UNIVERSITY PARTNER

UNIVERSITY OF WOLVERHAMPTON

BIRATNAGAR INTERNATIONAL COLLEGE

Internet Software Architecture (4CS017)

Report writing

<Individual task>

Weather App

Student ID : 2505471
Student name : Alisha Sigdel
Group : 4
Lecturer : Arvind Nepal
Module leader : Bishal Khadka
Tutor : Arvind Nepal
Word Count: 1660
Submitted on : 12 Feb 2025

Acknowledgement

I would also use this chance to express my sincere appreciation to all teachers for their great guidance, encouragement, and constructive criticism in preparing this report. Their encouragement and support helped my knowledge to progress and to make this work better in terms of quality.

I also appreciate Biratnagar International College for providing required resources and a learning-friendly learning environment that facilitated completion of this report.
I also appreciate my relatives, my friends, and my colleagues for their encouragement and support throughout this process.

Their encouragement was a great source of strength throughout this process. I also appreciate all authors, researchers, and sources used in composing the body of this report.

Table of content

Chapter 1: Reflecting on Personal and Academic Growth During My First Semester

This semester I have learned many new things that help me grow, both personally and academically. I have no knowledge at first about web development and programming, and this time I can start implementing more programming like HTML, CSS, JavaScript, and Python. For me, a rather big achievement is that I can design and build websites.

UML designing is one of the most important skills I learned, as it gave me a fundamental idea about planning and structuring the software before building it. On the same note, I also learnt the concept of web hosting which enabled me to make my projects to be visible to anybody and anybody who would want to use it online. I am pretty sure these skills can help me in my future career in technology as well as business.

I also looked at being able to make circuits, so I was able to learn the basic function of electronic components. This was a momentous one, since the two were brought together: software with hardware, and I gained an appreciation of both areas.

And personally speaking, I managed to better myself in terms of time management and problem solving. I learned to do my tasks well with the help of completing different projects and assignments and solving challenges when I encountered them. Along with that, I was becoming increasingly comfortable learning new things even if they were hard to begin with.

I think that this semester has been an excellent journey learning and improving oneself. I feel much more ready for any future challenges that may occur in the technological and Web development field.

Chapter 2: Summary on Prototypes

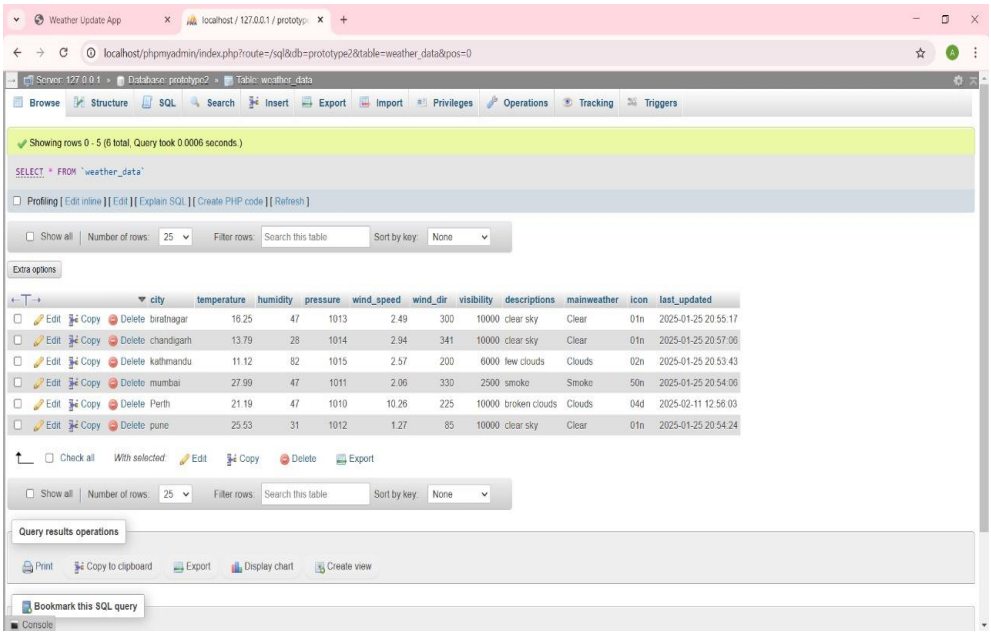
2.1 Prototype I

This is a clean and modern looking webpage which makes its content easy to read due to its transparent card layout. Aesthetic beauty has been provided by the background image like a forest with sunrays. In search bar, users can input a city name and click the "Get Weather" button to get weather details.
So, regarding weather, it gives essential information, such as:

City Name and Date – The selected location (Perth, AU) and current date.
Specific conditions: Clouds (main weather description), Overcast clouds.
Temperature – Displayed in Celsius (23°C).
Humidity (48%), Wind speed (10.47 m/s), Pressure (1010 hPa), and Visibility (10.0 km), Additional details

The visually pleasing design is proven to be functional towards weather updates; it comes as a prototype. A simple but effective way to show weather information is with HTML, CSS, and using API integration.

2.2 Prototype 2

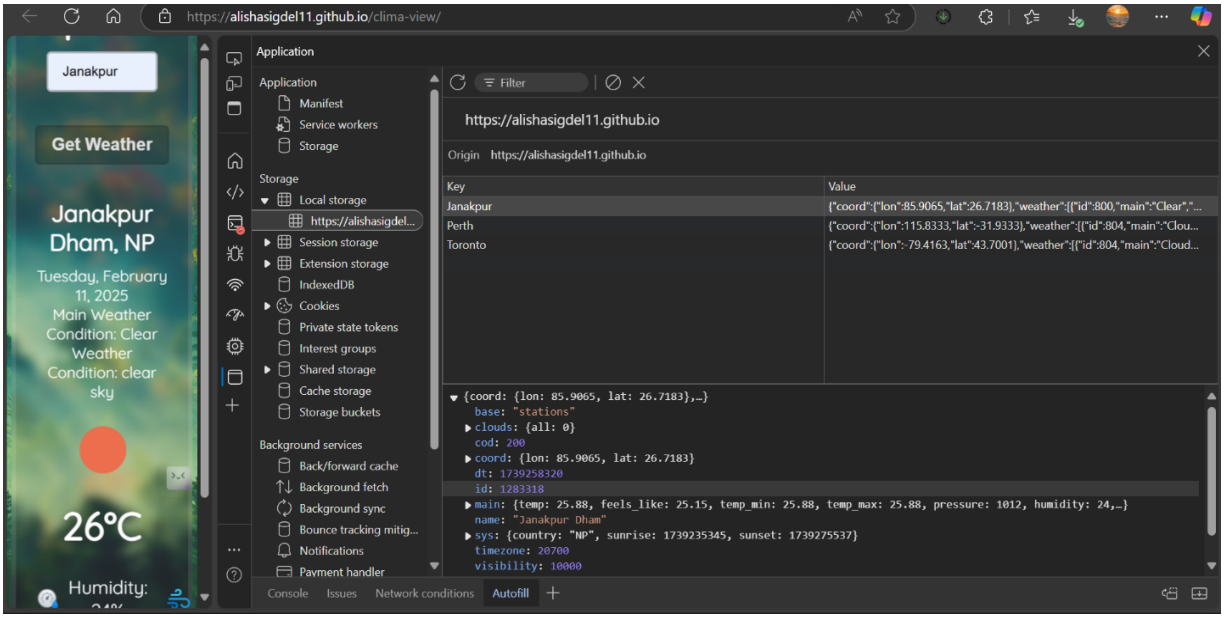


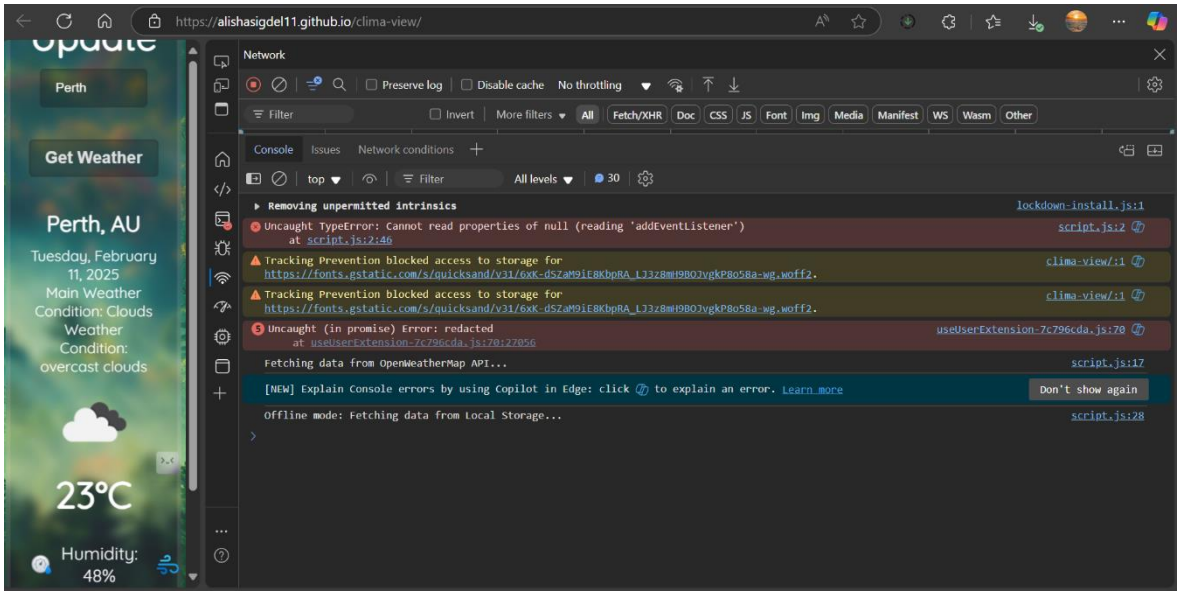
As shown in Prototype 2, PHP and MySQL are integrated to store and retrieve data of the weather webpage. This prototype is different from the previous one because it does not use real time calls to an API but, instead, stores the weather data in a MySQL database using PHP scripts. It enables users to get previously searched cities' weather data without needing to make a new API request, becoming more efficient.

Before fetching new information from the Weather API, the system first checks the database for existing data to reduce unnecessary API calls. The stored data is updated automatically if it is outdated. In this way users will have reliable and latest weather details. In an organized format website shows temperature, humidity, wind speed, other weather conditions, and all this page shows dynamic values.

Another benefit of this prototype includes persistent data storage, optimized performance and reduced API usage. Taken as a whole, Prototype 2 speeds up, scales, and makes the weather page more efficient, scalable, and reliable.

2.3 Prototype 3





Prototype 3 not only integrates PHP and MySQL for data storage but local storage as well, after Prototype 1, that was only HTML, CSS, and an API and Prototype 2. Also, this version is available on GitHub, which makes it more readily accessible.

Like the previous prototypes, the design continues to stay the same with modern, transparent layout and a visually appealing background. In the search bar, a city name is entered by the user, and it is clicked to "Get Weather" to fetch the weather details of that city.

Online Mode

The weather data is fetched in real time from the Weather API and when the user is online only.

Store the fetched data in local storage to make use in future cases.

Offline Mode

Local storage is used to keep the webpage state to display weather data of cities which were previously searched by the user if the user goes offline.

Nevertheless, cities searched before getting offline will not have any saved data and subsequently won’t be accessible.

Advantages of Local Storage

Water Resiliency – Users can still use weather data for cities they did not find internet access in after a natural disaster.

Less Code – There is minimal code, and most of it is attributes.

Saving bandwidth and increasing efficiency, it will store data that has been previously fetched.

Hosting on GitHub

The advantage with this prototype is that it is hosted on GitHub, with many benefits involved as well.

Ease of usage – The users can easily do whatever they want on the weather webpage from any device with an internet connection.

Version Control – Allows easy tracking of changes and improvements.

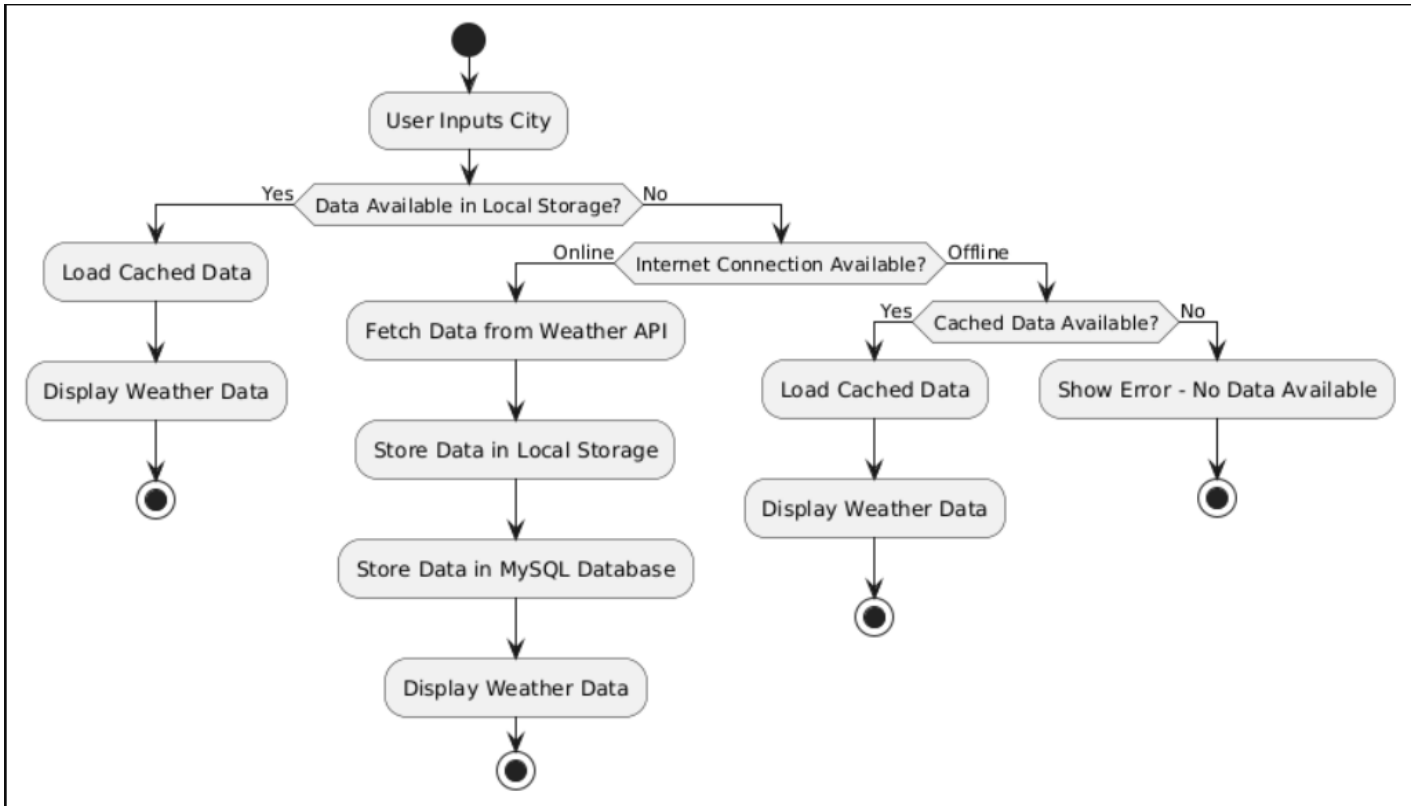
GitHub provides free Hosting – A cost effective solution for deploying web projects.

Contributing – Developers can easily contribute and update the project.

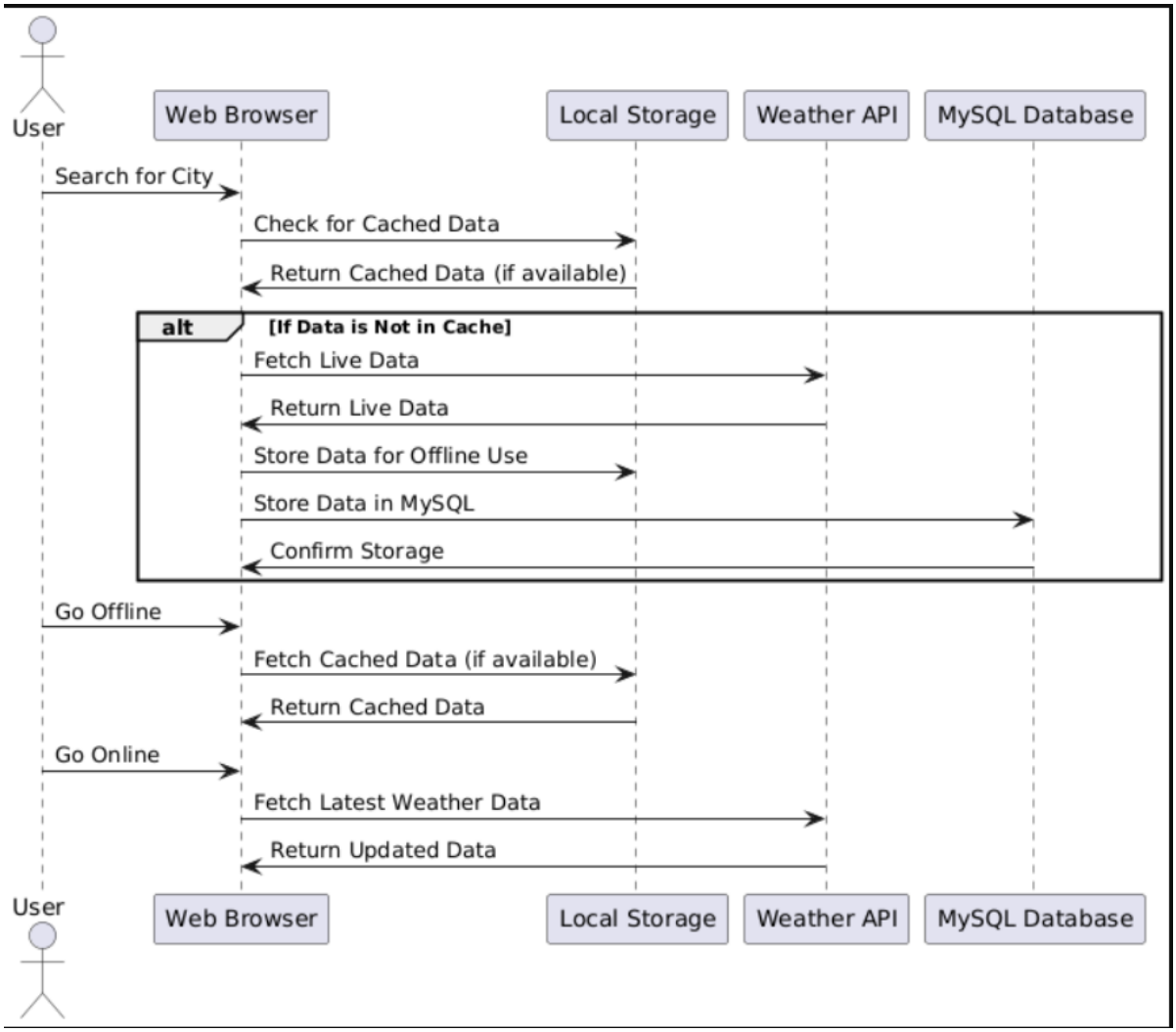
In Prototype 3 we built on the weather webpage using local storage; prototype it on GitHub for wider reach; patch the autocomplete through Google; and broke it up into component, so it is easier to manage and select by its classes. The webpage is made dynamic, efficient and user friendly by combining HTML, CSS, PHP, Java, MySQL, local storage and API integration.

Chapter 3: UML diagrams

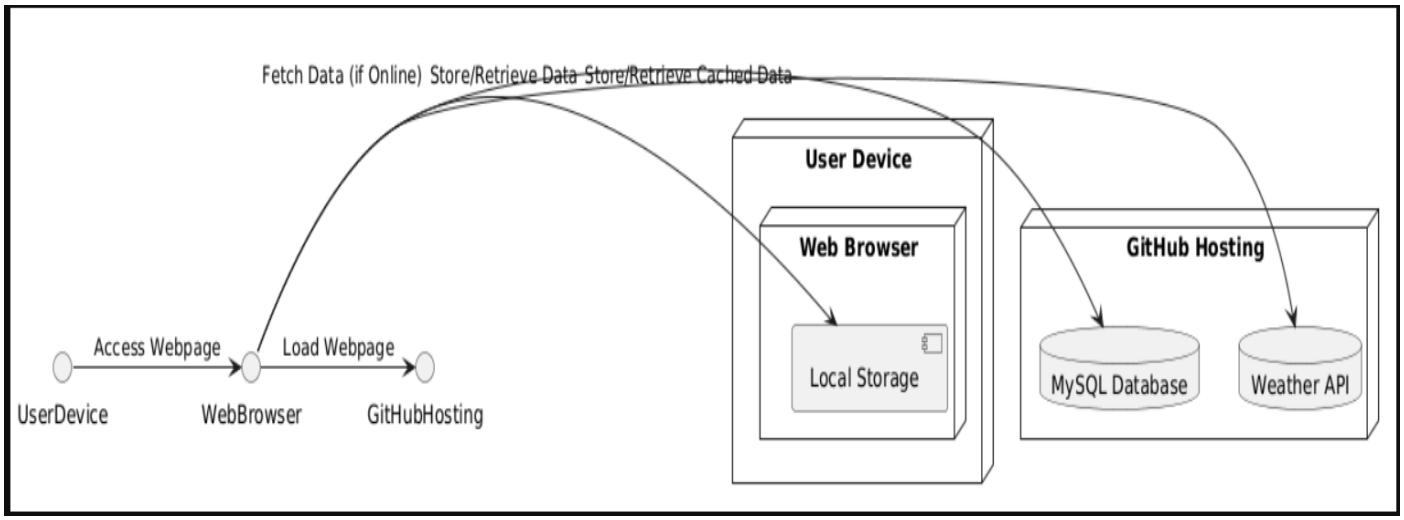
3.1 Activity diagram



3.2 Sequence diagram



3.3 Deployment diagram



Chapter 4: Summary of Web hosting

The weather application was hosted on GitHub Pages, a free and consistent web hosting service provided by GitHub. How the application was hosted is explained below:

1. Choosing GitHub Pages for Hosting

Hosting static website with GitHub Pages is great because it just works with HTML, CSS, JavaScript without need server for hosting. Since our project is basically a front-end-based weather application that fetches data from a weather API, GitHub pages was the perfect choice.

2. Steps to Deploy the Application

Created a Git Hub Repository – uploaded more than project files (HTML, CSS, JavaScript) to Git Hub.
We enabled GitHub pages and selected main branch as the source in the repository settings.
Live URL was generated – You received a public URL where the webpage is being hosted and being accessible by anyone.

3. Advantages of Using GitHub Pages

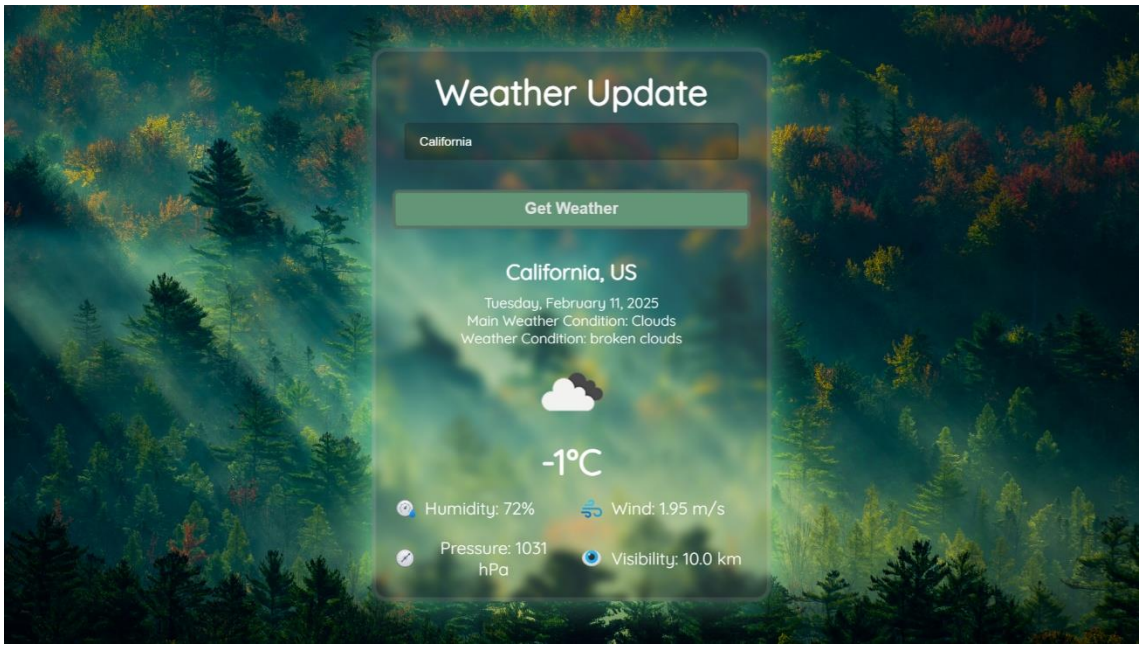
No Cost or Set Up – It is Free & Easy to use.
Automatic Update – Whenever any changes in the GitHub repository are made, live website is updated automatically.
GitHub Pages is fast & secure – As the name implies, our hosting environment is fast and very secure.

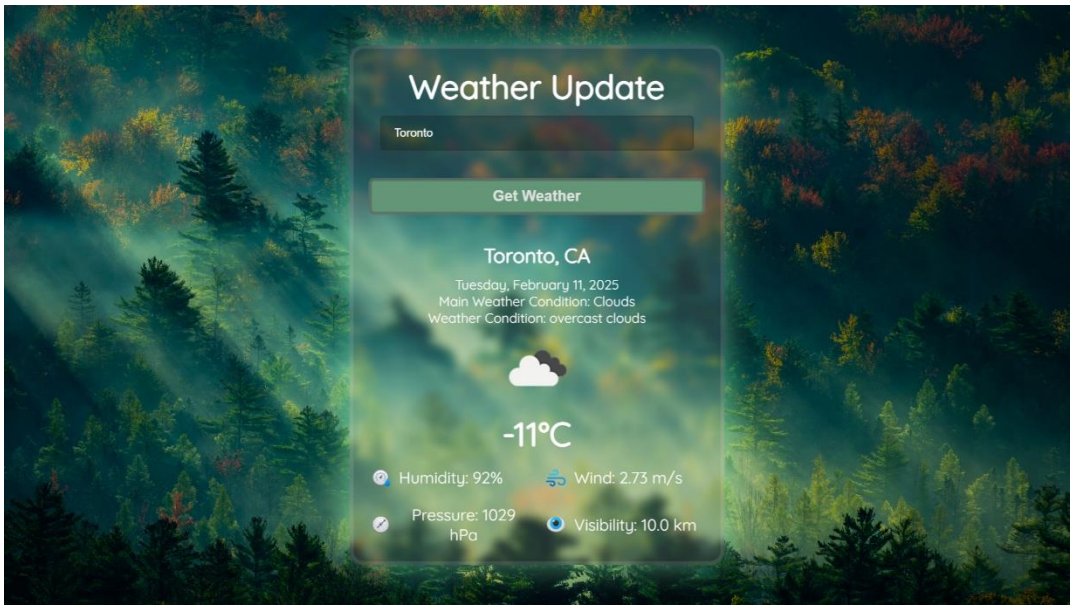
4. Conclusion

As we do not need the backend to share the project, we have been able to host the weather application on GitHub Pages. In addition, this experience gave me hands on learning in web development and hosting.

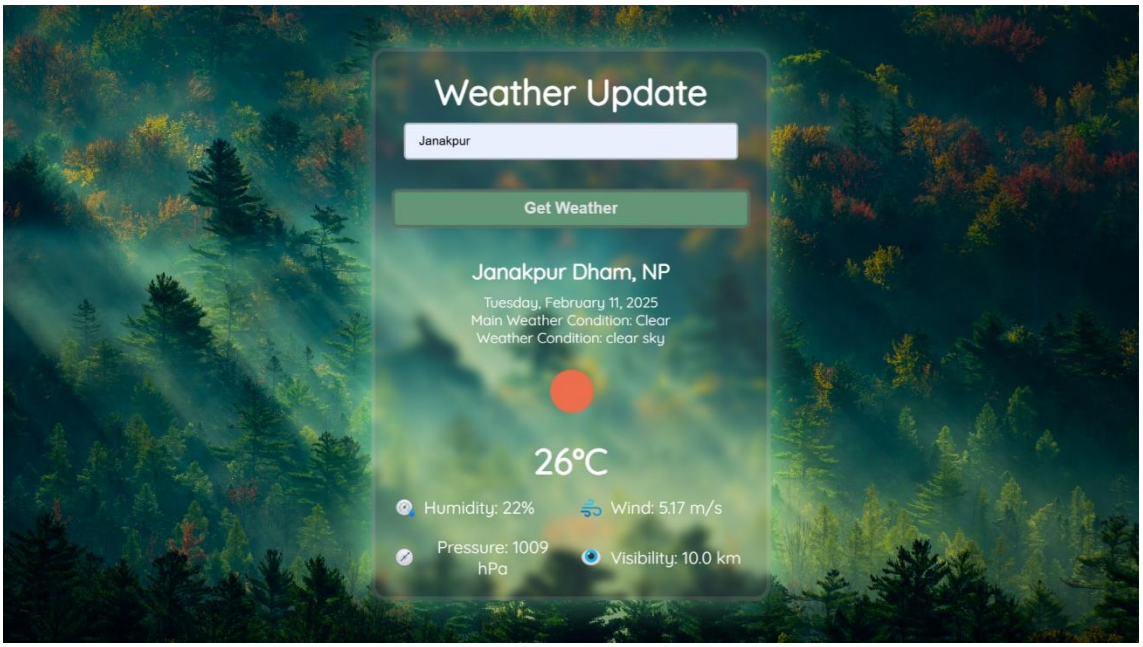
Chapter 5: Screenshots of hosted applications

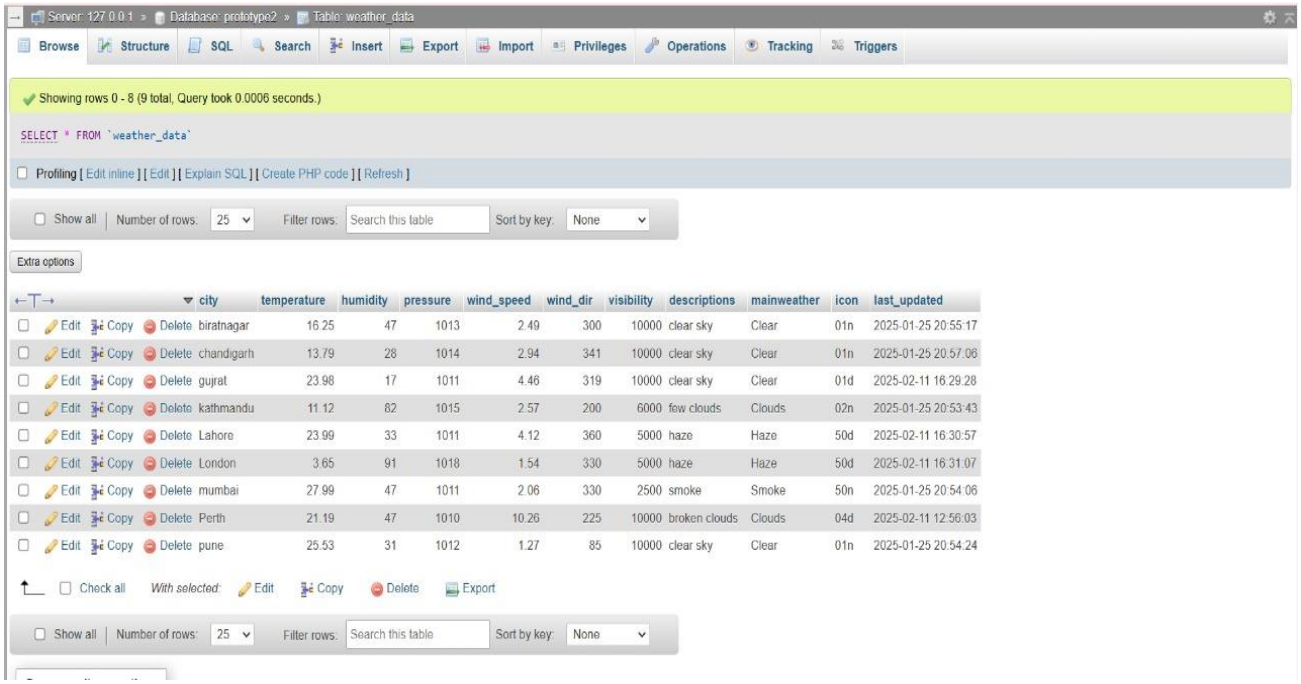
5.1 Screenshots of prototype 1



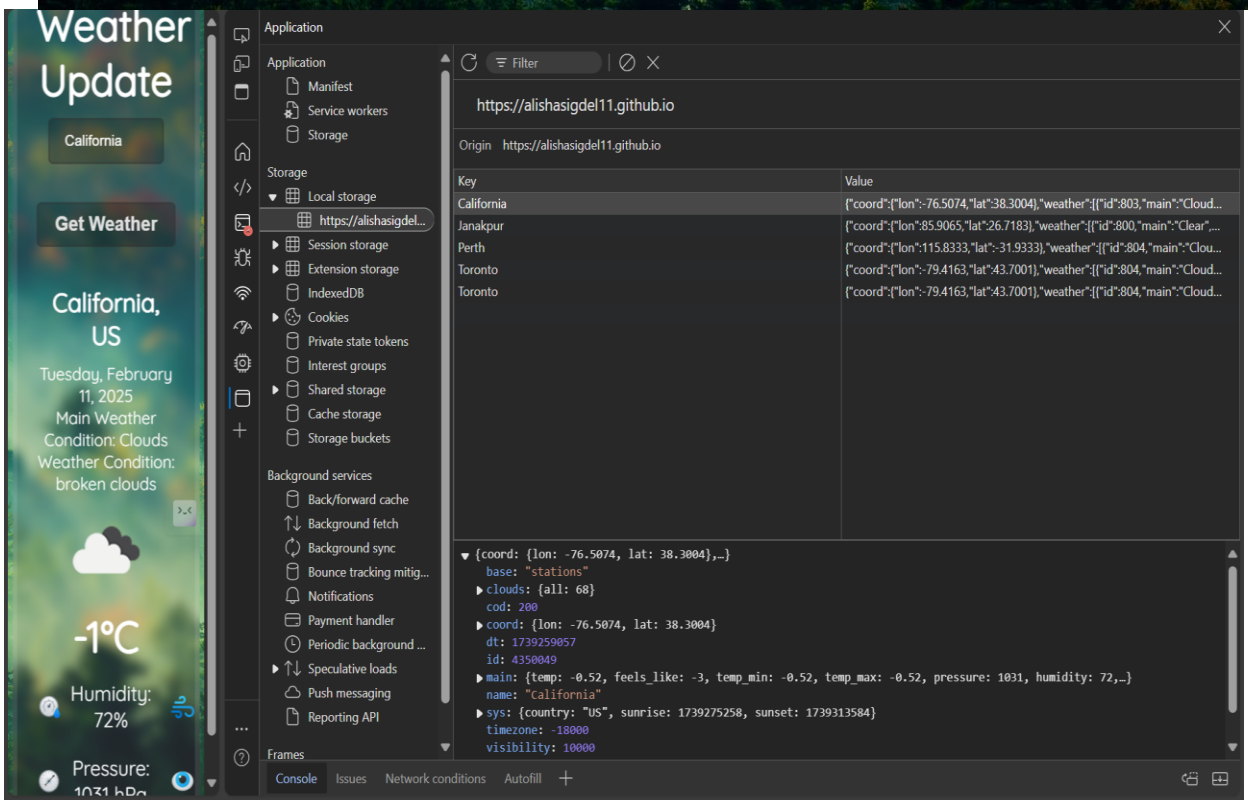


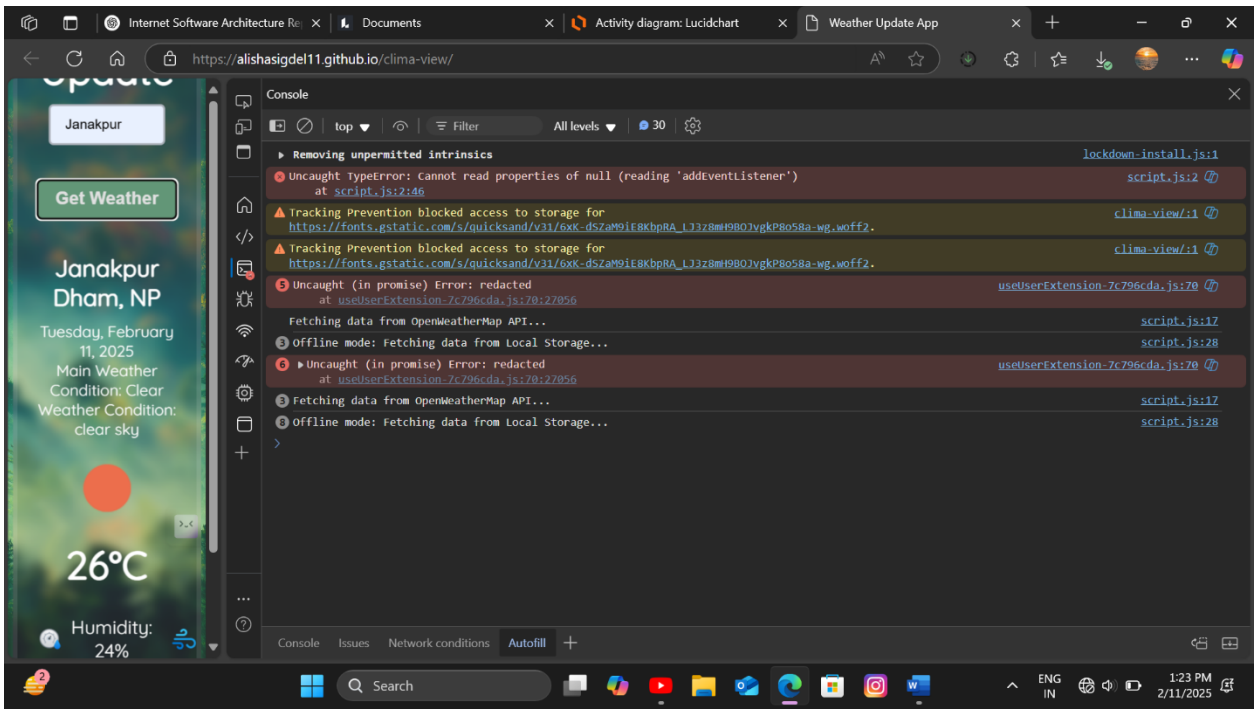
5.2 Screenshots of prototype 2

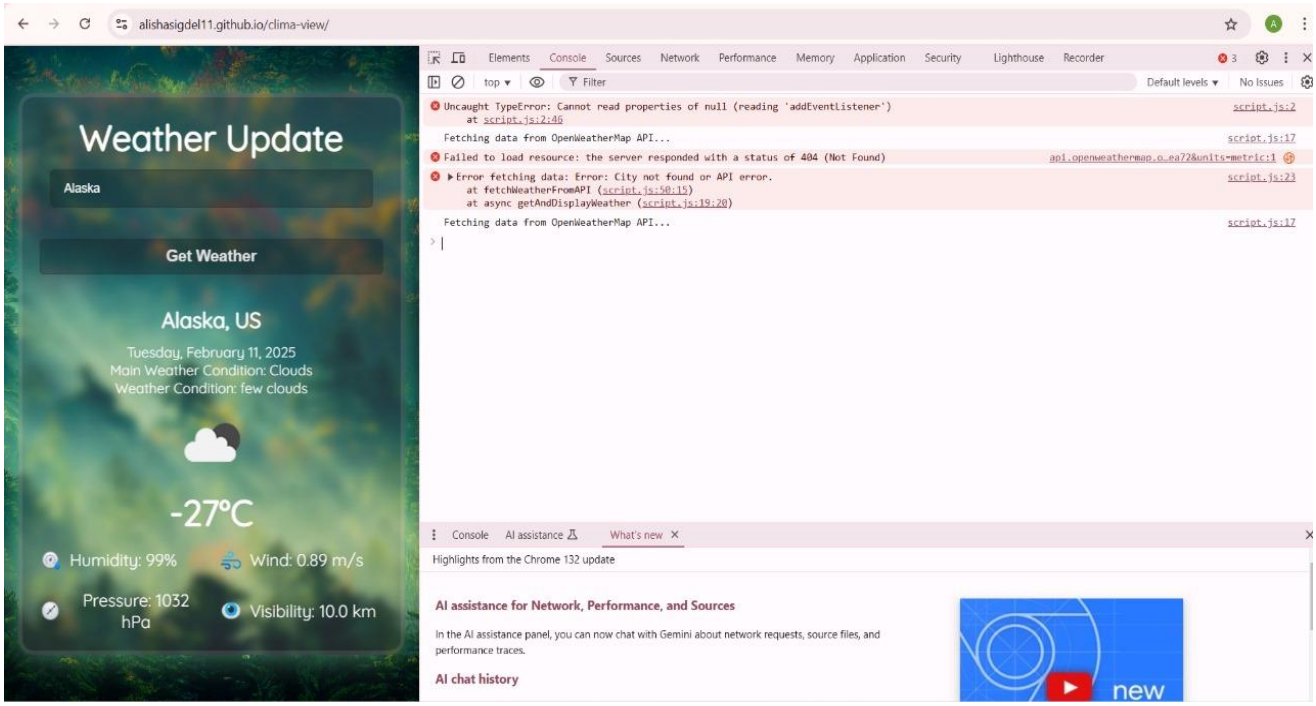


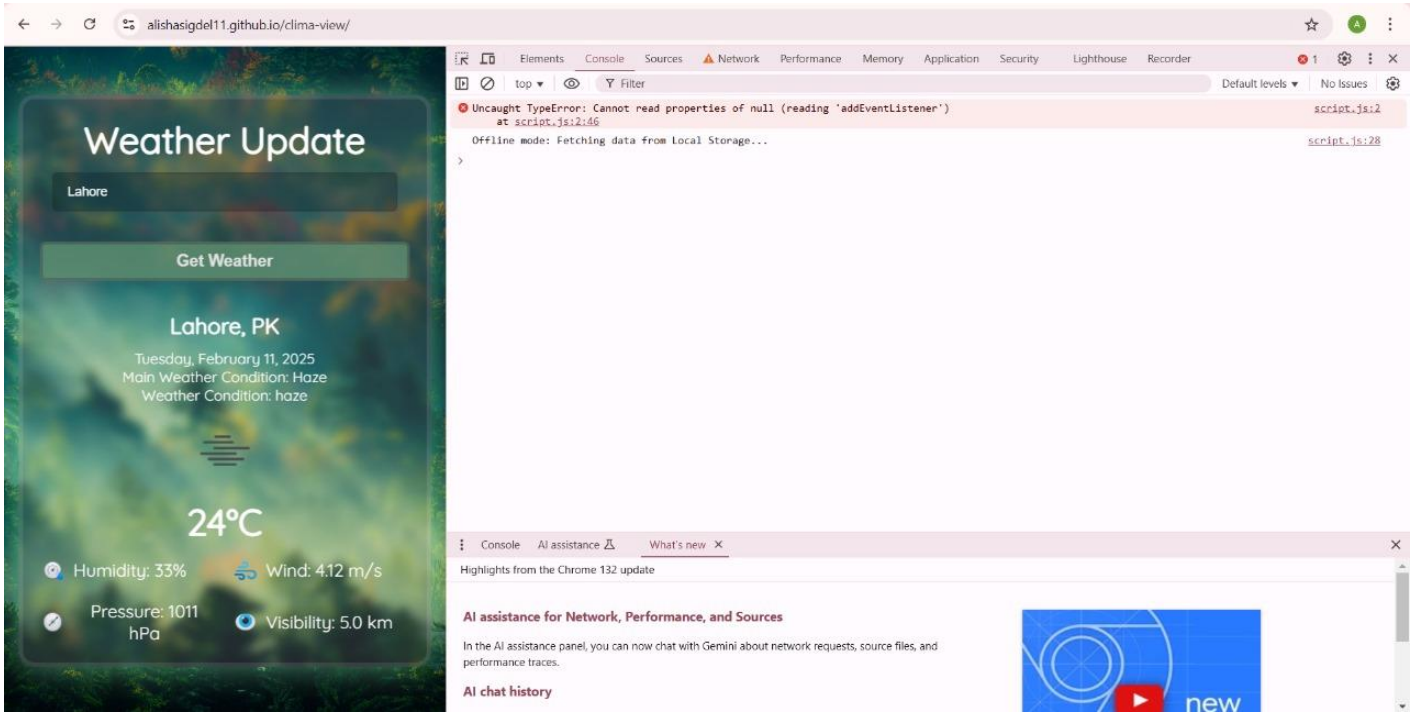


5.3 Screenshots of prototype 3




Internet Software Architecture Re
Documents
Activity diagram: Lucidchart
Weather Update App
https://alishasigdel11.github.io/clima-view/
Janakpur
Get Weather
Janakpur Dham, NP
Tuesday, February 11, 2025
Main Weather Condition: Clear
Weather Condition: clear sky
26°C
Humidity: 24%
Console
top
Filter
All levels
30
Removing unpermitted intrinsics
lockdown-install.js:1
Uncaught TypeError: Cannot read properties of null (reading 'addEventListener')
at script.js:2:46
script.js:2
Tracking Prevention blocked access to storage for https://fonts.gstatic.com/s/quicksand/v31/6xK-dSZaM9iE8KbpRA_LJ3z8mH9BOJvgkP8o58a-wg.woff2.
clima-view/:1
Uncaught (in promise) Error: redacted
at useUserExtension-7c796cda.js:70:27056
useUserExtension-7c796cda.js:70
Fetching data from OpenWeatherMap API...
script.js:17
Offline mode: Fetching data from Local Storage...
script.js:28
Console
Issues
Network conditions
Autofill
Search
ENG IN
1:23 PM
2/11/2025


alishasigdel11.github.io/clima-view/
Weather Update
Alaska
Get Weather
Alaska, US
Tuesday, February 11, 2025
Main Weather Condition: Clouds
Weather Condition: few clouds
-27°C
Humidity: 99%
Wind: 0.89 m/s
Pressure: 1032 hPa
Visibility: 10.0 km
Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder
Uncaught TypeError: Cannot read properties of null (reading 'addEventListener')
at script.js:2:46
Fetching data from OpenWeatherMap API...
Failed to load resource: the server responded with a status of 404 (Not Found)
Error fetching data: Error: City not found or API error.
at fetchWeatherFromAPI (script.js:50:15)
at async getAndDisplayWeather (script.js:19:20)
Fetching data from OpenWeatherMap API...
Console AI assistance What's new
Highlights from the Chrome 132 update
AI assistance for Network, Performance, and Sources
In the AI assistance panel, you can now chat with Gemini about network requests, source files, and performance traces.
AI chat history
new


alishasigdel11.github.io/clima-view/
Weather Update
Lahore
Get Weather
Lahore, PK
Tuesday, February 11, 2025
Main Weather Condition: Haze
Weather Condition: haze
24°C
Humidity: 33%
Wind: 4.12 m/s
Pressure: 1011 hPa
Visibility: 5.0 km
Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder
Uncaught TypeError: Cannot read properties of null (reading 'addEventListener')
at script.js:2:46
Offline mode: Fetching data from Local Storage...
Console AI assistance What's new
Highlights from the Chrome 132 update
AI assistance for Network, Performance, and Sources
In the AI assistance panel, you can now chat with Gemini about network requests, source files, and performance traces.
AI chat history
new

Chapter 6: Learning Outcomes and Conclusion

During this project, I learned and were able to use the skills regarding web development, API integration, and UML diagram design.

Learning Outcomes:

Skill Learned: Created a structured webpage using HTML, CSS and JavaScript.

Resource Management: When working with DOM elements, learned to manage resources by not unnecessarily changing the whole document and re-rendering it with new data.

As a result of my work on UML Diagrams, I was able to create Activity, Sequence, and Deployment diagrams to visually express how the system operates and how its architecture looks.

Deploying a website somewhere and making it accessible for the internet: first investigating the different ways to host a website.

Challenges: Solving problems related to API requests, handling of data, as well as UI design.

Conclusion:

This project has helped me improve my technical and problem-solving skills; hence I can utilize theoretical knowledge on problem solving in theory and practice. It has been extremely useful to experience working with real world data, design interactive web pages, and learn about system workflow. This project enabled me to build real-life functional and user-friendly web applications.

Chapter 7: Domain Link

https://alishasigdel11.github.io/clima-view/