Name Surname: Esra Nur Özüm

Course: SWE 573 Software Development Practice

Date: 18.05.2025

Project Name: https://swe573-spring25.onrender.com/

Project Github Repository URL: https://github.com/esranrzm/SWE573_SPRING25

Project tag version URL: https://github.com/esranrzm/SWE573_SPRING25/releases/tag/v0.9

Username and passwords for the application login:

- Username: admin, password: admin123 (admin of the website and has special write and update permissions)
- Username: veli, password: veli123
- Username: esranrzm, password: esra123
- Username: johnd, password: john123

HONOR CODE

Related to the submission of all the project deliverables for the Swe573 Spring 2025 semester project reported in this report, I declare that:

- I am a student in the Software Engineering MS program at Bogazici University and am registered for Swe573 course during the Spring 2025 semester.

- All the material that I am submitting related to my project (including but not limited to the project repository, the final project report, and supplementary documents) have been exclusively prepared by myself.

- I have prepared this material individually without the assistance of anyone else with the exception of permitted peer assistance which I have explicitly disclosed in this report.

Esra Nur Özüm

Table of Contents

Project Details

1. Overview

This project aims to create a web-based research and discovery platform designed to help curious individuals gather, organize, and connect information collaboratively. The core objective is to build a tool that supports collective intelligence by enabling users to start and contribute to research topics, structure diverse data, share further opinions, and form meaningful connections between ideas.

The platform will serve primarily as a space for exploration and knowledge building. Users will be able to create profiles, start a new research topic, and collaborate with others through a flexible system that enables adding tags (e.g., #Istanbul, #BogaziciUniversity, etc.), adding comment on research, contributing to the research connection graph, and searching among the researches and users. A graph-based data structure will be used to visualize and manage the connections between different pieces of information, promoting deep investigation and discovery.

There are two main user roles: regular users who can freely create and join research activities, and admin who can manage users and research. The platform includes fundamental features like user registration, login, and research creation, but will remain simple and focused, without complex permissions or notifications.

2. Software Requirements Specifications

1. Research topics, Connections, and User operations

1.1 User Operations related Requirements

1.1.1 The system shall enable user to create a new account

1.1.2 The system shall enable users to login with their correct credentials

1.1.3 The system shall enable users to update their biography, username, password, name, and surname

1.1.4 The system shall allow users to delete their account

1.1.5 The system shall show meaningful error messages to the users when a problem occurs in the system, or an incorrect data entrance happens

1.1.6 The system shall require users to provide profile information including:

- Skills
- Name
- Surname
- Email
- Password
- Age
- Gender

1.2 Research Topic related Requirements

1.2.1 The system shall allow users to create a new topic.

1.2.2 The system shall allow users to update their research topic details such as description, tags, and title

1.2.3 The system shall allow users to delete their research topic

1.2.4 The system shall allow users to comment on research

1.2.5 The system shall allow users to update their comment

1.2.6 The system shall allow users to delete their comment

1.2.7 The system shall allow users to see the research details for the selected research

1.2.8 The system shall enable users to search among the research

1.2.9 The system shall display only 10 matching results per page

1.2.10 The system shall enable tag-based filtering for research projects

1.3 Graph Operations Requirements

1.3.1 The system shall allow users to add a node to the graph

1.3.2 The system shall allow users to delete a node from the graph

1.3.3 The system must prevent deletion of a node that is not created by that user

1.3.4 The system shall enable users to add a connection between the node that they created and any other node in the graph

1.3.4.1 The system must prevent a connection that has the target and source node as same (loop edge)

1.3.4.2 The system must show an error message if the user selects same node for target and source node

1.3.4.3 The user must enter a label to be able to create a connection between nodes

1.3.4.4 The system shall enable users to only see their own nodes in target node list

1.3.5 The system shall allow users to delete a connection that they created before

1.3.6 The system shall enable users to zoom in and zoom out the graph

1.3.7 The system shall allow users to see the graph

1.3.8 The system shall allow users to search the details of the selected node from the wikidata

2. Stakeholders and Users

2.1 User Roles

2.1.1 The system shall support two user roles: Admin and Regular User

2.1.2 The system shall allow Regular Users to perform the operations mentioned in section 1

2.1.3 The system shall allow Admins to remove users and content when necessary

2.2 Project Leadership

2.2.1 The primary decision-maker for the project is the stakeholder (developer)

3. Platform Availability

3.1 Web Application Accessibility

3.1.1 The system shall be available as a web application.

3.1.2 The system shall be available only with the internet connection

4. Data Management

4.1 Structuring and Linking Information

4.1.1 The system shall utilize network-based or graph-based storage for connecting information.

4.1.2 The system shall resemble a crime-solving board, allowing users to progressively add and link findings.

5. Search and Filtering

5.1 Content Search and Tag Filtering

5.1.1 The system shall allow users to search for content on the home page.

5.1.2 The system shall allow users to navigate between search results.

5.1.3 The system shall notify users when no matching results are found.

6. Language Support

6.1 Supported Languages

6.1.1 The system shall be available in English only.

7. Error Handling

7.1 Display Informational Messages

7.1.1 The system shall display informational messages to users in case of errors, crashes, or downtime.

8. Deployment and Infrastructure

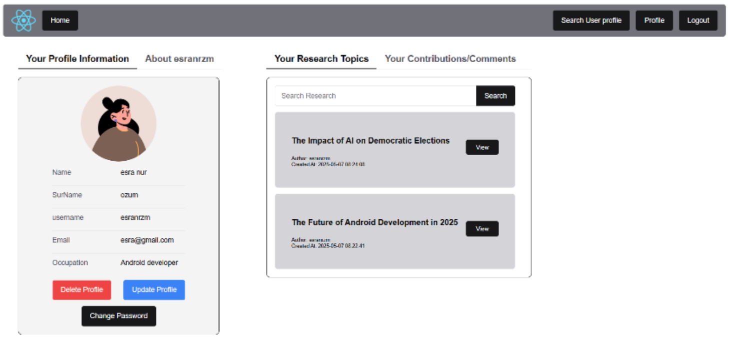
8.1 Cloud Deployment

8.1.1 The project must be deployed

8.1.2 The project must be Dockerized for easy deployment and scalability.

3. Design Documents

- Profile Page

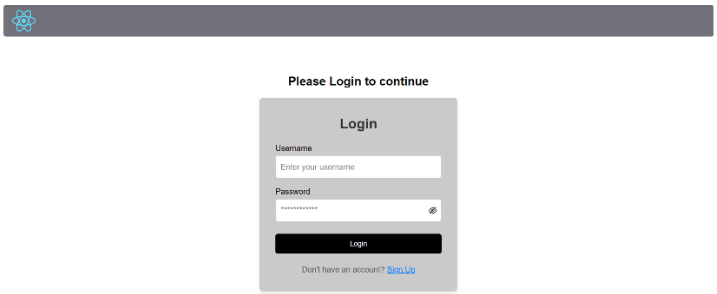


Name	esra nur
SurName	ozum
username	esranrzm
Email	esra@gmail.com
Occupation	Android developer

The Impact of AI on Democratic Elections
Author: esranrzm
Created At: 2025-05-07 08:24:08

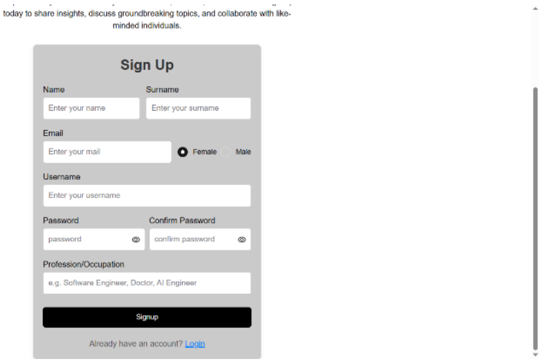
The Future of Android Development in 2025
Author: esranrzm
Created At: 2025-05-07 08:22:41

- Login Screen



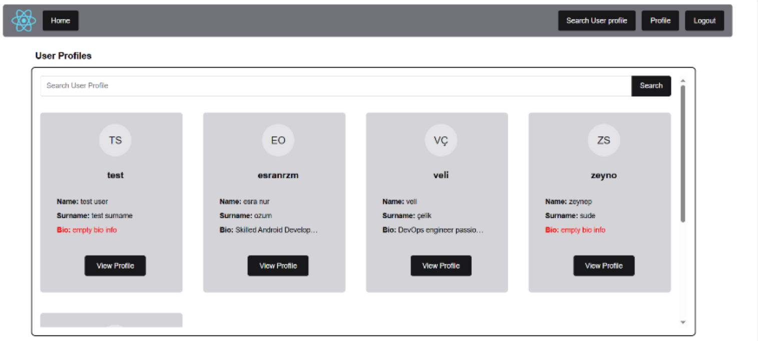
Password: ************

- Register Screen


today to share insights, discuss groundbreaking topics, and collaborate with like-minded individuals.

● Female ○ Male

- Search Other User Screen



User Profiles

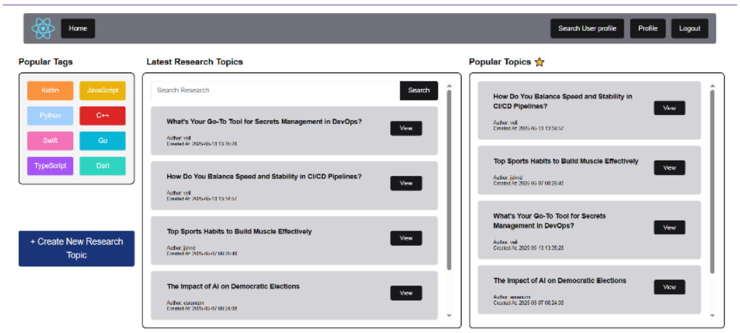
test
Name: test user
Surname: test surname
Bio: empty bio info

esranrzm
Name: esra nur
Surname: ozum
Bio: Skilled Android Develop...

veli
Name: veli
Surname: çelik
Bio: DevOps engineer passio...

zeyno
Name: zeynep
Surname: sude
Bio: empty bio info

- Home Page



Popular Tags

Latest Research Topics

What's Your Go-To Tool for Secrets Management in DevOps?
Author: veli
Created At: 2025-05-13 13:35:28

How Do You Balance Speed and Stability in CI/CD Pipelines?
Author: veli
Created At: 2025-05-13 13:34:57

Top Sports Habits to Build Muscle Effectively
Author: johnd
Created At: 2025-05-07 08:26:40

The Impact of AI on Democratic Elections
Author: esranrzm
Created At: 2025-05-07 08:24:08

Popular Topics ⭐

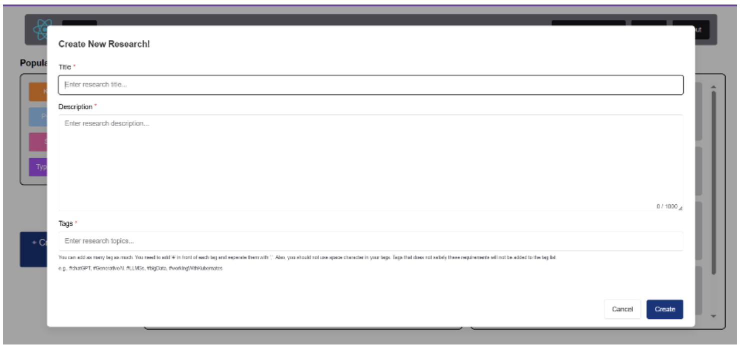
How Do You Balance Speed and Stability in CI/CD Pipelines?
Author: veli
Created At: 2025-05-13 13:34:57

Top Sports Habits to Build Muscle Effectively
Author: johnd
Created At: 2025-05-07 08:26:40

What's Your Go-To Tool for Secrets Management in DevOps?
Author: veli
Created At: 2025-05-13 13:35:28

The Impact of AI on Democratic Elections
Author: esranrzm
Created At: 2025-05-07 08:24:08

- Create Research screen



Create New Research!

Title *

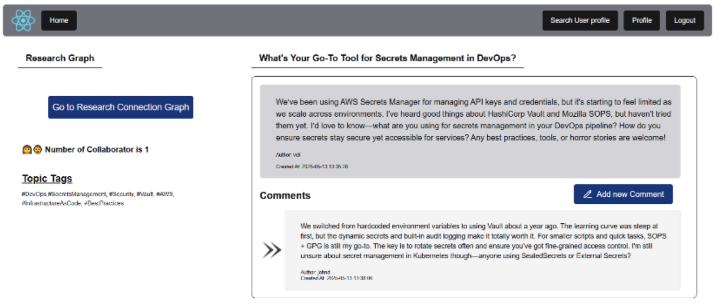
Description *

Tags *

You can add as many tag as much. You need to add '#' in front of each tag and seperate them with ','. Also, you should not use space character in your tags. Tags that does not satisfy these requirements will not be added to the tag list.

e.g. #chatGPT, #GenerativeAI, #LLMs, #bigData, #workingWithKubernetes

- Research Details



Research Graph

What's Your Go-To Tool for Secrets Management in DevOps?

We've been using AWS Secrets Manager for managing API keys and credentials, but it's starting to feel limited as we scale across environments. I've heard good things about HashiCorp Vault and Mozilla SOPS, but haven't tried them yet. I'd love to know—what are you using for secrets management in your DevOps pipeline? How do you ensure secrets stay secure yet accessible for services? Any best practices, tools, or horror stories are welcome!

Author: veli
Created At: 2025-05-13 13:35:28

👩👦 Number of Collaborator is 1

Topic Tags

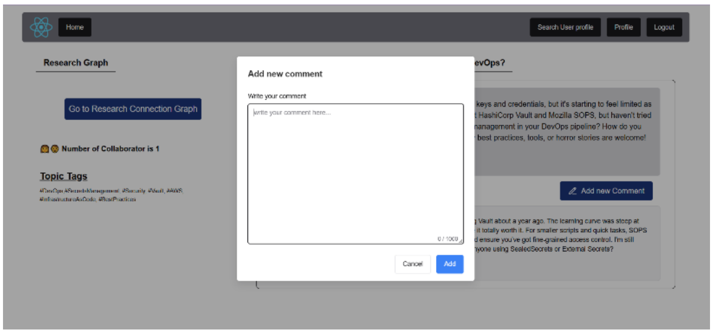
#DevOps,#SecretsManagement, #Security, #Vault, #AWS, #InfrastructureAsCode, #BestPractices

Comments

We switched from hardcoded environment variables to using Vault about a year ago. The learning curve was steep at first, but the dynamic secrets and built-in audit logging make it totally worth it. For smaller scripts and quick tasks, SOPS + GPG is still my go-to. The key is to rotate secrets often and ensure you've got fine-grained access control. I'm still unsure about secret management in Kubernetes though—anyone using SealedSecrets or External Secrets?

Author: johnd
Created At: 2025-05-13 13:38:06

- Add Comment Screen


Home
Search User profile
Profile
Logout
Research Graph
Go to Research Connection Graph
Number of Collaborator is 1
Topic Tags
#DevOps #SecretsManagement, #Security, #Vault, #AWS, #InfrastructureAsCode, #BestPractices
evOps?
Add new Comment
Cancel
Add

Add new comment

Write your comment

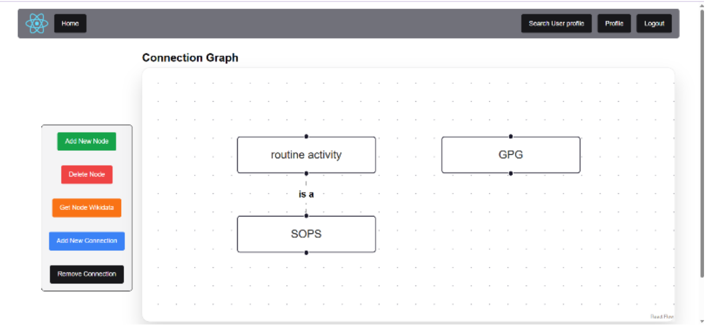
write your comment here...

0 / 1000

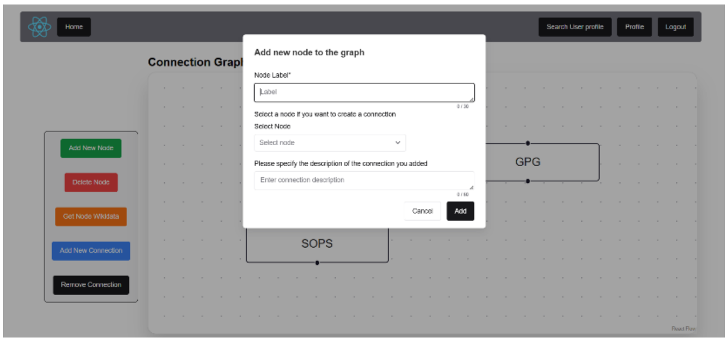
keys and credentials, but it's starting to feel limited as
t HashiCorp Vault and Mozilla SOPS, but haven't tried
nanagement in your DevOps pipeline? How do you
best practices, tools, or horror stories are welcome!

j Vault about a year ago. The learning curve was steep at
it totally worth it. For smaller scripts and quick tasks, SOPS
d ensure you've got fine-grained access control. I'm still
nyone using SealedSecrets or External Secrets?

- Graph Screen



- Add Node Screen



Add new node to the graph

Node Label*

Label

0 / 30

Select a node if you want to create a connection

Select Node

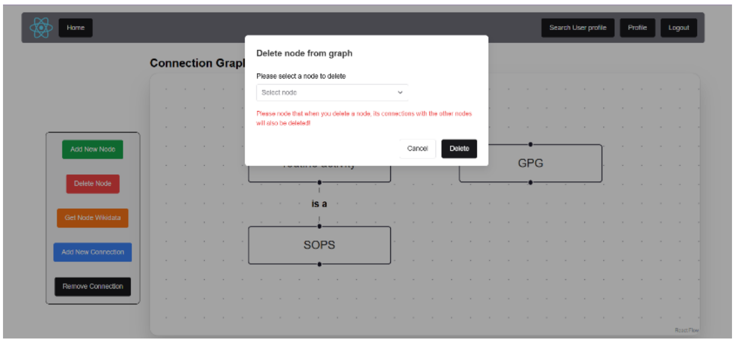
Select node

Please specify the description of the connection you added

Enter connection description

0 / 60

- Delete Node Screen



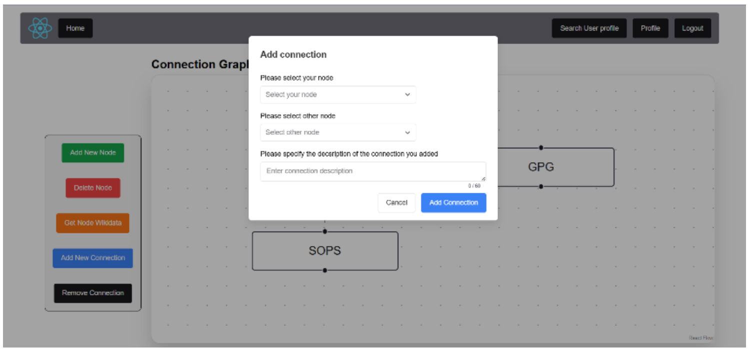
Delete node from graph

Please select a node to delete

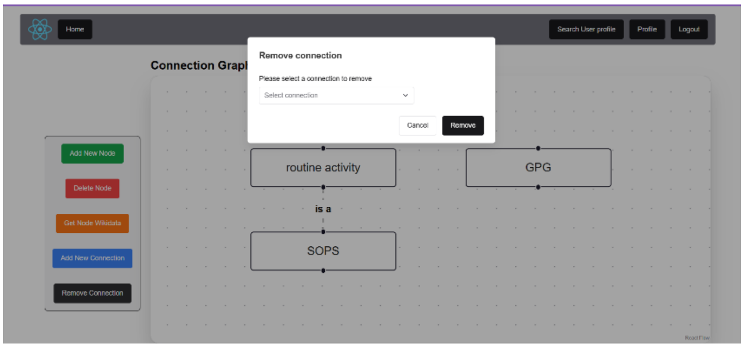
Select node

Please node that when you delete a node, its connections with the other nodes will also be deleted!

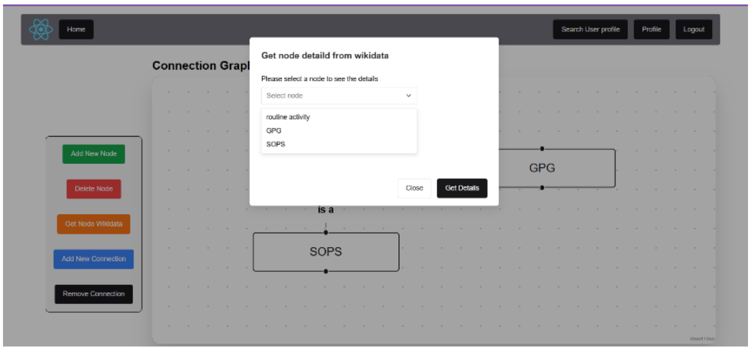
- Add Connection Screen



- Remove Connection Screen



- Get Wikidata Details Screen



4. Status of the project

- The system shall enable user to create a new account => completed
- The system shall enable users to login with their correct credentials => completed
- The system shall enable users to update their biography, username, password, name, and surname => completed
- The system shall allow users to delete their account => completed
- The system shall show meaningful error messages to the users when a problem occurs in the system, or an incorrect data entrance happens => completed
- The system shall require users to provide profile information including: => partially completed

  - Skills
  - Name
  - Surname
  - Email
  - Password
  - Age => not exist
  - Gender
- The system shall allow users to create a new topic => completed
- The system shall allow users to update their research topic details such as description, tags, and title => completed
- The system shall allow users to delete their research topic => completed
- The system shall allow users to comment on research => completed
- The system shall allow users to update their comment => completed
- The system shall allow users to delete their comment => completed
- The system shall allow users to see the research details for the selected research => completed
- The system shall enable users to search among the research => completed
- The system shall display only 10 matching results per page => partially completed (system shows all matching results
- The system shall enable tag-based filtering for research projects => not completed
- The system shall allow users to add a node to the graph => completed
- The system shall allow users to delete a node from the graph => completed
- The system must prevent deletion of a node that is not created by that user
- The system shall enable users to add a connection between the node that they created and any other node in the graph => completed
  - The system must prevent a connection that has the target and source node as same (loop edge)
  - The system must show an error message if the user selects same node for target and source node
  - The user must enter a label to be able to create a connection between nodes
  - The system shall enable users to only see their own nodes in target node list
- The system shall allow users to delete a connection that they created before => completed
- The system shall enable users to zoom in and zoom out the graph => completed
- The system shall allow users to see the graph => completed
- The system shall allow users to search the details of the selected node from the wikidata => partially completed (fetching data with wikidata endpoints sometimes times out and does not reflect the property meaning or entity name)
- The system shall support two user roles: Admin and Regular User => completed
- The system shall allow Regular Users to perform the operations mentioned in section 1 => completed
- The system shall allow Admins to remove users and content when necessary => completed
- The primary decision-maker for the project is the stakeholder (developer) => completed
- The system shall be available as a web application => completed
- The system shall be available only with the internet connection => completed
- The system shall utilize network-based or graph-based storage for connecting information => completed

- The system shall resemble a crime-solving board, allowing users to progressively add and link findings => completed
- The system shall allow users to search for content on the home page => completed
- The system shall allow users to navigate between search results => completed
- The system shall notify users when no matching results are found => completed
- The system shall be available in English only => completed
- The system shall display informational messages to users in case of errors, crashes, or downtime => partially completed (no notification for downtime issues)
- The project must be deployed => completed
- The project must be Dockerized for easy deployment and scalability => completed

5. Status of Deployment

The project is deployed and can be reached via the following link https://swe573-spring25.onrender.com/

Components of the project are also dockerized and used for the deployment process.

6. Installation Manual

Make sure the following are installed on your system:

- Node.js (v16 or above recommended)
- Python (v3.8+)
- pip (Python package manager)
- Git
- Docker & Docker Compose (Optional)

Installation Guide for cloning the repo

- Clone the Repository and go to the project directory
  - git clone https://github.com/esranrzm/SWE573_SPRING25.git
  - cd the_location_of_the_project
- Navigate to the backend folder:
  - Cd backend
- Create and activate a virtual environment:
  - python -m venv venv
  - venv\Scripts\activate
- Install backend dependencies:
  - pip install -r requirements.txt
- Create a `.env` file in the `backend/` directory and put the followings inside it:
  - SECRET_KEY=your_secret_key_here
  - DB_URL=your_database_url_here
- Run the Flask server:
  - flask run
- Navigate to the frontend folder:

    - cd ../frontend
- Install frontend dependencies:
    - npm install
- Run the development server:
    - npm run dev

Installation Guide running docker

- If you prefer to run the app using Docker, make sure Docker and Docker Compose are installed.
- You also need to add .env file to the main directory for docker db credentials
- Run:
    - docker-compose up –build

7. User Manuel

To be able to use the system, you need to have a valid username and password. If you do not have those values, you need to go to the register page and create a new user. After creating a valid user, you can do all operations mentioned in the requirements section.

- How to create research?
    - You need to go to the home page
    - Click on “+ Create New Research Topic” button
    - To complete the operation, you need to enter all required information and click on create button
    - Once it is created, you will see a message at the top of the page and your research will be seen on the research list on home page
    - If you want to see the details of it, you can click on “view” button
- How to add a comment?
    - You need to go to specific research
    - Once you see the details page, you need to click on “add new comment” button below the research description section.
    - You need to fill the opened dialog and click on “add” button to add your comment
    - You will see a message on top of the page and your comment will shown in the comments section
- How to see the graph of the research
    - You need to go to specific research
    - Once you see the details page, you need to click on “Go to Research Connection graph” button
    - You will be directed to the graph screen and see the latest content
- How to add a node to the graph?
    - You need to click on add new node button on the button panel on the graph page
    - You need to enter the label for the node that you want to add
    - If you also want to connect it with an already existed node, you have to select the node and also enter the description for the connection you want to create
    - Once you add your node, you will see a message on top of the page and your node will be seen on the updated graph

- How to delete a node from the graph?
    - Note that you can only delete your own nodes and connections
    - To delete a node, you need to click on “delete node” button on the screen
    - You need to select the node you want to delete form the list. If you have not created a node on the graph, you will see an empty list.
- How to create a connection between nodes?
    - Note that you can only create connections between one of your nodes to the any other node on the graph
    - You need to select your node and the second node in the shown dialog. Entering a description is optional. You can choose to not add a description for your connection
    - Once you finished your will see a message on top of the page and your connection will be shown in the updated graph in 2 seconds
- How to remove a connection from the graph?
    - Note that your can only remove the connections that you created before
    - You need to click on “remove connection” button on the screen
    - You need to select the connection that you want to remove from the list
    - If you have not created a connection on the graph, you will see an empty list
- How to handle profile operations?
    - You can handle all profile operations in your profile page
    - You can update your updatable profile information by using the buttons, update profile, delete profile, and change password.
- How to search and see other users?
    - There is a button named as “Search User profile” on the navigation bar above the screen.
    - You need to click on it
    - You can use the search box to write your search text
    - Once you write it, you need to click on search button
    - If you want to remove the search filter, you need to remove the text and click on search button again
    - If you want to see the details of the user profile, you can click on “view profile” button

8. Test Results

Unit tests are implanted for the following operations and run with sample data:

- Create user profile => test passed
- Update user profile => test passed
- Delete user profile => test passed
- Create research => test passed
- Update research => test passed
- Delete research => test passed
- Create comment => test passed
- Update comment => test passed
- Delete comment => test passed

All requirements that have implemented successfully have tested and passed all the tests without any problem

9. References

- https://www.chakra-ui.com/ is used for UI components
- https://render.com/docs/deploys is used for deployment related problems
- AWS is used for DB storage