



# IT314 - Software Engineering

Non-functional Testing

Team Members:

202201320 SUMIT KUMAR VISHWAKARMA  
202201285 JIVANI MADHAV BHAVESH  
202201366 BHAVYA SHAH  
202201333 KAKADIYA SHRAVANKUMAR PARESHBHAI  
202201323 SHRESTHA BHAUMIKKUMAR THAKKAR  
202201328 VRAJ KHOKHARIYA  
202201350 KANAKAD KRUSHANGKUMAR RAMESHBHAI  
202201346 MODI NISHARG VIPULKUMAR  
202201266 RATHVA KARNIKKUMAR VINODBHAI  
202201277 SHARVIL TEJAS OZA

# Non Functional Testing

## Load Testing:

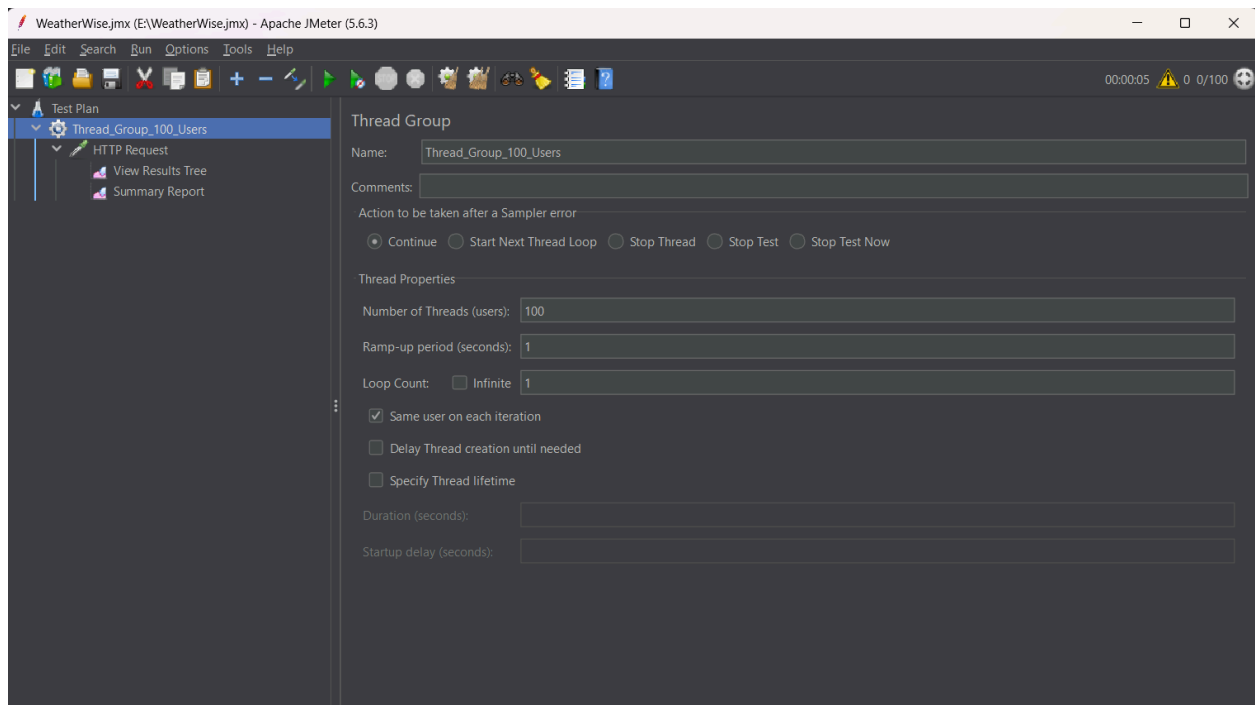
We conducted load testing under normal conditions to evaluate how many users our website can handle simultaneously. For this, we used HTTP GET method to simulate user activity. Attached are screenshots from Apache JMeter, showcasing the test results.

**Software Used :** [Apache Jmeter](#)

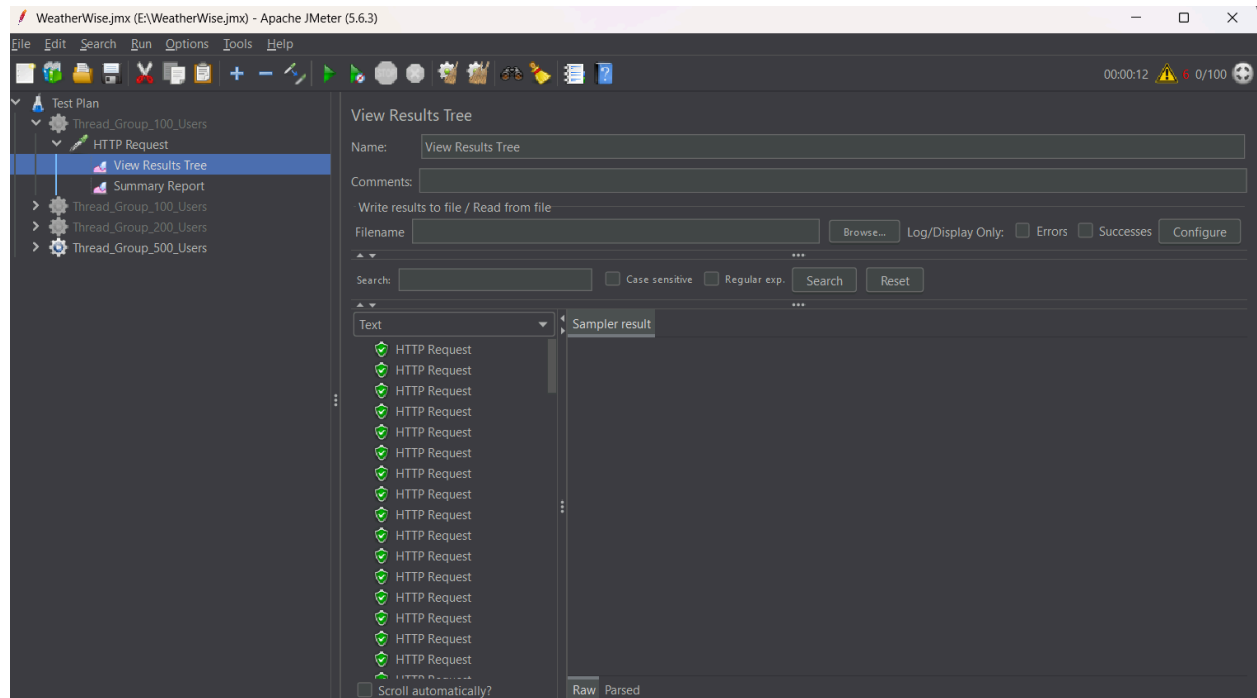
Testing Home Module GET request to PATH '[weatherwise.pythonanywhere.com](http://weatherwise.pythonanywhere.com)'

## Test 1: For 100 users

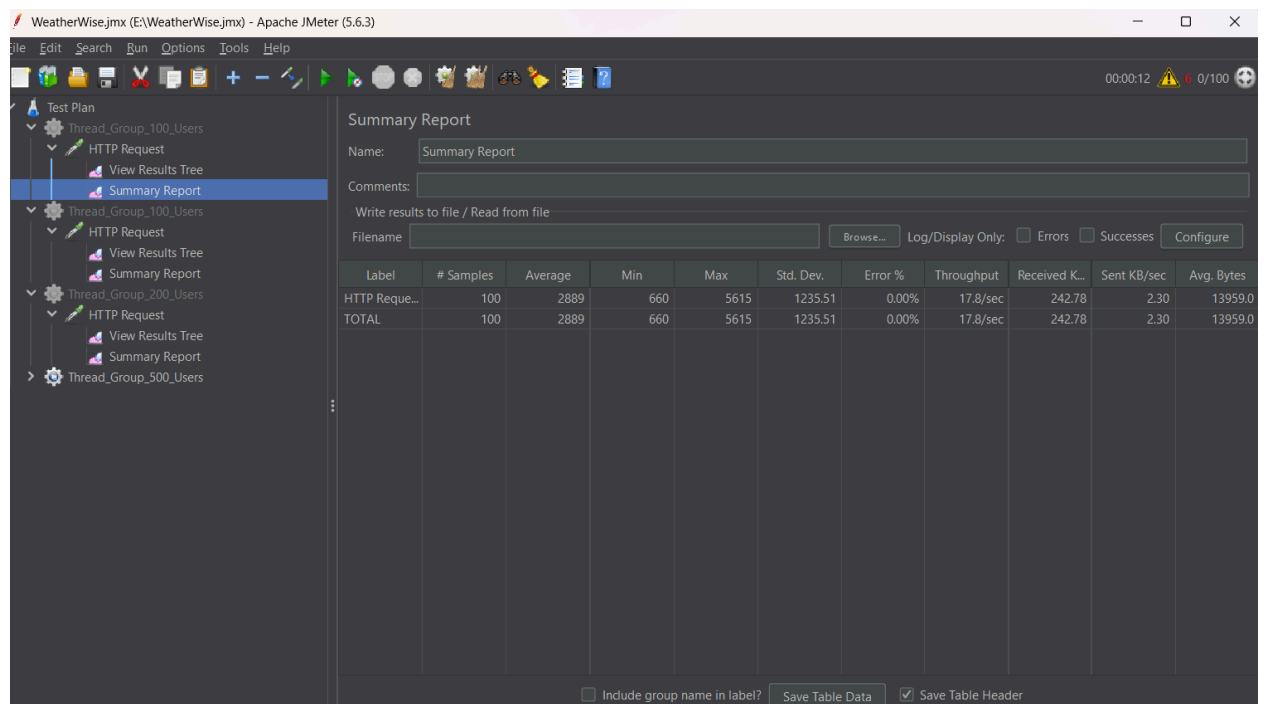
Scenario: We have accessed the home page of router with the same user from multiple hosts (100 and 500) and will be measuring the load balance and throughput of the system



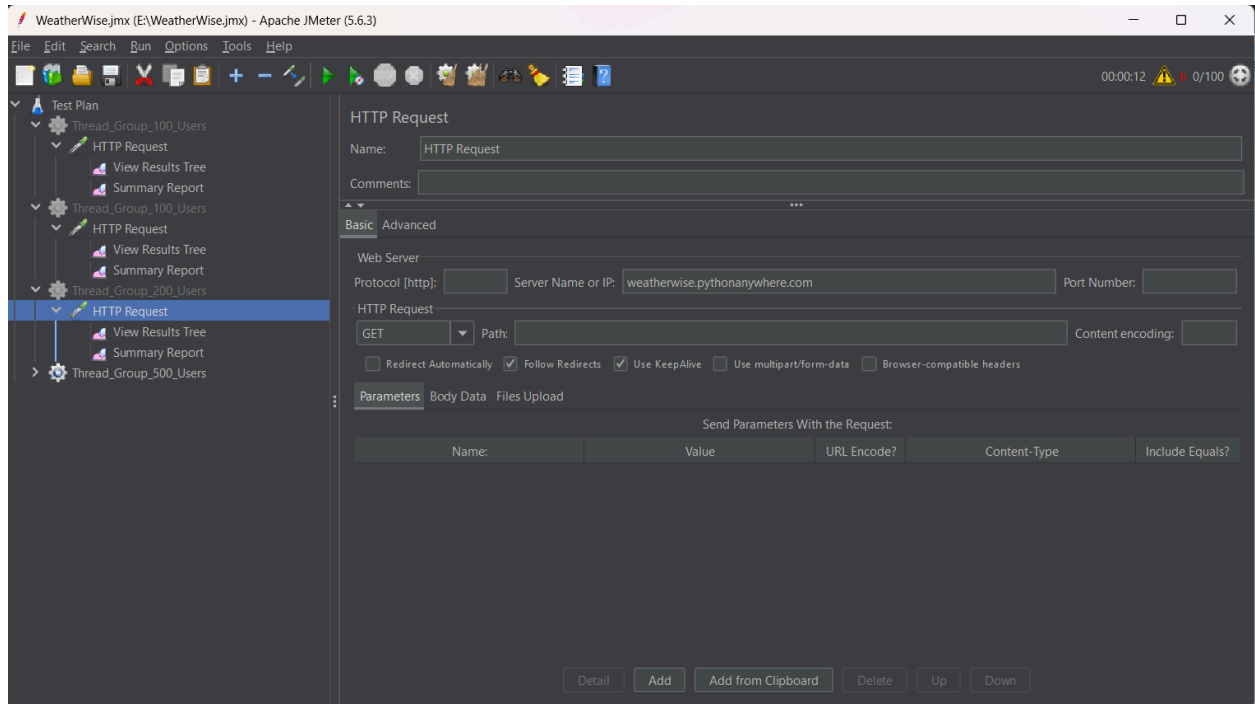
Result (Result Tree):



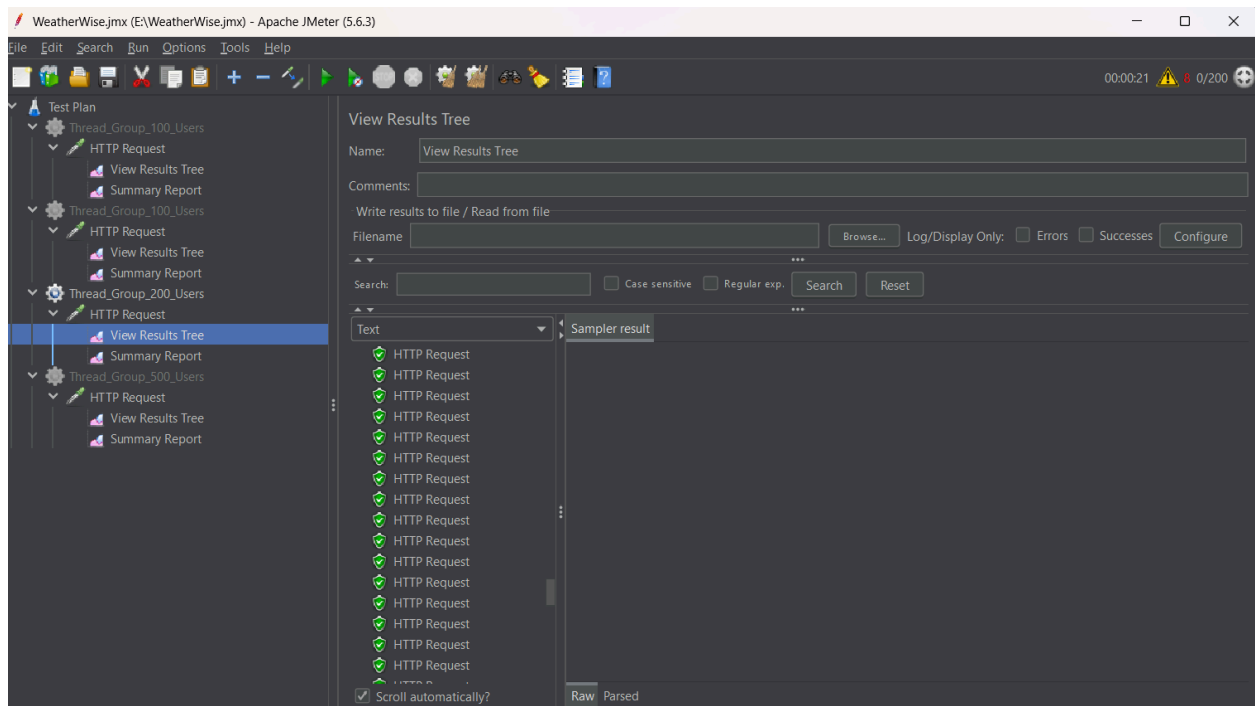
### Result (Summary Report):



## Test 2: For 200 users



## Result (Result Tree):



## Result (Summary Report):

WeatherWise.jmx (E:\WeatherWise.jmx) - Apache JMeter (5.6.3)

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received K...	Sent KB/sec	Avg. Bytes
HTTP Reque...	200	6664	437	21042	7712.26	40.50%	9.1/sec	81.93	0.92	9238.0
TOTAL	200	6664	437	21042	7712.26	40.50%	9.1/sec	81.93	0.92	9238.0

☐ Include group name in label? Save Table Data ☒ Save Table Header

## Test 3: For 500 users

WeatherWise.jmx (E:\WeatherWise.jmx) - Apache JMeter (5.6.3)

HTTP Request

Name: HTTP Request

Comments:

Basic Advanced

Web Server

Protocol (http): Server Name or IP: weatherwise.pythonanywhere.com Port Number:

HTTP Request

GET Path: Content encoding:

☐ Redirect Automatically ☒ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data ☐ Browser-compatible headers

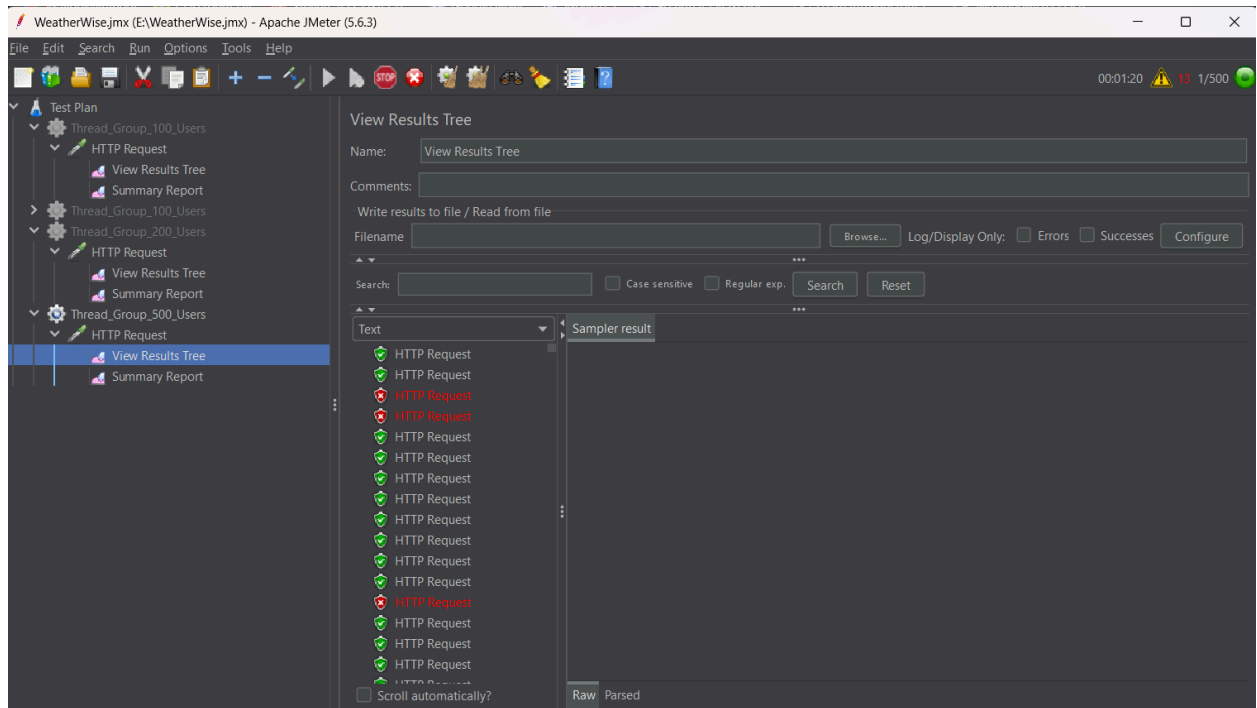
Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?
-------	-------	-------------	--------------	-----------------

Detail Add Add from Clipboard Delete Up Down

Result (Result Tree):



### Result (Summary Report):

WeatherWise.jmx (E:\WeatherWise.jmx) - Apache JMeter (5.6.3)

File Edit Search Run Options Tools Help

00:01:49 13 0/500

Test Plan

- Thread\_Group\_100\_Users
  - HTTP Request
    - View Results Tree
    - Summary Report
- Thread\_Group\_200\_Users
  - HTTP Request
    - View Results Tree
    - Summary Report
- Thread\_Group\_500\_Users
  - HTTP Request
    - View Results Tree
    - Summary Report

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename:  Browse... Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received K...	Sent KB/sec	Avg. Bytes
HTTP Reque...	500	18192	769	109130	7363.24	79.60%	4.6/sec	22.13	0.12	4952.6
TOTAL	500	18192	769	109130	7363.24	79.60%	4.6/sec	22.13	0.12	4952.6

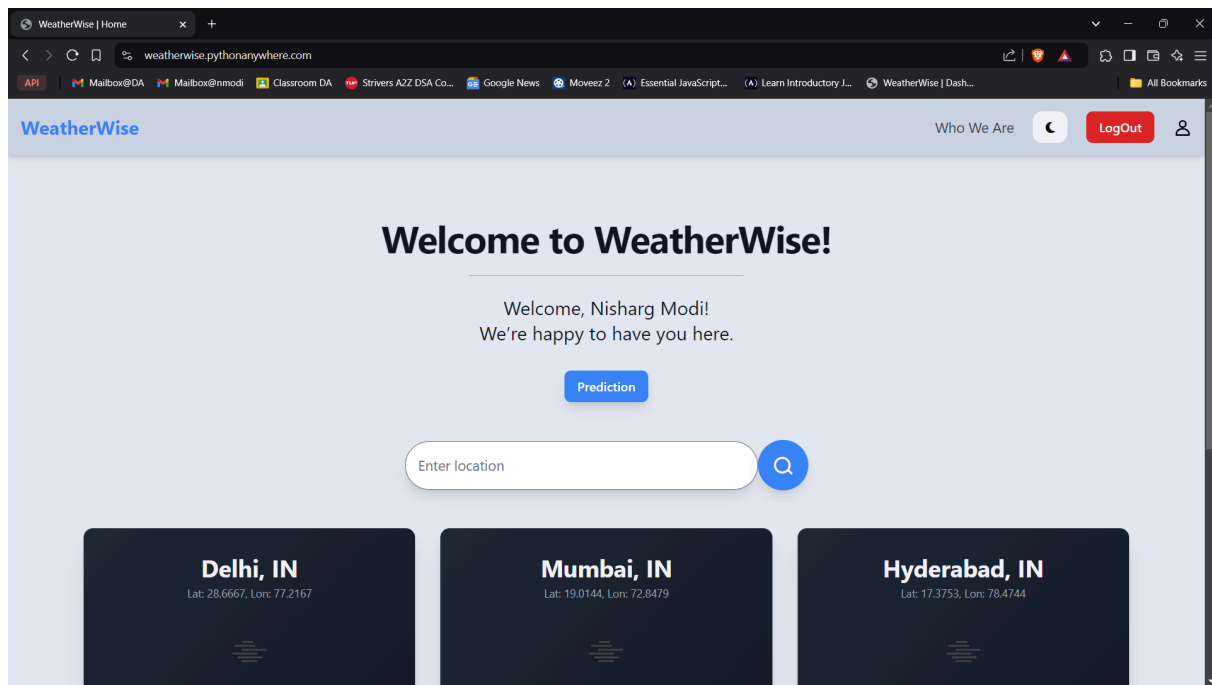
☐ Include group name in label?  ☒ Save Table Header

Moreover, we are using [OpenWeather API](#) which gives 1000 calls per day, and another API we are using is [WeatherStack API](#) which gives 100 calls per month on a free plan. Which also restricts our maximum user load.

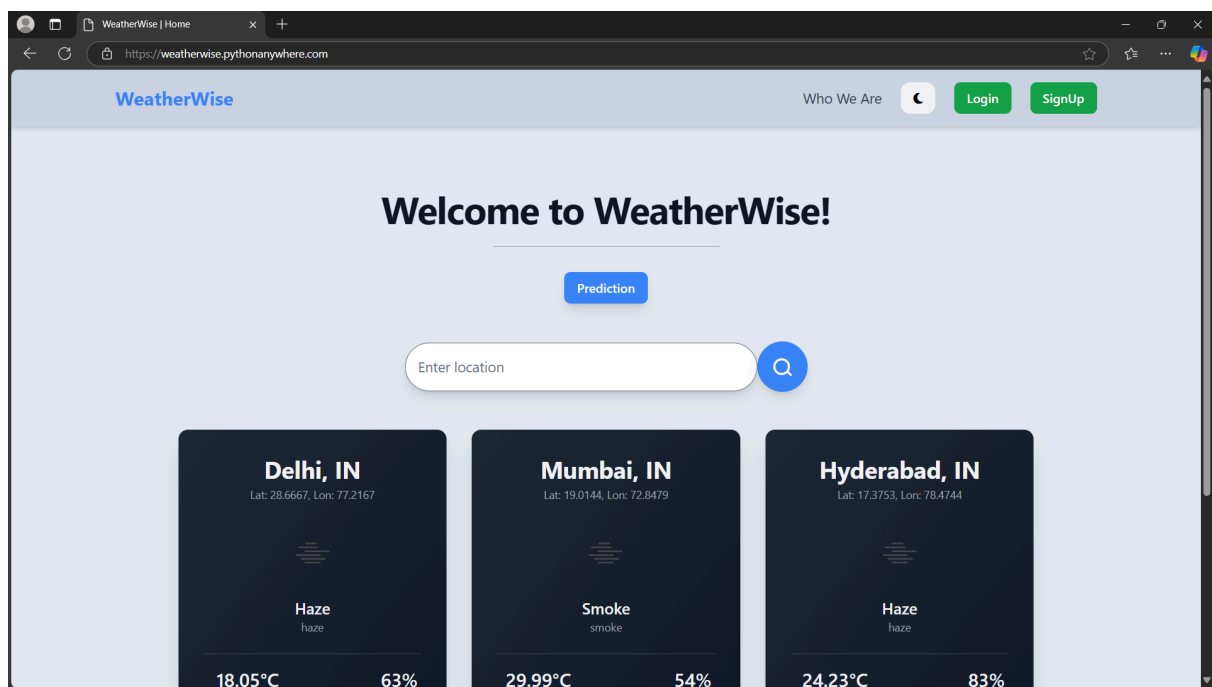
# Compatibility testing

We've made sure that the website is compatible with the latest web browsers covering Edge, Chrome and Brave and also for all devices like mobile and laptops.

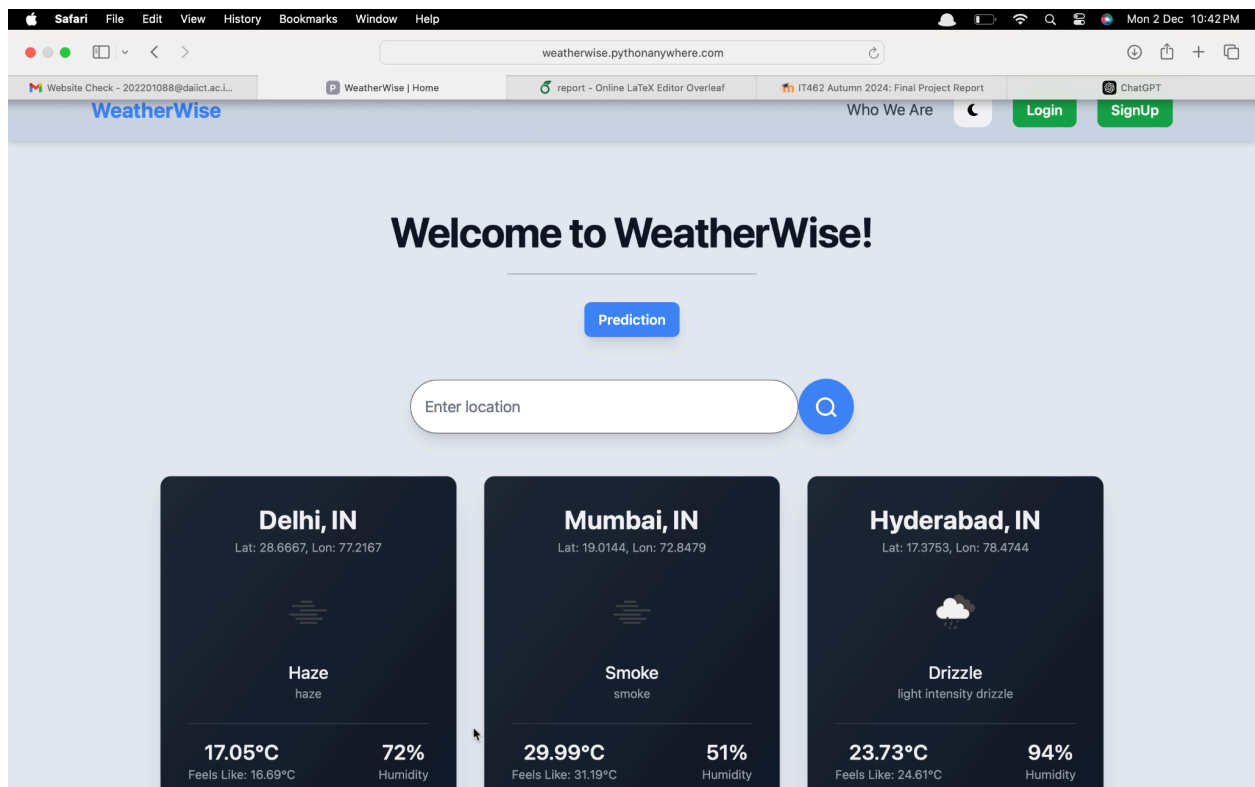
Brave:



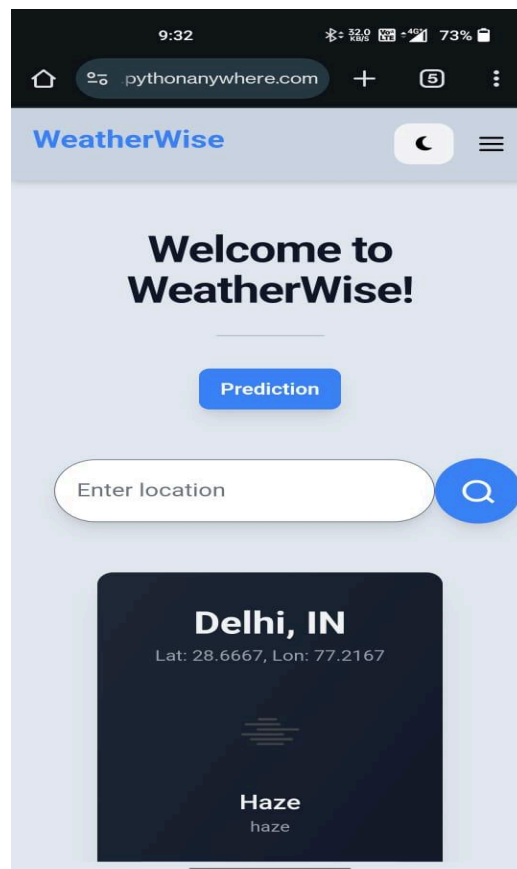
Edge:



Safari:



Chrome(Android):





# Security Testing:

During the performance testing of the website, it was observed that the server response included a cookie with an insecure configuration. Specifically, the Set-Cookie header for the csrftoken cookie did not include the HttpOnly flag. This flag is crucial for enhancing cookie security, as it restricts access to cookies from client-side scripts, reducing the risk of attacks such as cross-site scripting (XSS). This issue compromised our website's security.

M

Insecure cookie setting: missing HttpOnly flag

Confirmed

Evidence

URL

https://weatherwise.pythonanywhere.com/

Cookie Name  
csrftoken

The server responded with Set-Cookie header(s) that does not specify the HttpOnly flag:

Set-Cookie: csrftoken=jM2KCC2q8TKm1dB1nsSEAQFoNJ5Cfvqu

This issue can be fixed by enabling the HttpOnly flag for cookies in your application settings or backend configuration.

We have scanned our website with a third party “<https://pentest-tools.com/>” Below is the report for the same.

