# CAB230 Assignment 2 --- Client Side

Yongzhi ZHOU

N10044329

09/05/2024

# Contents

Introduction....3
Purpose & description....3
Completeness and Limitations....3
Use of End Points....4
/countries....4
/volcanoes....5
/volcanoes/{id}....6
/user/register....7
/user/login....8
Modules Used....9
react-router-dom....9
Ag-grid-react....9
Bootstrap/reactstrap....9
Pigeon-maps....9
React-chartjs-2....9
Application Design....10
Navigation and Layout....10
Usability and Quality of Design....15
Accessibility....16
Technical Description....17
Architecture....17
Test Plan....18
Difficulties / Exclusions / unresolved & persistent errors /....19
Extension (Optional)....20
User guide....21
References....24
Appendix A....25

## Introduction

### Purpose & description

This website is designed to provide users with specific information about volcanoes around the world, with the aim of providing a good interactive and visual experience. To provide users with a good interactive experience as well as a good visual experience, a more user-friendly modular design has been used. According to Richard H. Hall and Patrick Hanna (2004), black text on a white background is the best requirement for readability. For this reason, I have set the background colour of the site to white, which is more user oriented and increases the user experience.

The data presented in this website is represented in charts and I have chosen Pie for this purpose so that the data can be displayed more efficiently.

### Completeness and Limitations

The navigation, login, registration, country and volcano information, and the charts are all working perfectly and smoothly. However, there is a small problem with the switching between the "login" and "logout" buttons, after logging in the page needs to be refreshed before the "login" to switch to "logout". The location of the volcano is rendered smoothly and perfectly in the map component API function.

**Use of End Points**

*/countries*

*/volcanoes*

Country: China Populated Within: 100km search

| Name | Region | Subregion |
|---|---|---|
| Arxan-Chaihe | Kamchatka and Mainland Asia | China (eastern) |
| Changbaishan | Kamchatka and Mainland Asia | China (eastern) |
| Hainan Volcanic Field | Philippines and SE Asia | Southeast Asia |
| Honggeertu | Kamchatka and Mainland Asia | China (eastern) |
| Kunlun Volcanic Group | Kamchatka and Mainland Asia | China (western) |
| Jingbo | Kamchatka and Mainland Asia | China (eastern) |
| Longgang Group | Kamchatka and Mainland Asia | China (eastern) |
| Keluo Group | Kamchatka and Mainland Asia | China (eastern) |

Page Size: 100 1 to 10 of 10 Page 1 of 1

*/volcanoes/{id}*

*Unlogged-in status

## Hainan Volcanic Field

country: China

region: Philippines and SE Asia

subregion: Southeast Asia

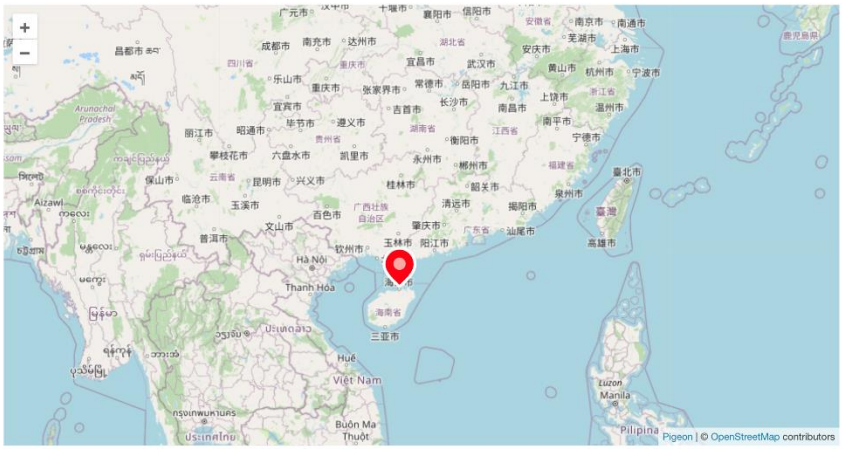
last_eruption: 1933 CE

summit: 196

elevation: 643

latitude: 19.9050

longitude: 110.2290




*Login status

## Hainan Volcanic Field

country: China

region: Philippines and SE Asia

subregion: Southeast Asia

last_eruption: 1933 CE

summit: 196

elevation: 643

latitude: 19.9050

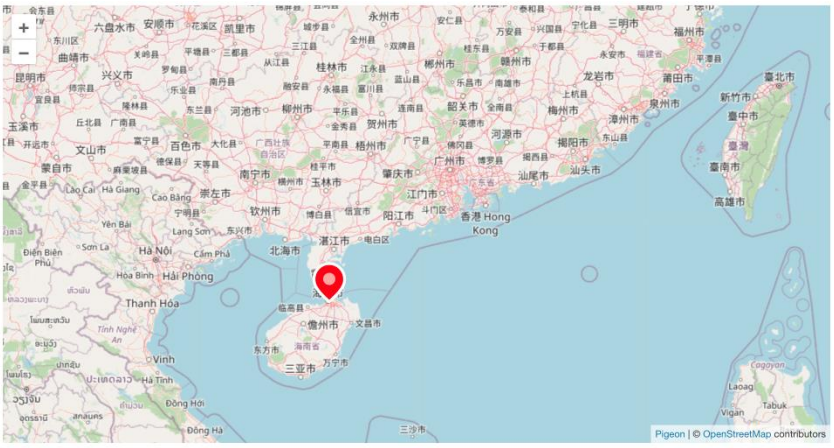
longitude: 110.2290

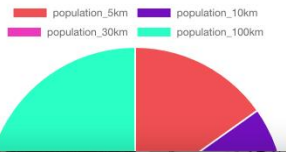
population_5km: 1731229

population_10km: 1731229

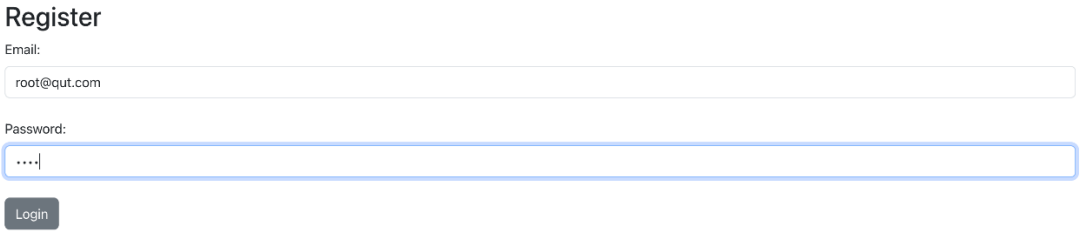
population_30km: 1731229

population_100km: 6213231

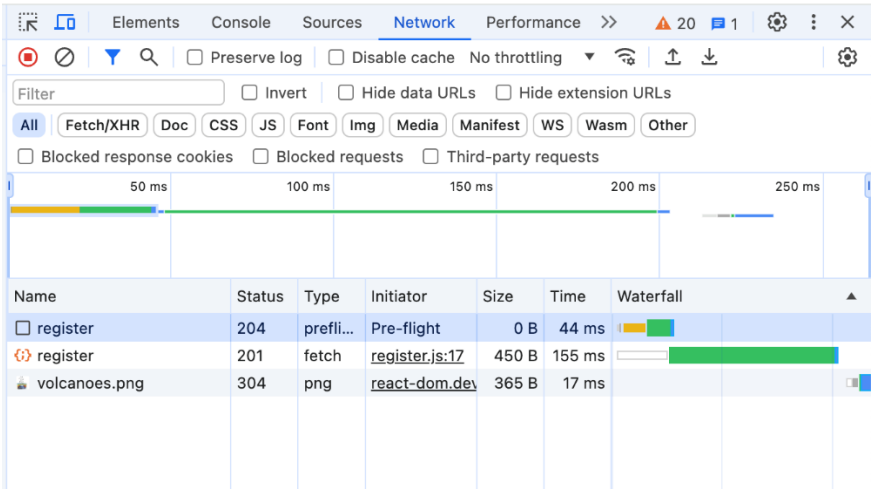


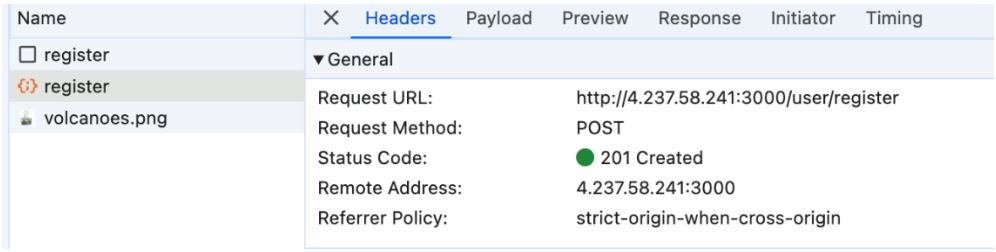



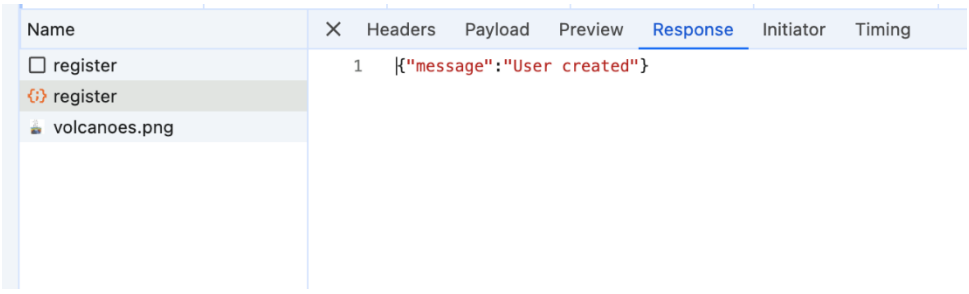
*/user/register*



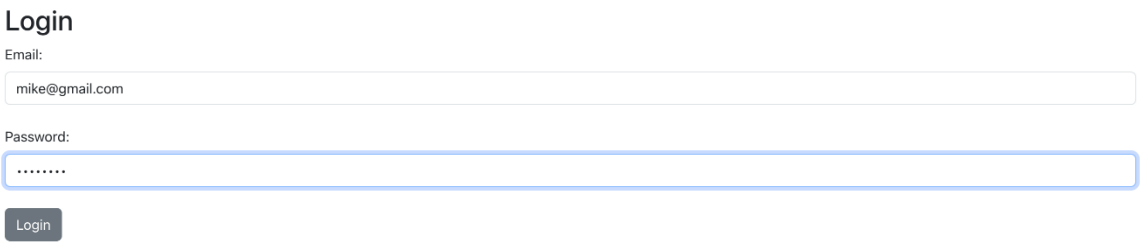

*Information of Inspect



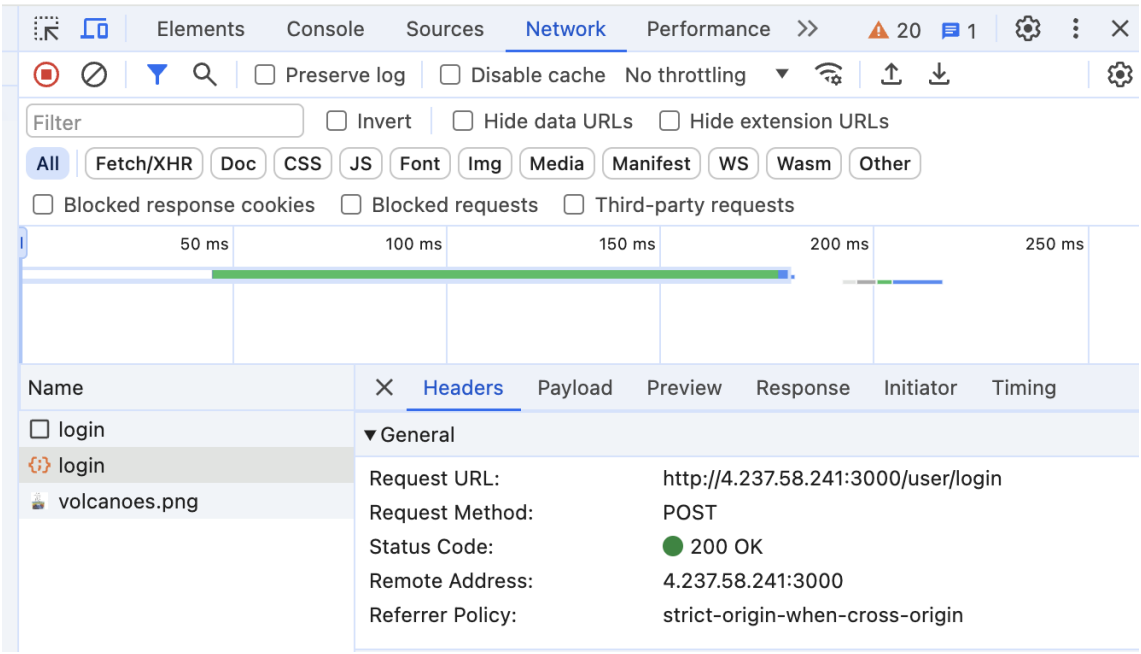



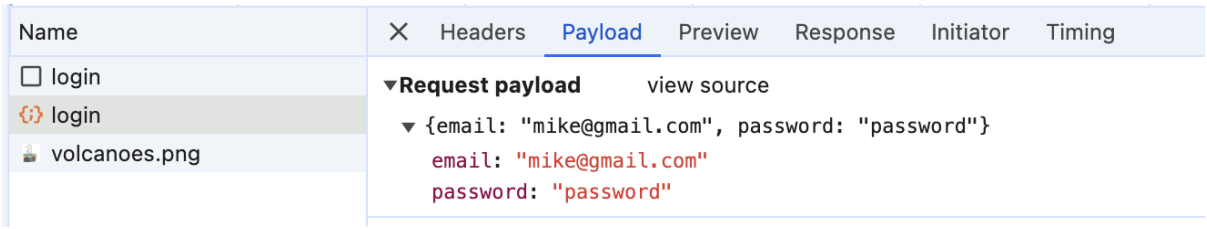



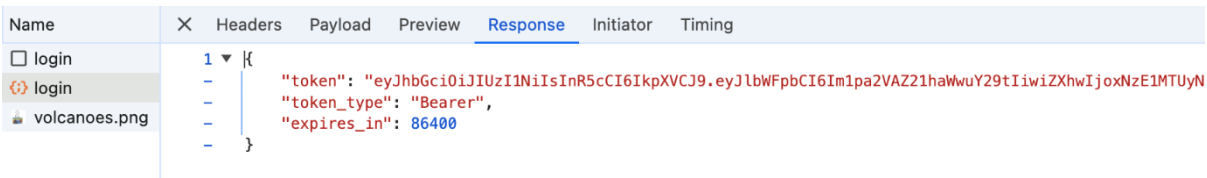
*/user/login*




*Information of Inspect



**Modules Used**

*react-router-dom*

Module provides the basic functionality needed for routing navigation and page jumps in single page applications.

https://reactrouter.com/en/6.23.0/start/tutorial

*ag-grid-react*

Module provides advanced data networking components and rich functionality such as sorting, filtering, grouping, virtual scrolling and cell editing.

https://www.ag-grid.com/react-data-grid/getting-started/

*bootstrap/reactstrap*

The Bootstrap module provides a set of ready-made CSS styles and JavaScript plugins for quickly building beautiful interfaces and interactions. And Reactstrap is a React component library for Bootstrap, which encapsulates Bootstrap components into React components for easy use in React applications.

https://reactstrap.github.io/?path=/story/home-installation--page

https://getbootstrap.com/docs/5.1/getting-started/introduction/

*pigeon-maps*

Module provides the basic components and APIs for integrating map functionality

https://pigeon-maps.js.org/docs/

*react-chartjs-2*

Module is used in the Chart.js wrapper library for React applications and can be used to create various types of charts, such as line charts, bar charts, pie charts, etc. Developers can easily use the functionality provided by Chart.js in React components, including configuring chart data, styles, and interactive behaviour.

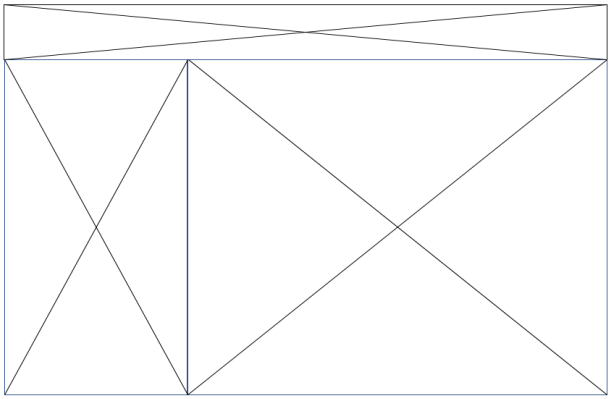
https://react-chartjs-2.js.org/

**Application Design**

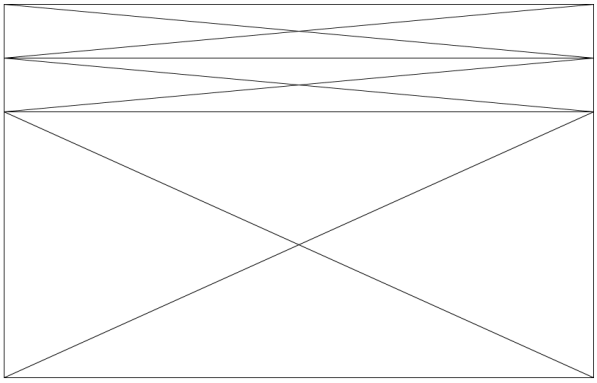
## Navigation and Layout

When designing this website, I focused on simplicity, clarity, small user experience and the ornamental nature of the interface. Based on my past experience in designing websites, I used the Reactstrap module, which is based on Bootstrap. It's easy to use, it divides the whole width of the page into 12 parts, you can also define the size of the module you want, and in the corresponding module to place the components or data you want.

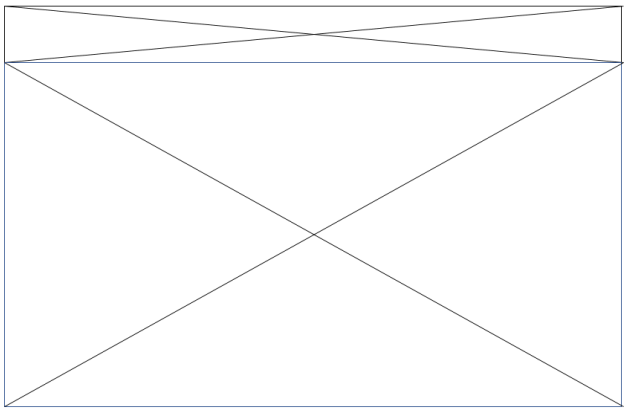
**Sketch Groups:**



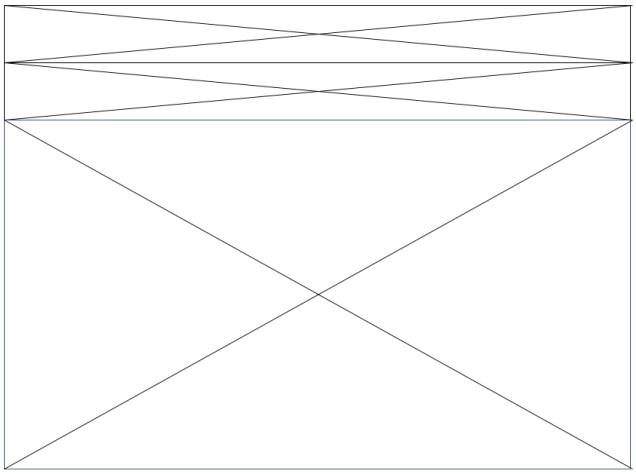
*Sketch 1*



*Sketch 2*

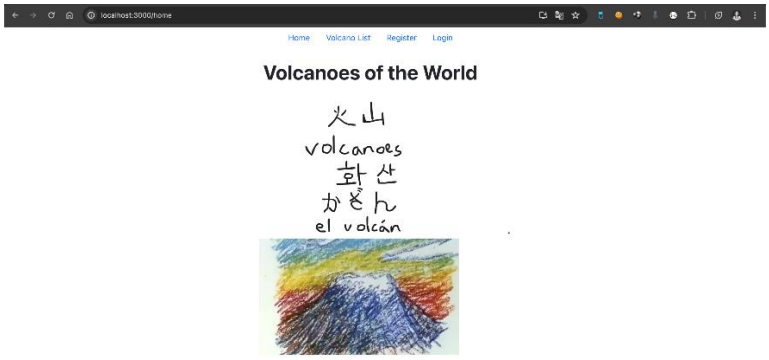


*Sketch 3*

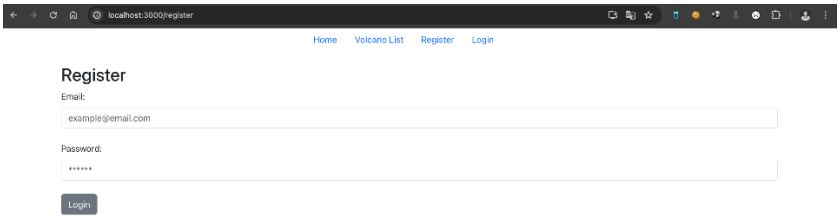


*Sketch 4*

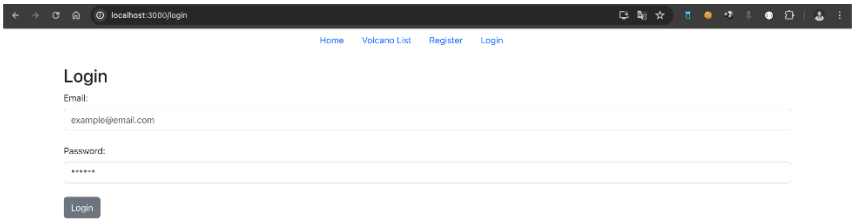
**Application Screen Flow Process:**



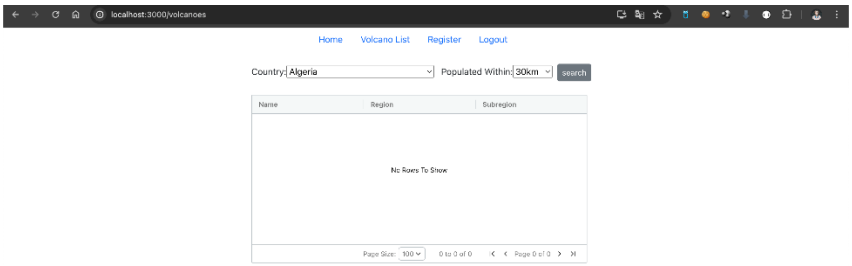

*1.Home Page*



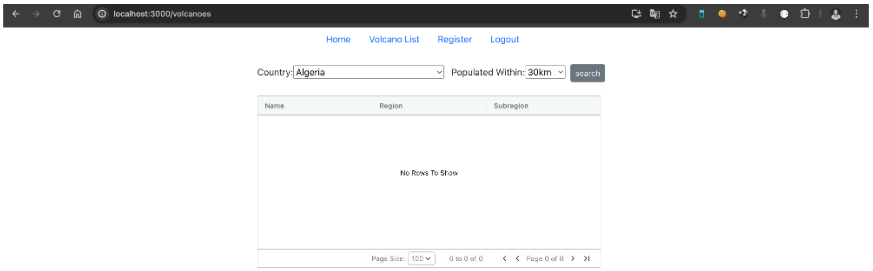

*2.Click "Register" button into Register Page*

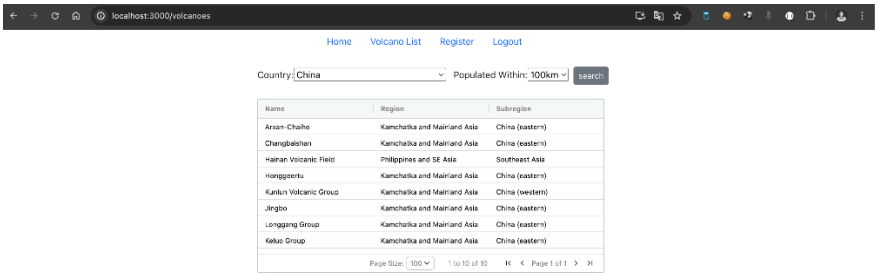



*3.Click "Login" button to login account when user finish creating account.*

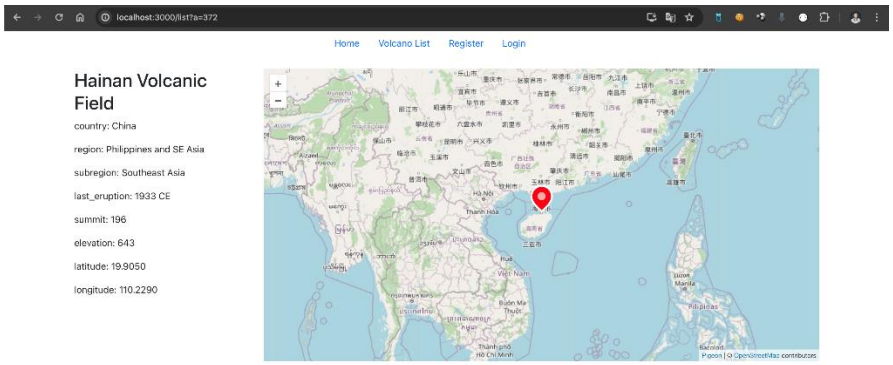


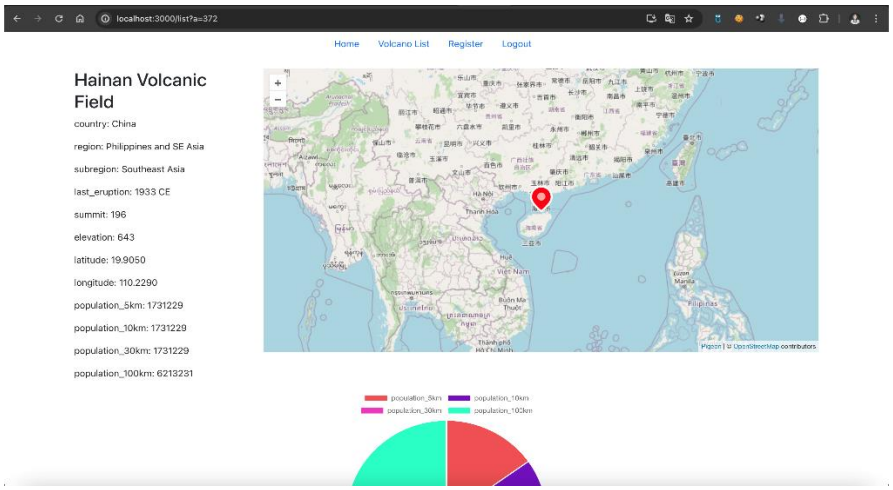

*4.Click "Volcano List" enter Volcano Page*




*5. Select the corresponding option and click "Search" to present the data information.*




*6. Present the user with specific information about the volcano as well as a chart.*

## Usability and Quality of Design

The following comments were made about the design of the site:

The display of data is well organized, and the content is easy to read and highly readable. The layout of the site is clear, with no complex or confusing interfaces, and meets the criteria for simplicity. The navigation interface is clear and intuitive, with good accessibility and predictability. When a user visits the site for the first time, he or she knows that the navigation interface is clear and does not need to be prompted. This is in line with the user-oriented qualities of the design principles. The functionality of the site is basically in line with the user's expectations in terms of the basic functionality of the application, after all, the main task of the job is to request the corresponding API interface and then return or modify the corresponding data. The visual design is largely consistent, with a small use of fonts and setting font sizes, and little to no colour. The simplicity and readability of the site is high. The design is highly usable, as the user simply selects their country and number, and the data is displayed quickly and easily.

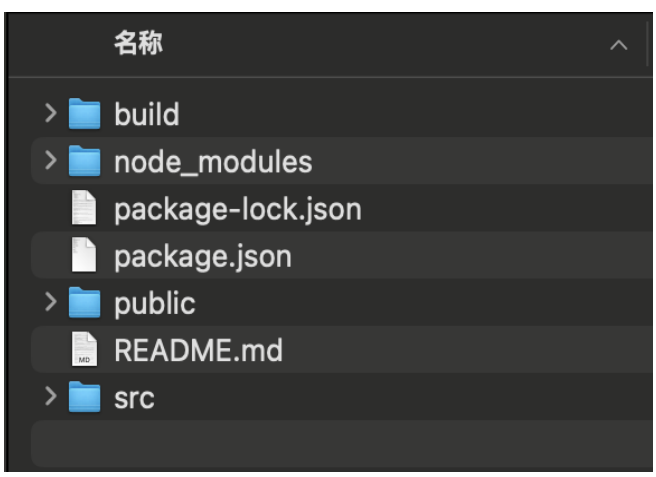
Accessibility

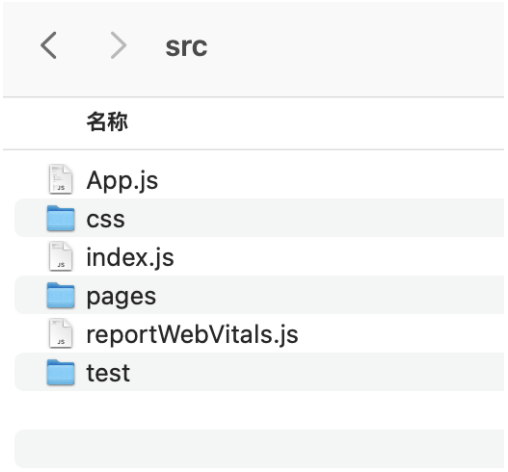
| In General | YES | NO | N/A |
|---|---|---|---|
| Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). This includes images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video. | √ | | |
| Ensure that all information conveyed with colour is also available without colour, for example from context or markup. | √ | | |
| Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions). | √ | | |
| Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document. | √ | | |
| Ensure that equivalents for dynamic content are updated when the dynamic content changes. | √ | | |
| Until user agents allow users to control flickering, avoid causing the screen to flicker. | √ | | |
| Use the clearest and simplest language appropriate for a site's content. | √ | | |
| **And if you use tables** | | | |
| For data tables, identify row and column headers. | √ | | |
| For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells. | √ | | |
| **And if you use frames** | | | |
| Title each frame to facilitate frame identification and navigation. | √ | | |

**Technical Description**

## Architecture

***build/:*** Used to store application code and resources in the development environment
***node_modules/:*** Stores various third-party libraries and their dependencies
***package-lock.json:*** Contains details of all direct dependencies and their dependencies in the project, including version numbers, dependencies and where they are installed.
***package.json:*** Contains application dependencies and script configurations
***public/:*** Contains static files that don't need to be processed during the build process, such as HTML files and icon files
***README.md:*** Includes an introduction to the application and instructions for use
***src/:*** Main code directory for the entire application




***css/:*** The main component of the application, which usually contains the application's routing settings and top-level structure.
***pages/:*** Page files, which hold the different pages or attempted components of the application.
***index.css:*** Used to store the global style of the application. It is usually used to define the basic style rules of the whole application, including page layout, font style, colour, button style and other global style settings.
***index.js:*** The entry point to the application, where the root component is rendered into the DOM.
***reportWebVitals.js:*** For reporting on website performance metrics.
***test/:*** Files used to set up the test environment during testing and store the application's test files for testing components and features.

## Test Plan

| Task | Expected Outcome | Result | Screenshot/s (Appendix A) |
|---|---|---|---|
| Click "Home" to back home page | Back to Home Page | PASS | 1 |
| Click "Volcano List" to view info | Display the "volcano list" page. | PASS | 2 |
| Click "Register" to view register form | Navigate to register page | PASS | 3 |
| Click "Login" to view login form | Navigate to login page | PASS | 4 |
| Select "Country" and "Populated Within" data to search | Display the corresponding country information | PASS | 5 |
| Click on a column in the table to display information and map coordinates of the corresponding country | Displays specific information and coordinates of the corresponding country and its exact location on the map | PASS | 6 |
| Enter username and password to complete the login. | Complete Login | PASS | 7 |
| Enter username and password to complete the registration. | Complete Registration | PASS | 8 |
| Once you have logged in, you can search for the corresponding country, and it will show you the "population" value of the corresponding country, as well as the chart. | Displays all "population" values as well as specific charts. | PASS | 9 |

## Difficulties / Exclusions / unresolved & persistent errors /

The main obstacle I encountered was the programming language (JavaScript), although this is not my first time to face JavaScript, even though I have a lot of experience in programming languages such as: java, python, etc. But for this language I have a lot of unfamiliarity and uncertainty, for example, for the different data types of processing functions and the various functions of the method, for these obstacles I first took measures: to go to the W3C website, the processing functions of each data type to learn, and then use the local Visual Studio Code for experimentation. For the different function methods, I googled, went to the corresponding developer's website or went to YouTube to find and watch the corresponding solutions. Based on these instructions, I was able to complete the assignment without any problems. However, there were still some issues that could not be resolved, for example, when I finished logging in, my ideal state was that the "Login" button in the navigation bar would instantly change to a "Logout" button, but the reality was that I needed to do a site refresh to get my ideal state.

The corresponding code for solving the problem is as follows:

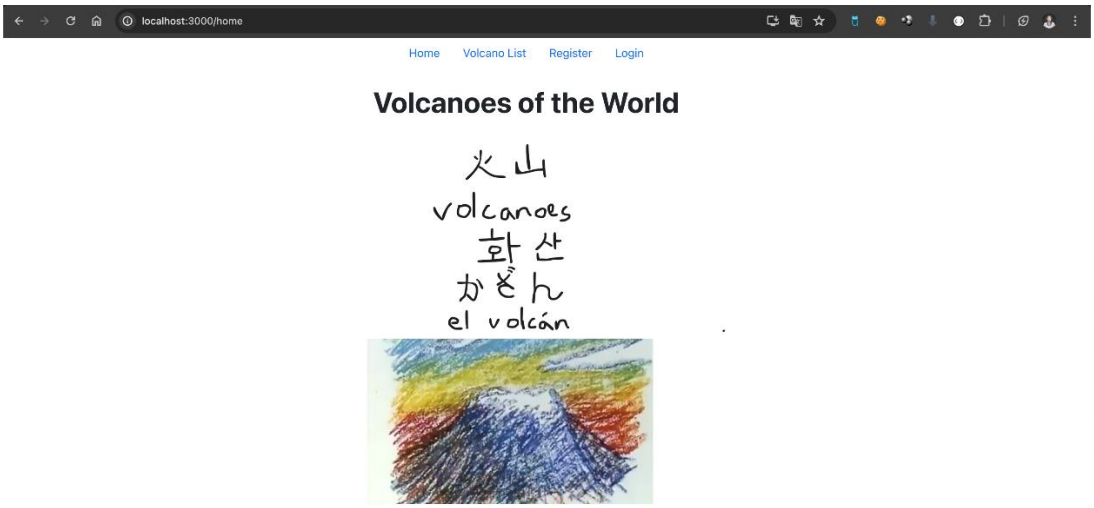
```
<NavItem>
  {
  (localStorage.getItem('token')!=null)?
  (<NavLink active onClick={deleteToken} href="#">Logout</NavLink>):
  (<NavLink active  href="/login">Login</NavLink>)
  }
</NavItem>
```

**Extension (Optional)**

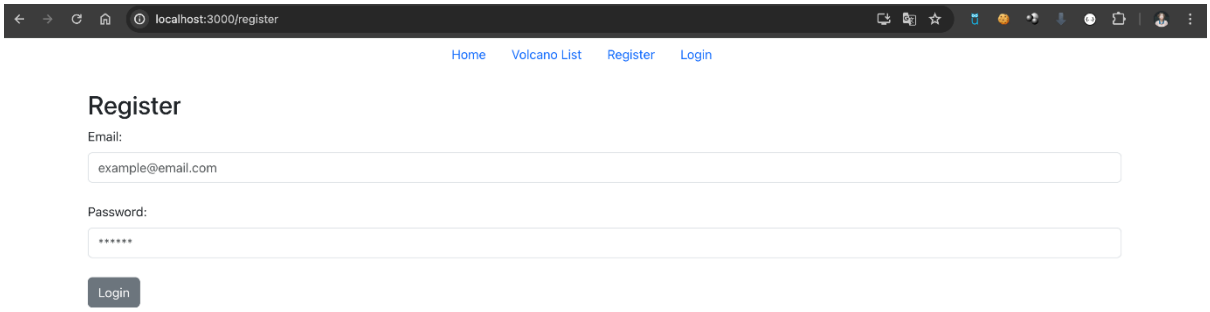
If there is enough time and enough information, I hope that when the user clicks on the corresponding volcano information. I would like to present not only the specific information about the volcano, but also a map of the volcano. I would like to present the volcano's surroundings in the form of a 3D image or 3D video. I want the user to feel like they are in the volcano when they click on the volcano.

**User guide**

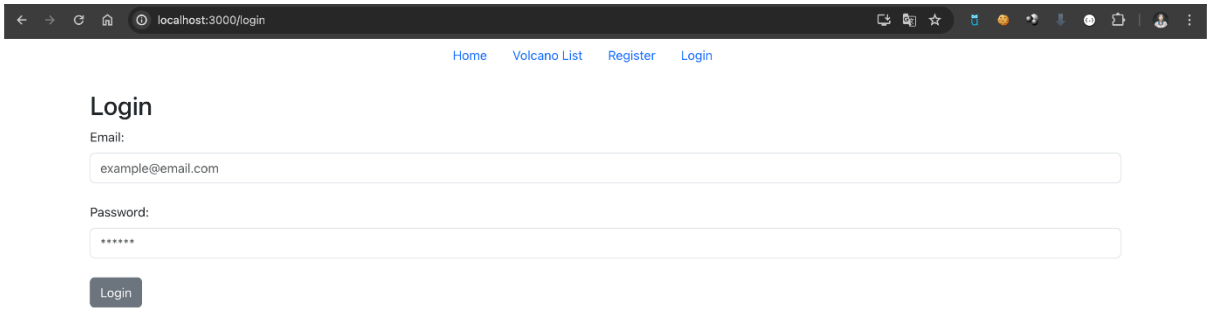
1. Enter the website address and go to the website



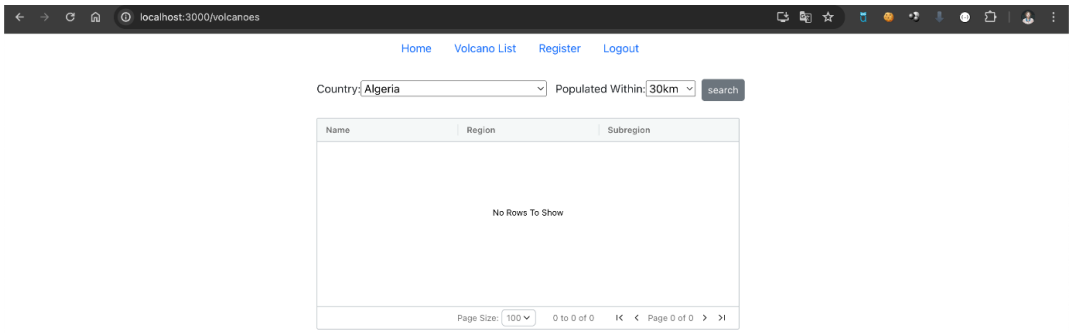

2. If you are a new user, go to the "Register" page and register as a new user.

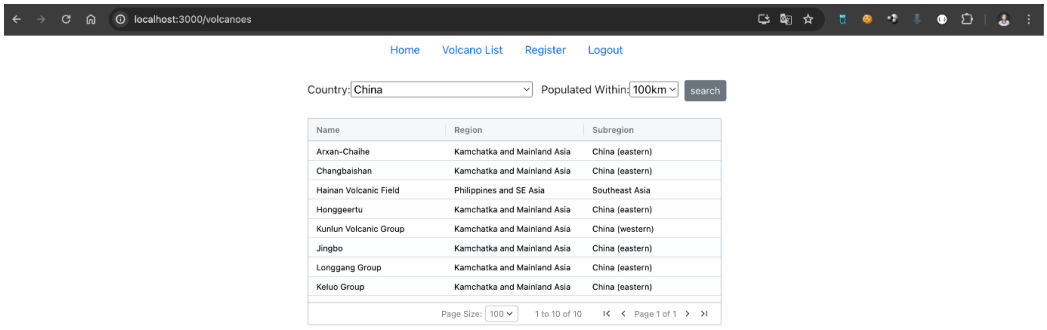


3. After completing the registration process, log in.

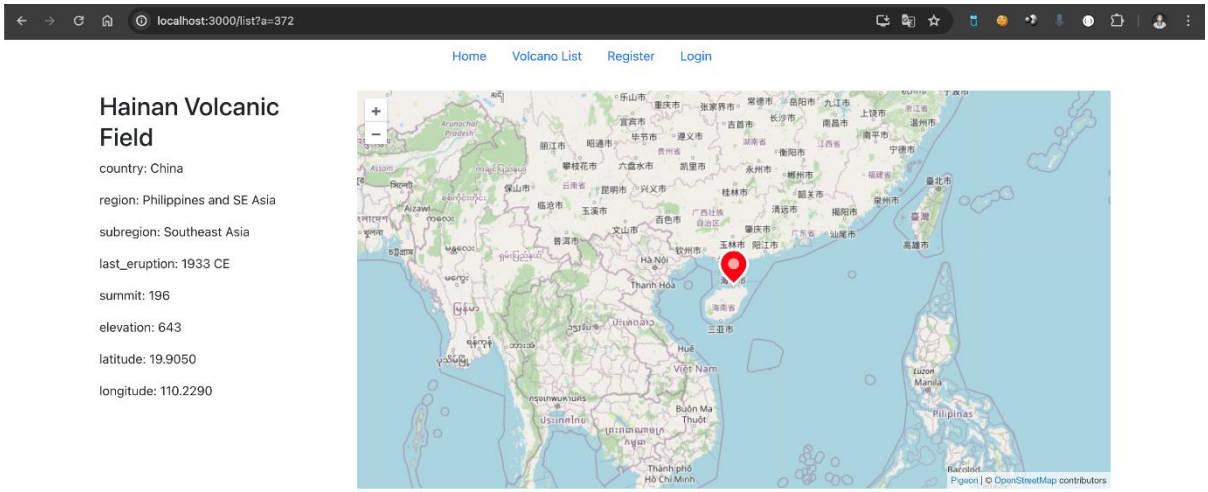



4. Click on "Volcanoes List" and select the country and populated within.

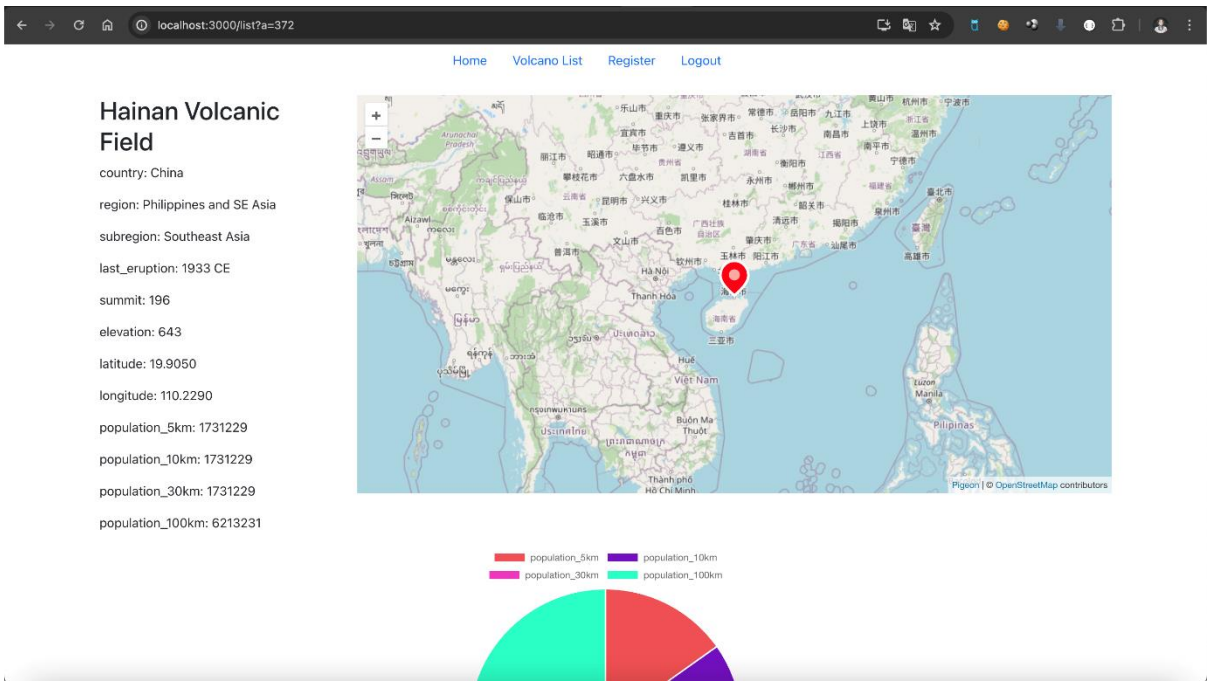


6. Select any of the columns to enter the corresponding information.



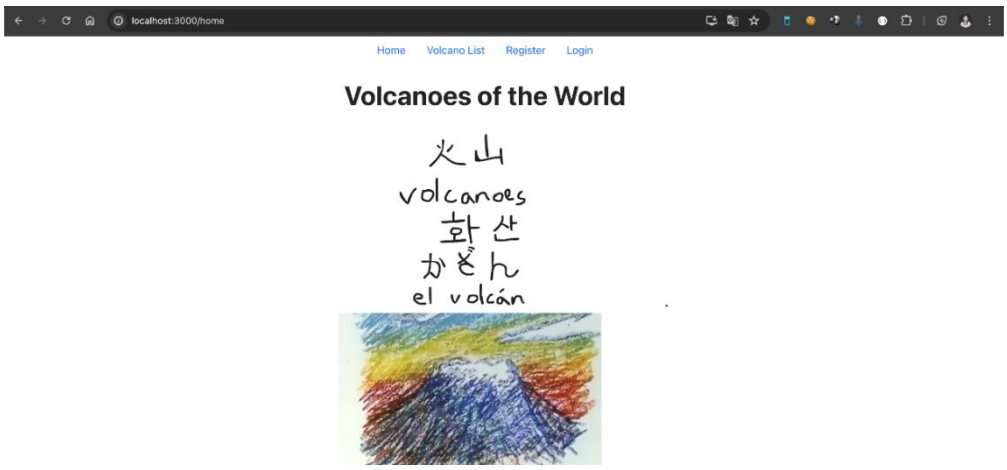

7. If you want to see more information and charts, you need to be logged in to see them.



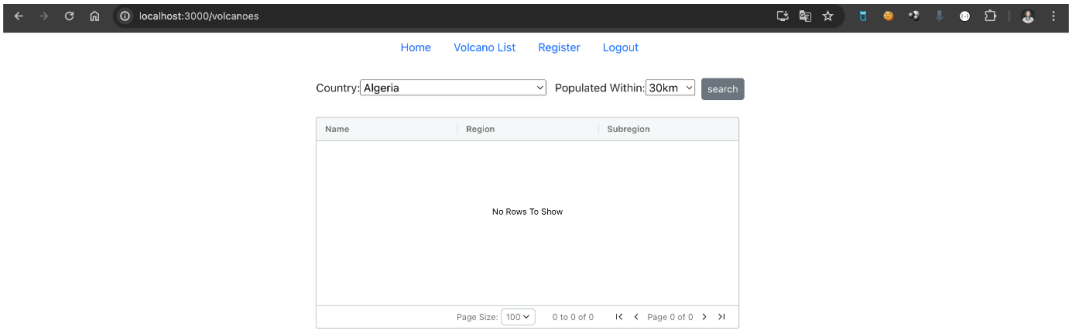
**References**

*Hall, R. H., & Hanna, P. (2004). The impact of web page text-background colour combinations on readability, retention, aesthetics and behavioural intention. Behaviour & Information Technology, 23(3), 183–195. https://doi.org/10.1080/01449290410001669932*

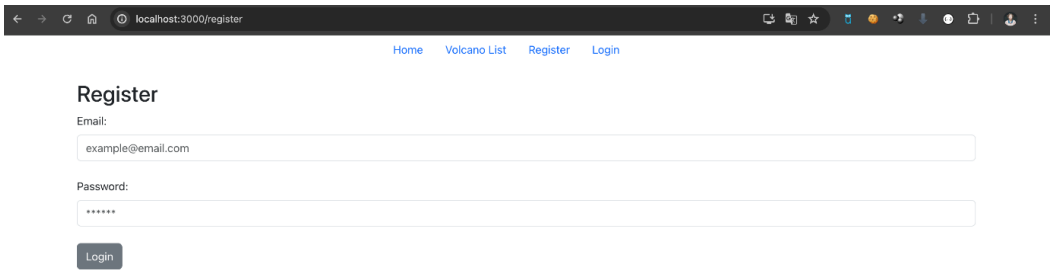
**Appendix A**



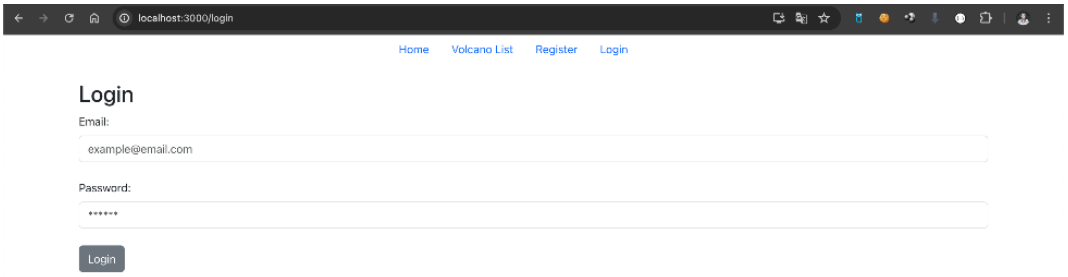

*1*



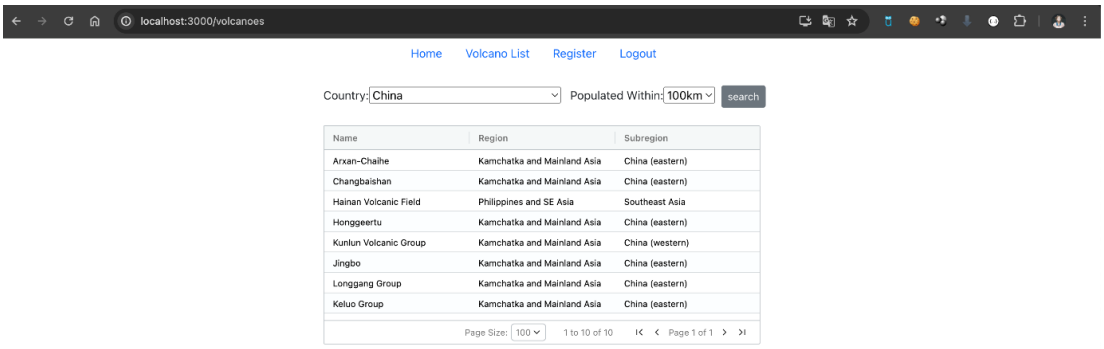

*2*


localhost:3000/register
Home
Volcano List
Register
Login
Register
Email:
example@email.com
Password:
******
Login

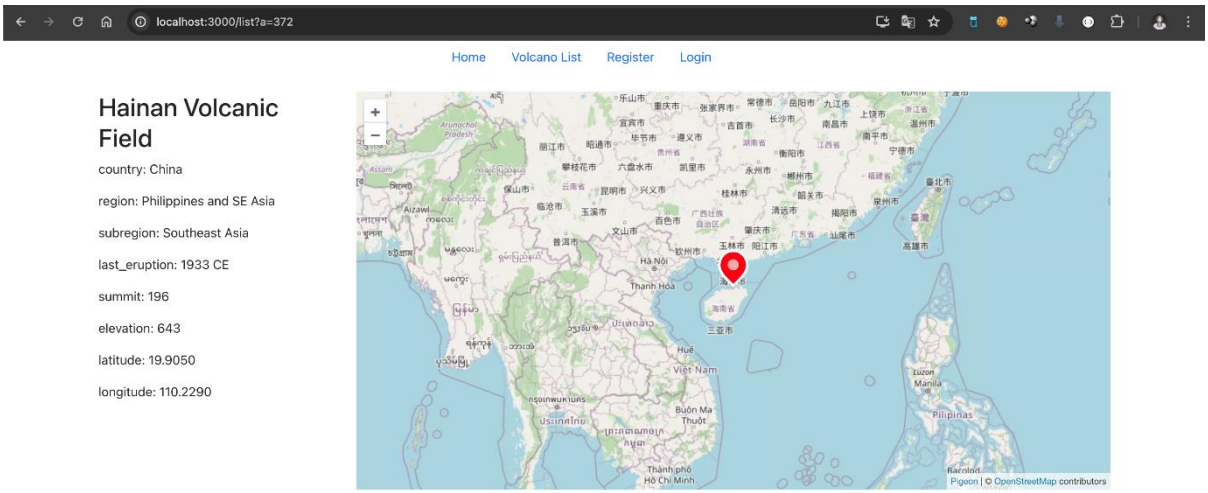
*3*


localhost:3000/login
Home
Volcano List
Register
Login
Login
Email:
example@email.com
Password:
******
Login

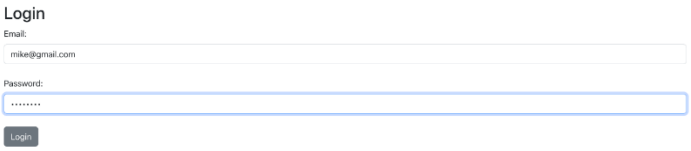
*4*

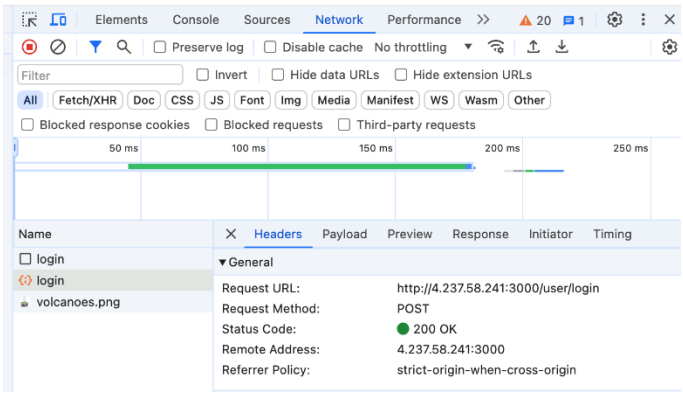

localhost:3000/volcanoes
Home
Volcano List
Register
Logout
Country: China
Populated Within: 100km
search
Name
Region
Subregion
Arxan-Chaihe
Kamchatka and Mainland Asia
China (eastern)
Changbaishan
Kamchatka and Mainland Asia
China (eastern)
Hainan Volcanic Field
Philippines and SE Asia
Southeast Asia
Honggeertu
Kamchatka and Mainland Asia
China (eastern)
Kunlun Volcanic Group
Kamchatka and Mainland Asia
China (western)
Jingbo
Kamchatka and Mainland Asia
China (eastern)
Longgang Group
Kamchatka and Mainland Asia
China (eastern)
Keluo Group
Kamchatka and Mainland Asia
China (eastern)
Page Size: 100
1 to 10 of 10
Page 1 of 1

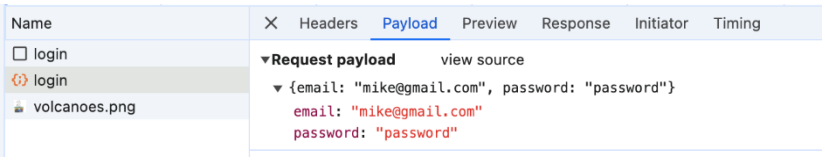
*5*


localhost:3000/list?a=372
Home
Volcano List
Register
Login
Hainan Volcanic Field
country: China
region: Philippines and SE Asia
subregion: Southeast Asia
last_eruption: 1933 CE
summit: 196
elevation: 643
latitude: 19.9050
longitude: 110.2290
Pigeon | © OpenStreetMap contributors

*6*

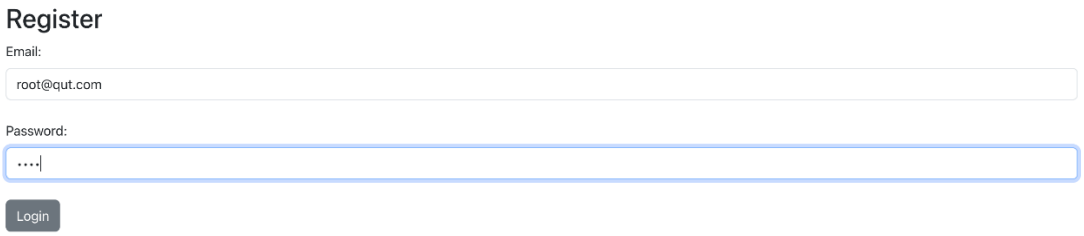

Login
Email:
mike@gmail.com
Password:
Login

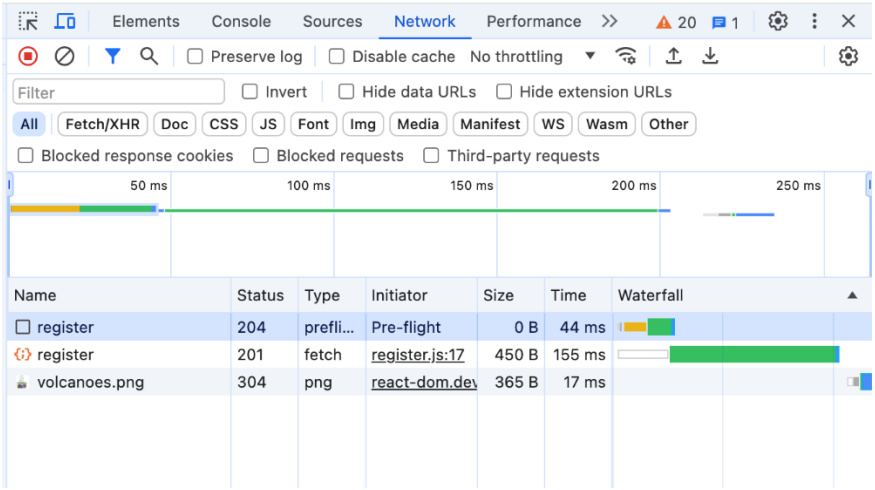

Elements
Console
Sources
Network
Performance
Preserve log
Disable cache
No throttling
Filter
Invert
Hide data URLs
Hide extension URLs
All
Fetch/XHR
Doc
CSS
JS
Font
Img
Media
Manifest
WS
Wasm
Other
Blocked response cookies
Blocked requests
Third-party requests
50 ms
100 ms
150 ms
200 ms
250 ms
Name
login
login
volcanoes.png
Headers
Payload
Preview
Response
Initiator
Timing
General
Request URL:
http://4.237.58.241:3000/user/login
Request Method:
POST
Status Code:
200 OK
Remote Address:
4.237.58.241:3000
Referrer Policy:
strict-origin-when-cross-origin

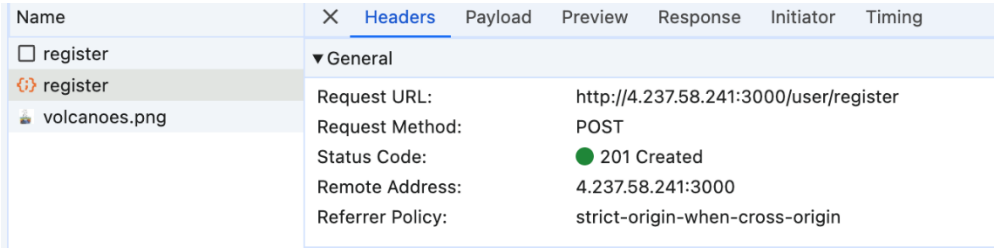

Name
login
login
volcanoes.png
Headers
Payload
Preview
Response
Initiator
Timing
Request payload
view source
{email: "mike@gmail.com", password: "password"}
email: "mike@gmail.com"
password: "password"

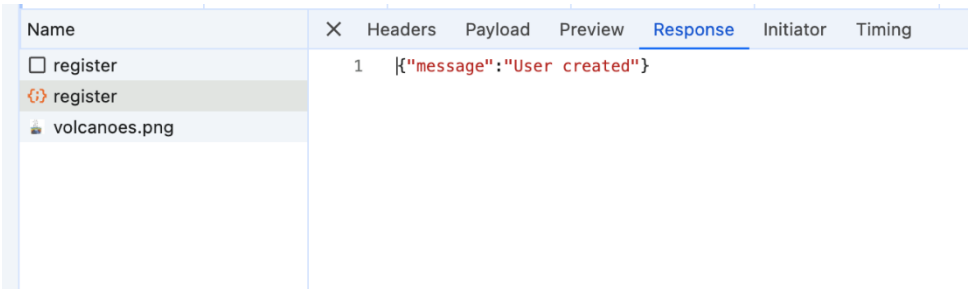
Name
login
login
volcanoes.png
Headers
Payload
Preview
Response
Initiator
Timing
{
"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlbWFpbCI6Im1pa2VAZ21haWwuY29tIiwiZXhwIjoxNzE1MTUyN
"token_type": "Bearer",
"expires_in": 86400
}

7

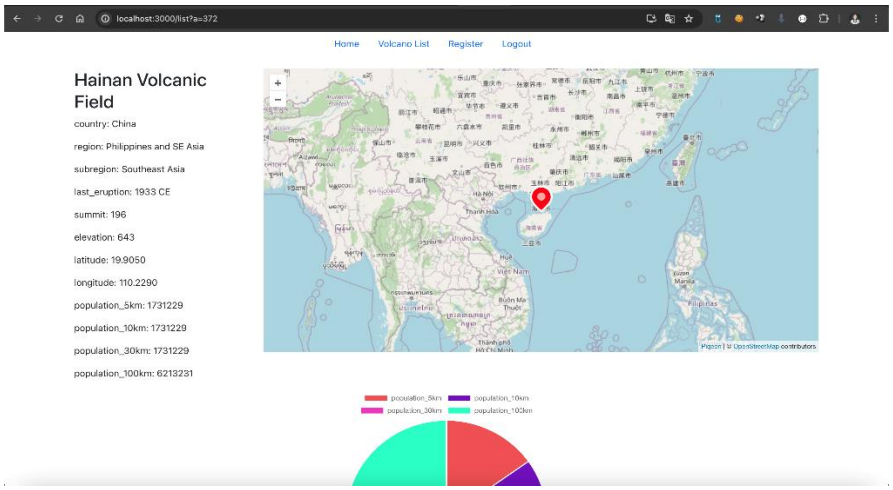

Register
Email:
root@qut.com
Password:
Login


Elements
Console
Sources
Network
Performance
Preserve log
Disable cache
No throttling
Filter
Invert
Hide data URLs
Hide extension URLs
All
Fetch/XHR
Doc
CSS
JS
Font
Img
Media
Manifest
WS
Wasm
Other
Blocked response cookies
Blocked requests
Third-party requests
50 ms
100 ms
150 ms
200 ms
250 ms
Name
Status
Type
Initiator
Size
Time
Waterfall
register 204 prefli... Pre-flight 0 B 44 ms
register 201 fetch register.js:17 450 B 155 ms
volcanoes.png 304 png react-dom.dev 365 B 17 ms


Name
register
register
volcanoes.png
Headers
Payload
Preview
Response
Initiator
Timing
General
Request URL: http://4.237.58.241:3000/user/register
Request Method: POST
Status Code: 201 Created
Remote Address: 4.237.58.241:3000
Referrer Policy: strict-origin-when-cross-origin


Name
register
register
volcanoes.png
Headers
Payload
Preview
Response
Initiator
Timing
1 {"message":"User created"}

*8*


localhost:3000/list?a=372
Home
Volcano List
Register
Logout
Hainan Volcanic Field
country: China
region: Philippines and SE Asia
subregion: Southeast Asia
last_eruption: 1933 CE
summit: 196
elevation: 643
latitude: 19.9050
longitude: 110.2290
population_5km: 1731229
population_10km: 1731229
population_30km: 1731229
population_100km: 6213231
population_5km
population_10km
population_30km
population_100km

*9*