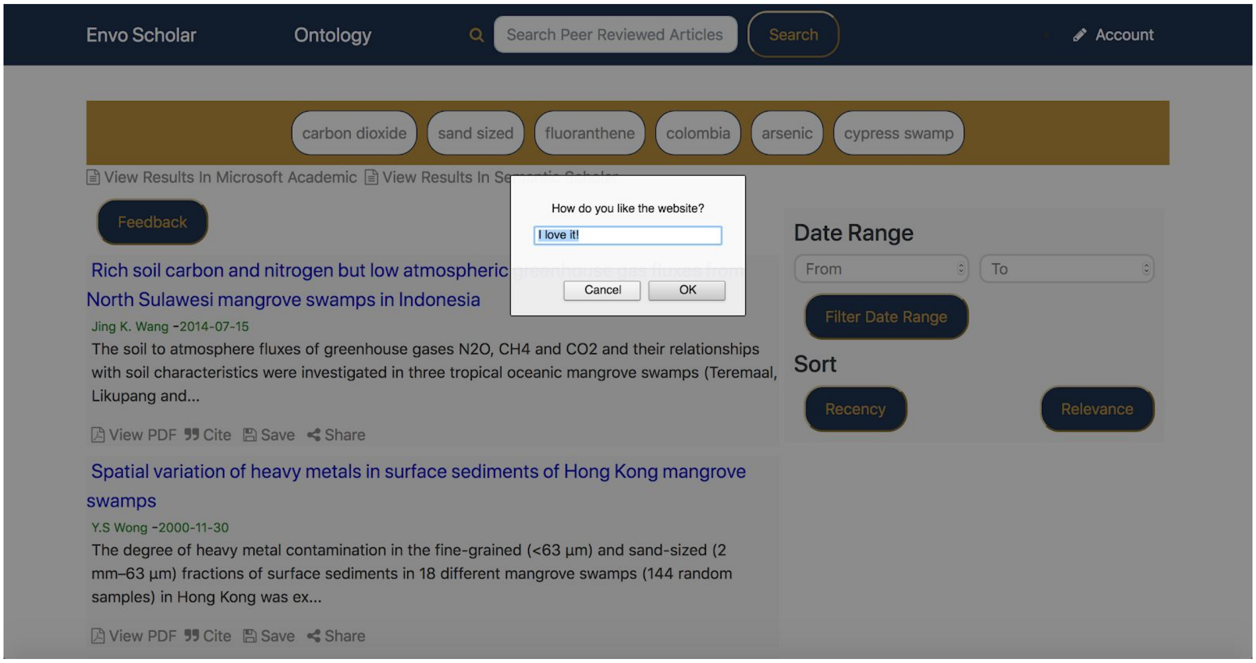
John Doe's Search History

swamps

magma

climate change





How do you like the website?

I love it!

Cancel OK

Rich soil carbon and nitrogen but low atmospheric greenhouse gas fluxes from North Sulawesi mangrove swamps in Indonesia

Jing K. Wang -2014-07-15

The soil to atmosphere fluxes of greenhouse gases N2O, CH4 and CO2 and their relationships with soil characteristics were investigated in three tropical oceanic mangrove swamps (Teremaal, Likupang and...

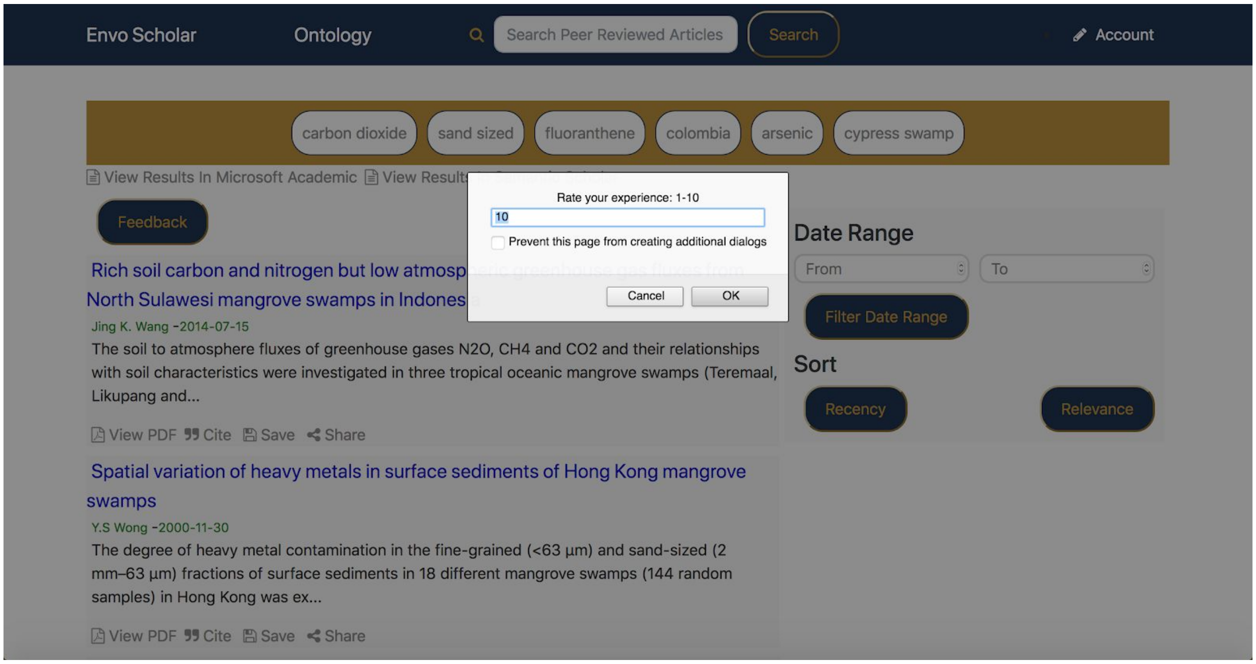
View PDF Cite Save Share

Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

Y.S Wong -2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 µm) and sand-sized (2 mm–63 µm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was ex...

View PDF Cite Save Share



Rate your experience: 1-10

10

□ Prevent this page from creating additional dialogs

Rich soil carbon and nitrogen but low atmospheric greenhouse gas fluxes from North Sulawesi mangrove swamps in Indonesia

Jing K. Wang -2014-07-15

The soil to atmosphere fluxes of greenhouse gases N2O, CH4 and CO2 and their relationships with soil characteristics were investigated in three tropical oceanic mangrove swamps (Teremaal, Likupang and...

Spatial variation of heavy metals in surface sediments of Hong Kong mangrove swamps

Y.S Wong -2000-11-30

The degree of heavy metal contamination in the fine-grained (<63 µm) and sand-sized (2 mm–63 µm) fractions of surface sediments in 18 different mangrove swamps (144 random samples) in Hong Kong was ex...

Appendix C - Sprint Review Reports

Sprint 1 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- N/A

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Filter and Search Results NVOS-7 - Sort and Filter Results

Sprint 2 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story NVOS-21 Login NVOS-21 - Login
- User Story NVOS-22 Create Profile NVOS-22 - Create Profile

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
  - We need the query itself to be sent to the database for processing.
- User Story NVOS-4 Display Results NVOS-4 - Display Results
  - The query should be displayed in the search bar and the database should send back a correct response.
- User Story NVOS-7 Filter and Sort Results NVOS-7 - Sort and Filter Results
  - In order to build the filters and sorts we need a results to test our code with.

Sprint 3 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story NVOS-13 View and Browse Concepts NVOS-13 - View and Browse Concepts
- User Story NVOS-14 Ontology Search Function NVOS-14 -Ontology Search Function

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- N/A

Sprint 4 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:45 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story NVOS-3 Query Entry NVOS-3 - Query Entry
- User Story NVOS-9 Data Structures Retrieved NVOS-9 - Data Structure Retrieved

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results
    - Filter by Date Range (input fields, not dropdown) only need to specify one but can take both dates
    - Filter by Journal (see academic.microsoft.com)
    - Put filter to the right
- User Story NVOS-4 Display Results NVOS-4 - Display Results
    - Each should have article has a "top citations"(see academic.microsoft.com)
    - Each article (Follow, share, etc.)
    - Pagination refresh page
    - Display 2 or 3 authors(depending on how it looks) add "..."

Sprint 5 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story NVOS-25 Personalized Results NVOS-25 - Personalize Results
- User Story NVOS-21 Login NVOS-21 - Login

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- User Story NVOS-7 Sort and Filter Results NVOS-7 - Sort and Filter Results
    - Fix Bugs with Filter
    - Move button to right side
    - Style the buttons
- User Story NVOS-23 View Profile NVOS-23 - View Profile
    - Save articles
    - Save search queries
- User Story NVOS-4 NVOS-4 - Display Results
    - Articles page to have a share via social media/link
    - Look into Medley for more info

Sprint 6 Review Meeting Minutes

Attendees: Bryan Bastida, Andrew Castillo

Start time: 3:30 PM

End time: 4:30 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- User Story NVOS-4 Display Results NVOS-4 - Display Results
- User Story NVOS-7 Sort and Filter NVOS-7 - Sort and Filter Results
- User Story NVOS-28 Cross Compare With Other Search Engines NVOS-28 - Cross Compare with other Search Engines
- User Story NVOS-36 General Feedback NVOS-36 - General Feedback

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- N/A

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

Installation/Maintenance:

Download/Install NodeJS: https://nodejs.org/en/

Install Angular: Go into the project folder, NVOS-Scholar, and type “npm install -g @angular/cli”
This will allow you to run the command “ng serve” which runs the Angular application.

Installing the program should be fairly simple. You should not have to install any libraries because the package.json files already have the libraries necessary to run the program. But just in case it does say that any libraries are missing, you can go to either the nvos-scholar folder or the user_database_server folder and type ‘npm install’. It will reinstall the libraries necessary. If you’re on a Mac and you get any permission conflicts, do ‘sudo npm install’ and type your password when prompted.

There are 3 other things you need to download. MongoDB, MongoDB Compass, and Cisco AnyConnect Secure Mobility Client.
Check references for links.

Developer User Manual:
Follow all of these steps before using Envo Scholar. You may be tempted to try it out once you got the front end working but functionality will be limited.

Navigate to the NVOS-Scholar folder.
Inside you will find two folders, nvos-scholar and user_database_server
These folders contain all the source code you will need.
Open 3 terminals.
With the first terminal, navigate to the nvos-scholar
Run ng serve --open
This will open the Angular based front end on your browser on localhost:4200

Once you have this running you can connect to the VPN.
The reason you need this VPN is to get access to the articles database which is being hosted from a server on FIU. If you are at FIU when running the program, you do not need to sign into the VPN because you're already on the same network.

The 2nd terminal will be used to run the MongoDB server
Check references for running MongoDB on Mac.
If you're on Windows or another OS, I would suggest finding another video that does the same thing but on your OS. No video will be suggested as we do not know the outcomes.

Once you've followed the video and are you are told to enter "mongod --dbpath $MONGO_HOME/data" (could possible be $MONGO_HOME/bin or whatever you named the folder), you will see in the terminal a message that says "waiting for connections on port 27017". At this point you want to open up MongoDB Compass. Leave the default settings and click Connect. You should now see on the terminal "connection accepted from...".

With the 3rd terminal, navigate to user_database_server
Run nodemon server.js
You should see "Server started on port: 3000" and "Mongodb connection succeeded
"

Now you are free to use Envo Scholar.

"Actual User" User Guide:

In the homepage of EnvoScholar, you are presented with a navigation bar on top that has the links Envo Scholar (will navigate you back to the homepage), Ontology (will navigate you to the Ontology), and Account (will navigate you to either the login page or your userprofile).

You are also presented with a search bar. In the search bar, you can type a search, let us say for example "contamination in the everglades". You can hit the Submit button or press Enter on your keyboard.

You will be navigated to the displayresults page where you will be presented with a gold bar that has 6 buttons. Any one of those buttons that you press will change your search to what is specified on that button. You will see a link to compare your search query with Microsoft Academic and Semantic Scholar. You will see a Feedback button that allows you to give general feedback on the user experience of the website. You will see a Date Range section that allows you to type the years of To and From as well as a Sort section for sorting by Recency and sorting by Relevance.

Each article has its own information showing separately in a gray box. You see the article title, at most 3 authors, some of the abstract, the date it was published, and links to View PDF, Cite, Save, and Share.

When you click on one of the article titles it will navigate you to the articleinfo page. In here you see the title again, the complete abstract, all of the authors, the date it was published, the links to View PDF, Cite, Save, and Share, a section called Citations, which at the moment has dummy text because the citations are not yet implemented in the database, and a section for Concepts, which at the moment is using Keywords as a placeholder.

From any of the pages you can navigate to the Ontology or your Account. So let us start off with Account. When you click on the Account link, if you are not signed in, it will navigate you to the login page, otherwise it will take you to the userprofile. In the sign in page you can type your sign in credentials if you already have an account. Otherwise, click on the Sign Up button and fill out the boxes you create an account. Once completed, you will be prompted that the information was saved successfully. Then click on the Sign In button and type in your information and click Sign In. You will be redirected to the userprofile page.

In the userprofile page, you will see your Saved Articles, can click on the Click History Button which will redirect you to the clickhistory page. There you will see your Click History and a button for Profile which takes you back to userprofile. In the userprofile page you will also see a button called Search History. This button will navigate you to the searchhistory page which will display the Search History and a button to go back to the userprofile.

At this moment users can only logout from the userprofile, clickhistory, or searchhistory pages.

Now from any page click on the Ontology. In the Ontology page, you are displayed 2 widgets. One widget has a search box and a vertical tree. Typing something into the search box will expand the tree to show you the path to what you typed (if it is part of the Envo ontology that is). And in the other widget you can view the ontology as a tree which is interactive.

Shortcomings/Wishlist:

- Personalized Results: Smart feedback based on user preference of relevance
- View Search History: Admin privileges
- Visualization of text
- Visualization of Concepts: User customizing their own query for accurate search
- Add Markups: User can mark up their own query for a more detailed search
- Popular Concepts: users see what is more popular for query construction
- User Specific Concepts: users see what their own preference is for query construction
- Drag and Drop Concepts: for query construction
- Article Features: connecting the View Citations given backend support, view article stats

References

Documentation for Angular 6: https://angular.io/docs
Documentation for NodeJS: https://nodejs.org/en/docs/
Documentation for MongoDB:
https://docs.mongodb.com/manual/?_ga=2.127843661.1176652703.1543341633-893410526.1541264451

Click the link below to download and make sure to specify your OS. (**MongoDB**)
https://www.mongodb.com/download-center/community

Click the link below to download and make sure to specify your OS. (**Compass**)
https://www.mongodb.com/download-center/compass

Follow the steps to download the client and to sign up for the Two Factor using DUO.
https://network.fiu.edu/vpn/
This should help with getting Two Factor with DUO.
https://castic.fiu.edu/main/app/core/helpguides/HowtoVPN.pdf

Youtube Videos that were used to set up the MongoDB and user registration/login:

MEAN Stack User Registration Using Node JS - Part 1
https://www.youtube.com/watch?v=m34FCkBd7UU&t=1110s

MEAN Stack User Registration Form With Angular 6 - Part 2

https://www.youtube.com/watch?v=V9zDNfVs7Z4

MEAN Stack Jwt Authentication in Node JS API - Part 1

https://www.youtube.com/watch?v=T8qepiTbJi4

MEAN Stack Login and Logout in Angular 6 - Part 2

https://www.youtube.com/watch?v=r0QYP61bCCM&t=25s