Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Student

Name: Mahitha Kalaga

Email: kalagam1@udayton.edu

Repository Information

Repository's URL: https://github.com/MahithaKalaga-cyber/waph-mahitha.git

This is a private repository for Mahitha Kalaga to store all the code from the course. The organization of this repository is as follows.

Labs

Hands-on exercises in Lectures

- Lab 0: Development Environment Setup
- Lab 1: Foundations of the Web
- Lab 2: Front-end Web Development

Report

The lab's overview

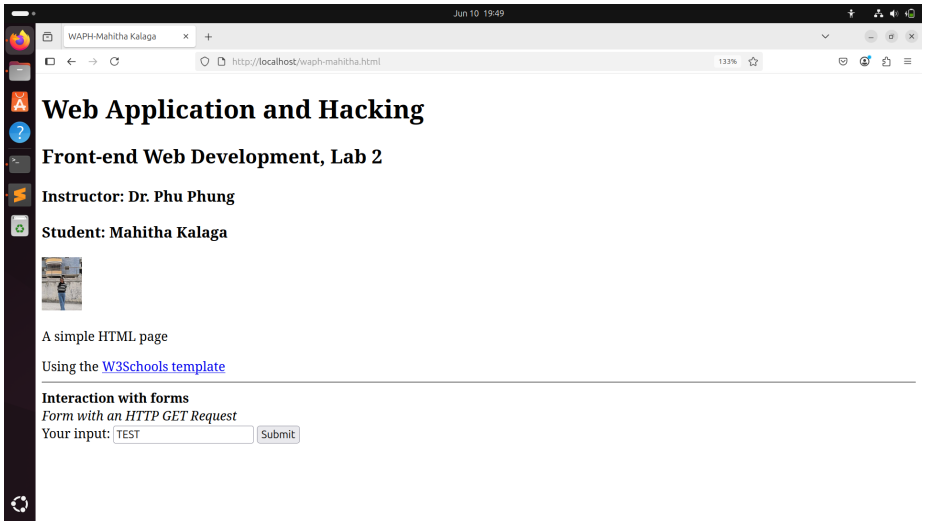
This lab focused on core front-end web development techniques using HTML, CSS, JavaScript, Ajax, jQuery, and Web API integration. The tasks reinforced lecture materials from weeks 4 to 6 and were implemented using a local Apache server. Through this lab, I gained experience in dynamic page updates, asynchronous requests, real-time clocks, and integrating third-party APIs into web interfaces.

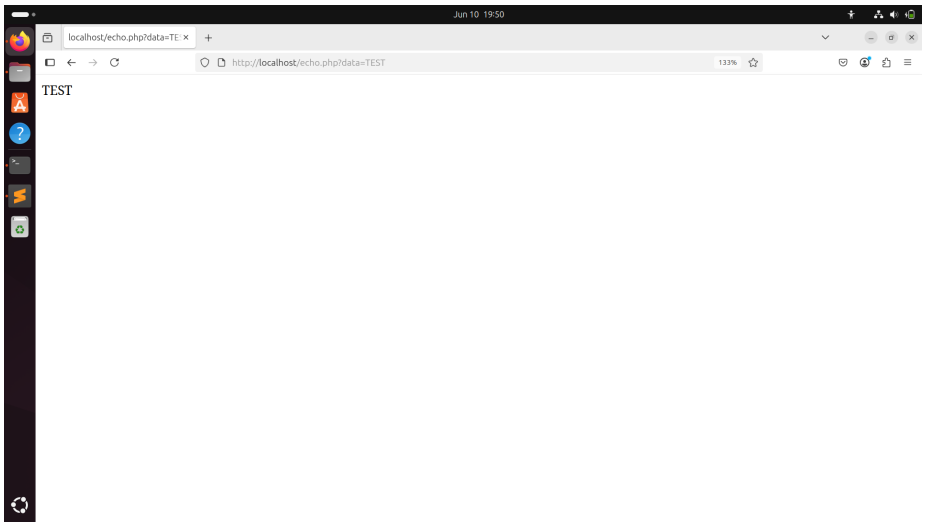
Lab's URL: Lab2

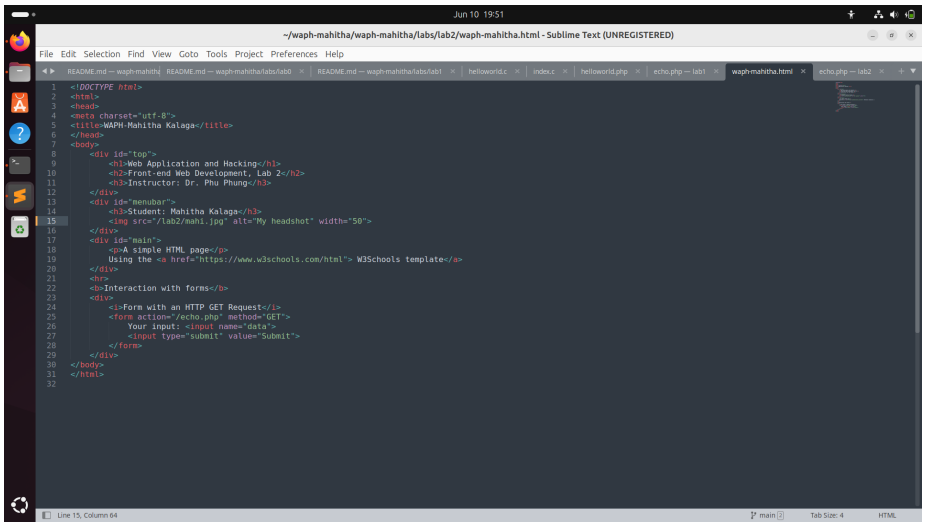
Part 1 - Basic HTML with Forms, and JavaScript

Task 1.A: A Basic HTML

Created waph-mahitha.html with proper HTML tags. Included: - Headshot image (150x150 pixels) - A simple form with and tags

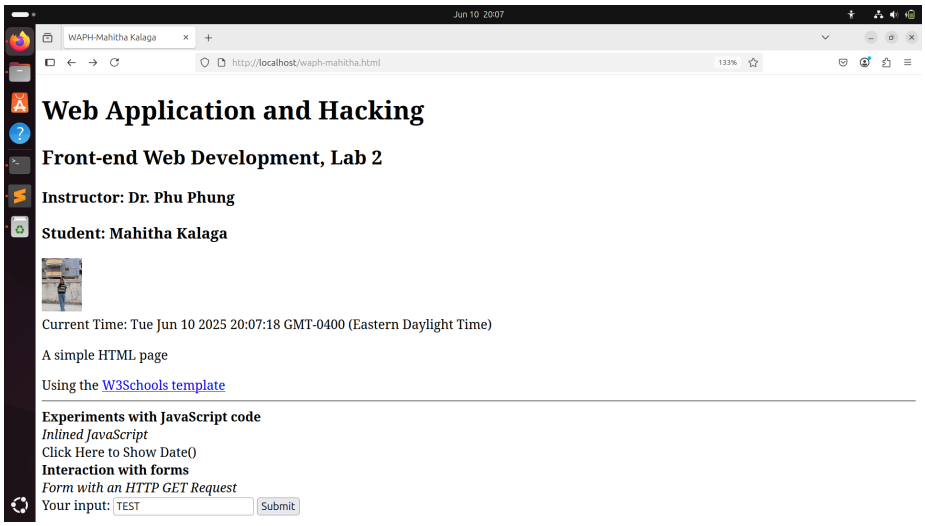


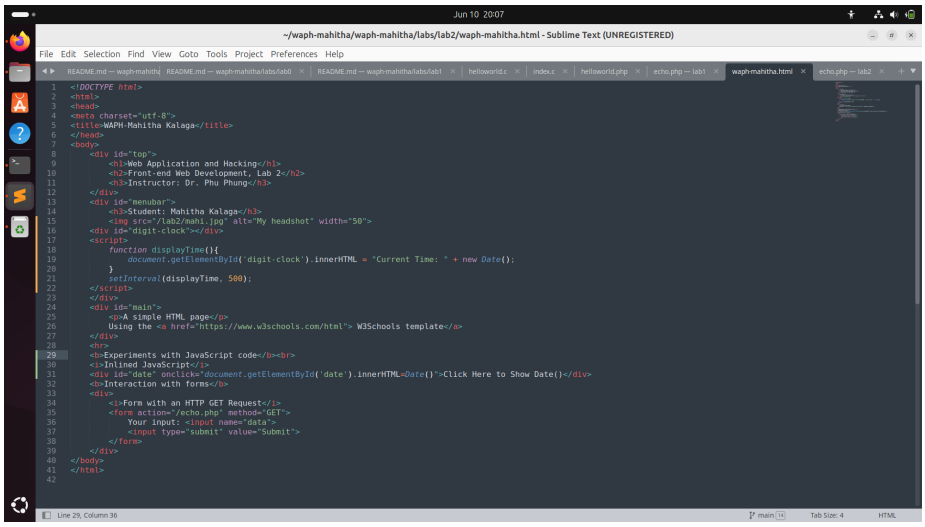




Task 1.B.i: Inline JavaScript

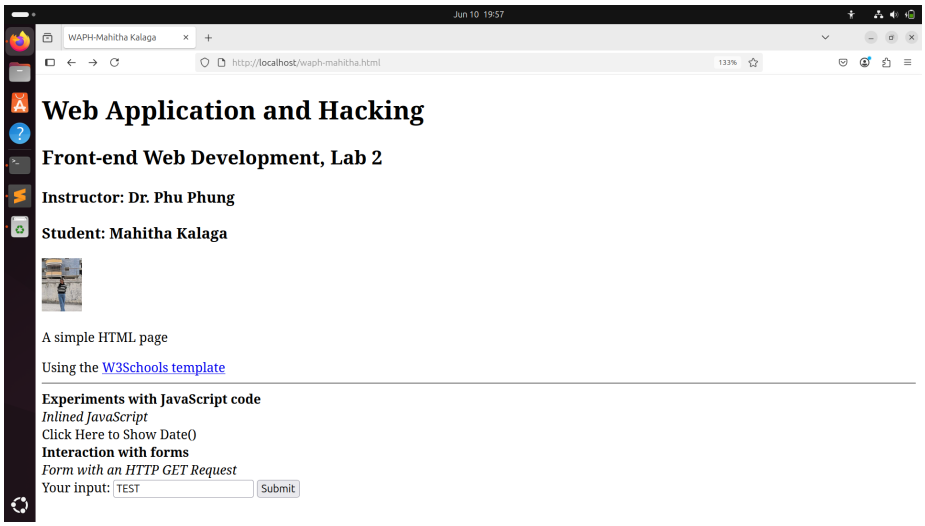
For the inline JavaScript task, I added an onclick event to a button that, when clicked, displays the current date and time. I also used the onkeypress attribute in a text input to log what keys were pressed to the browser console.

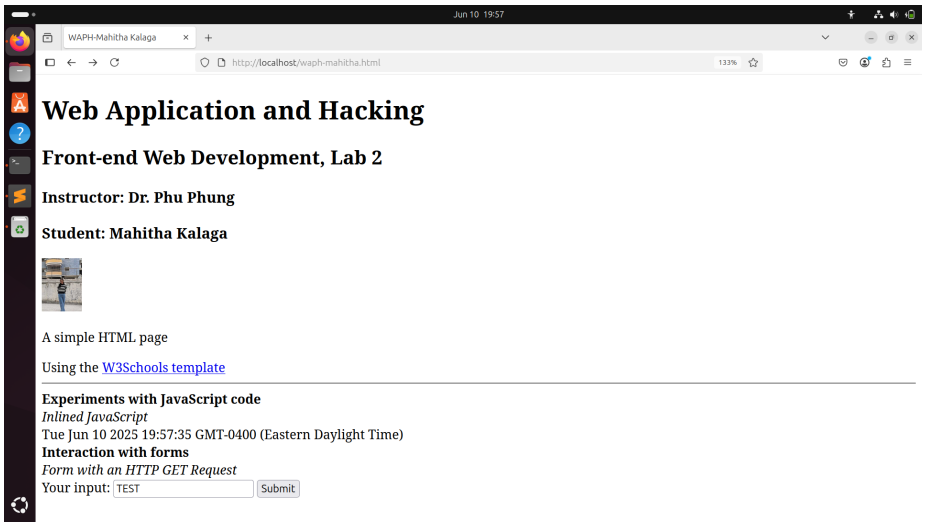


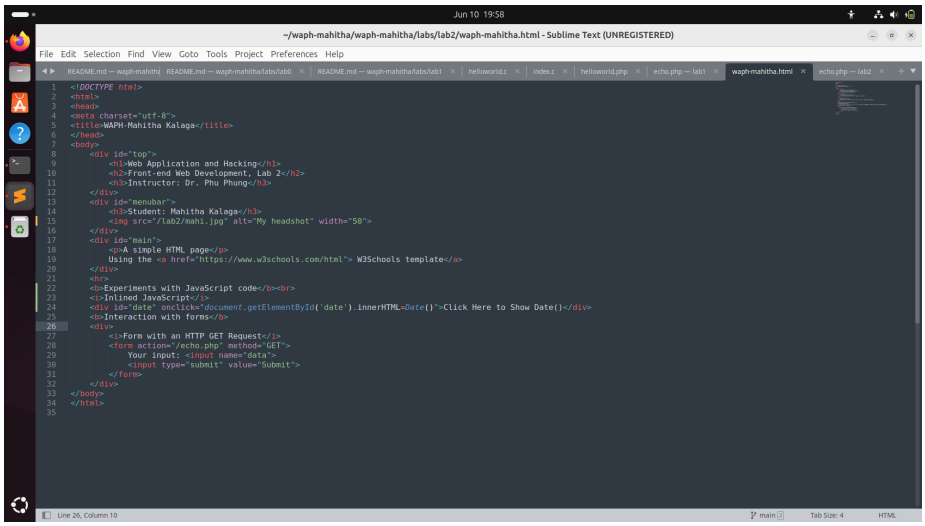


Task 1.B.ii: Digital Clock

Using setInterval() and JavaScript's Date() object, I created a live digital clock that updates every second. It displays the time in HH:MM,SS format. This was written inside a script tag and directly manipulated the inner content of a div.

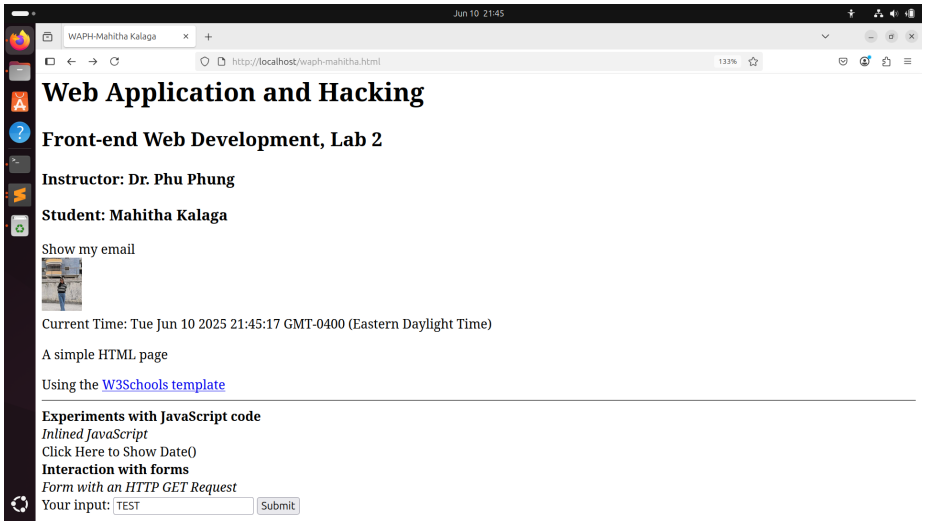

Jun 10 19:57
WAPH-Mahitha Kalaga
http://localhost/waph-mahitha.html
133%
Web Application and Hacking
Front-end Web Development, Lab 2
Instructor: Dr. Phu Phung
Student: Mahitha Kalaga
A simple HTML page
Using the W3Schools template
Experiments with JavaScript code
Inlined JavaScript
Click Here to Show Date()
Interaction with forms
Form with an HTTP GET Request
Your input: TEST
Submit

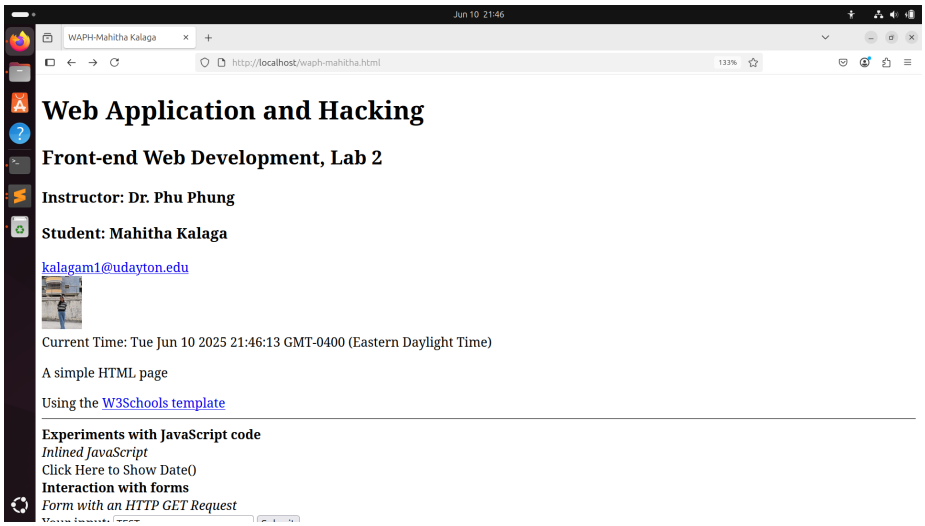

Jun 10 19:57
WAPH-Mahitha Kalaga
http://localhost/waph-mahitha.html
133%
Web Application and Hacking
Front-end Web Development, Lab 2
Instructor: Dr. Phu Phung
Student: Mahitha Kalaga
A simple HTML page
Using the W3Schools template
Experiments with JavaScript code
Inlined JavaScript
Tue Jun 10 2025 19:57:35 GMT-0400 (Eastern Daylight Time)
Interaction with forms
Form with an HTTP GET Request
Your input: TEST
Submit

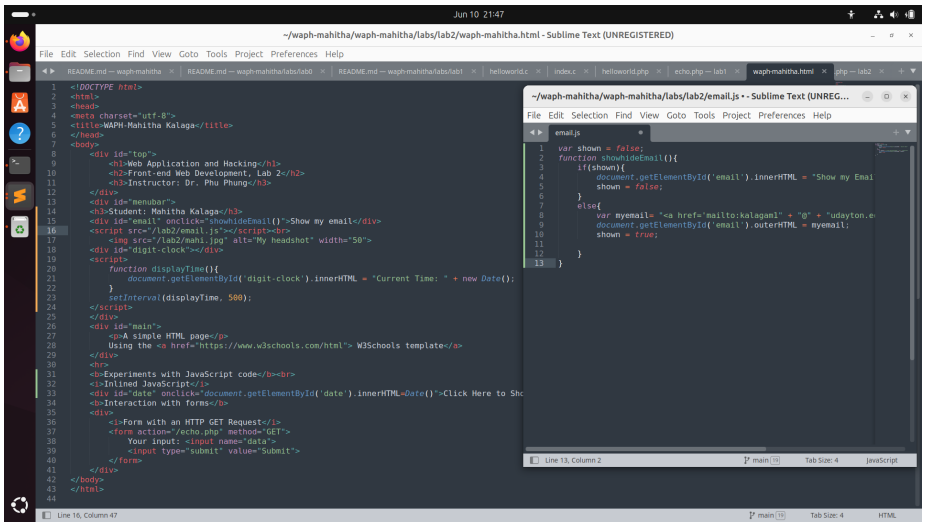


Task 1.B.iii: Show/Hide Email

Created a reusable and modular JavaScript file (email.js) that dynamically toggles the visibility of my email address. When the user clicks a div, the content switches between a label and a mailto: hyperlink. This component demonstrates external JS integration, conditional logic, and DOM element replacement.



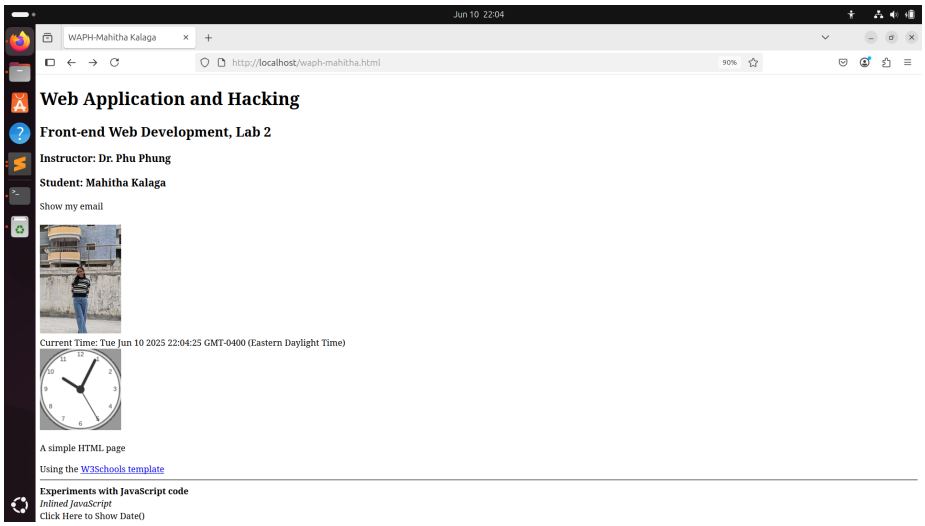


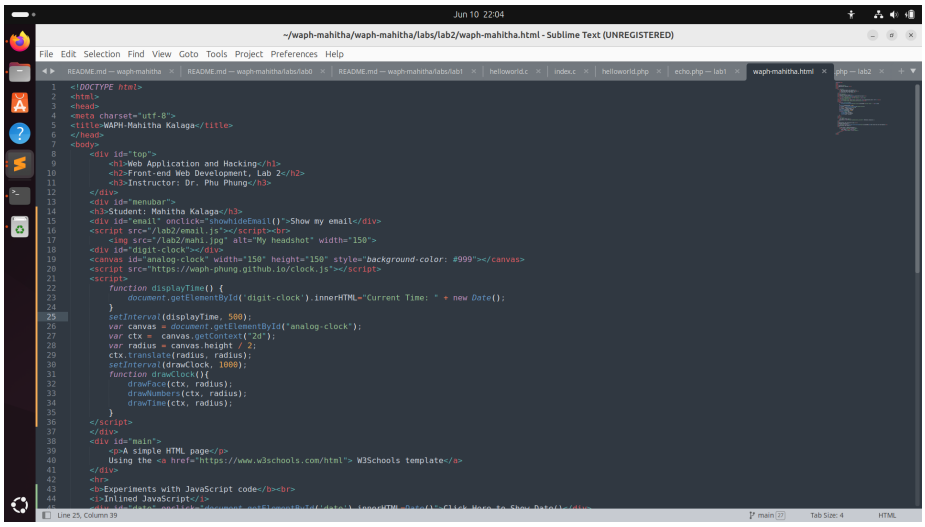


Task 1.B.iv: Analog Clock

Added an analog clock using a

element and the external script hosted at https:/waph-phung.github.io/clock.js. The script draws clock hands in real time using canvas rendering.

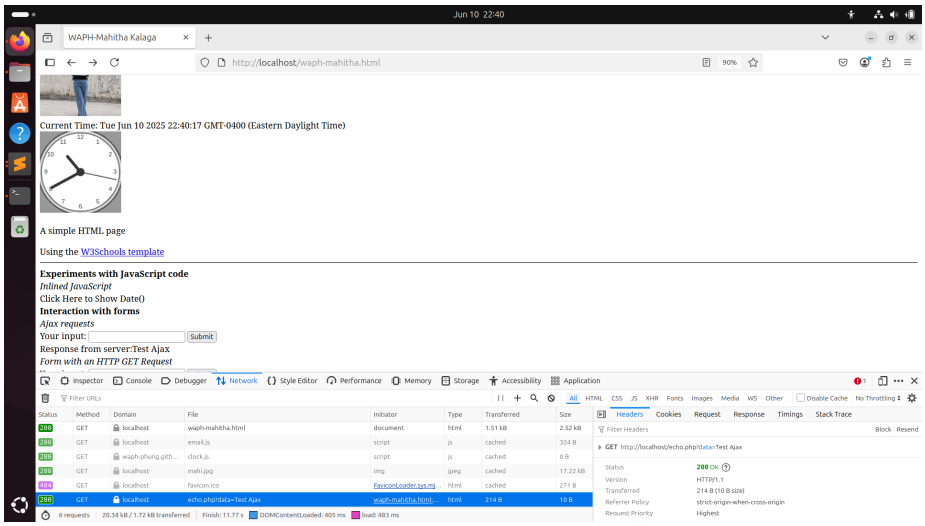


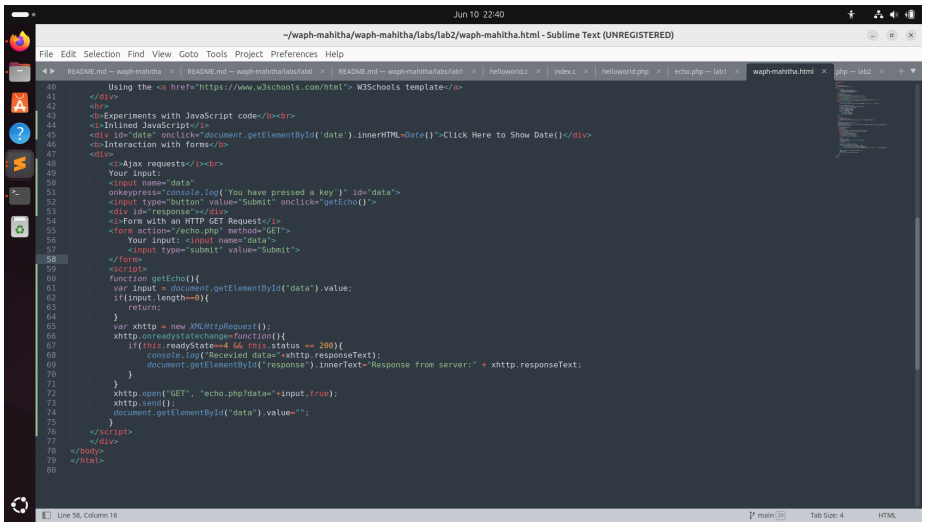


Task 2: Ajax, CSS, jQuery, and Web API Integration

Task 2.A: Ajax

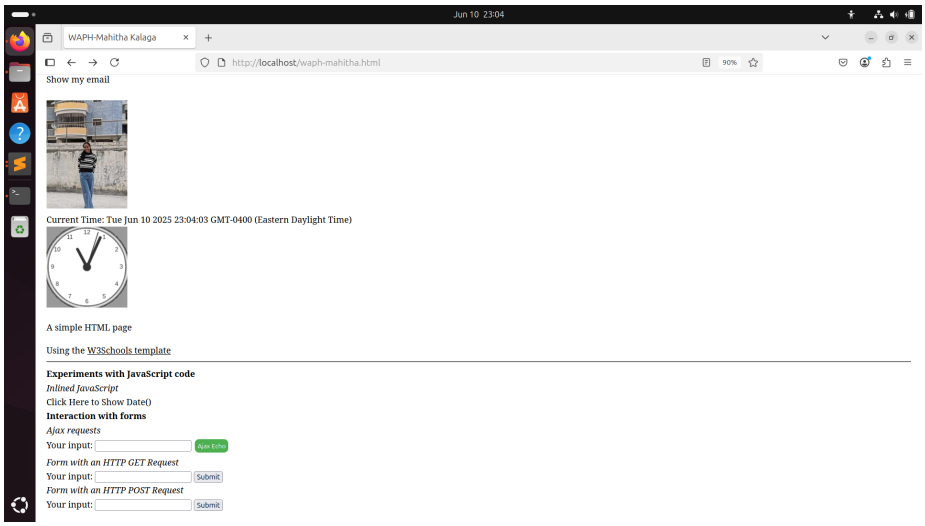
Implemented a form-driven Ajax request using XMLHttpRequest to send user input to echo.php. The servers response is retrieved and rendered within a target div element. By monitoring the request through browser developer tools, I gained insight into asynchronous communication and HTTP request/response lifecycles.

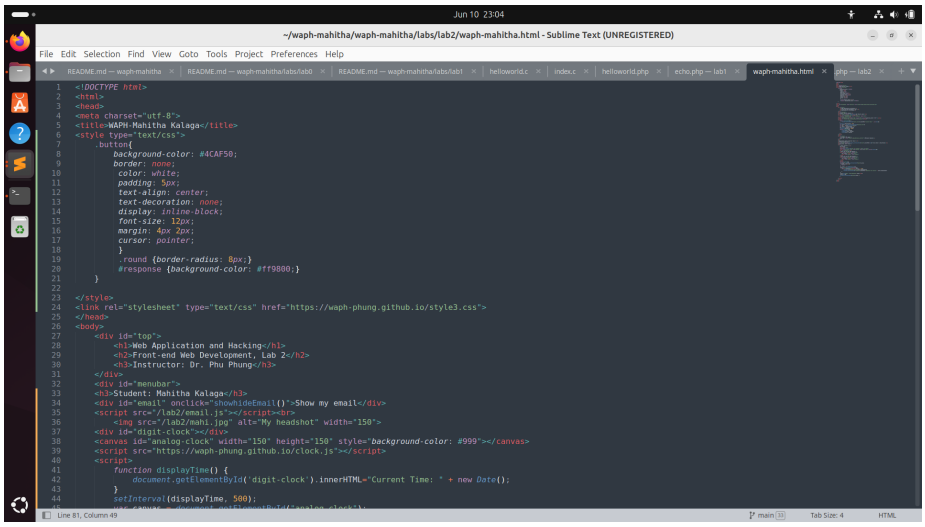




Task 2.B: CSS

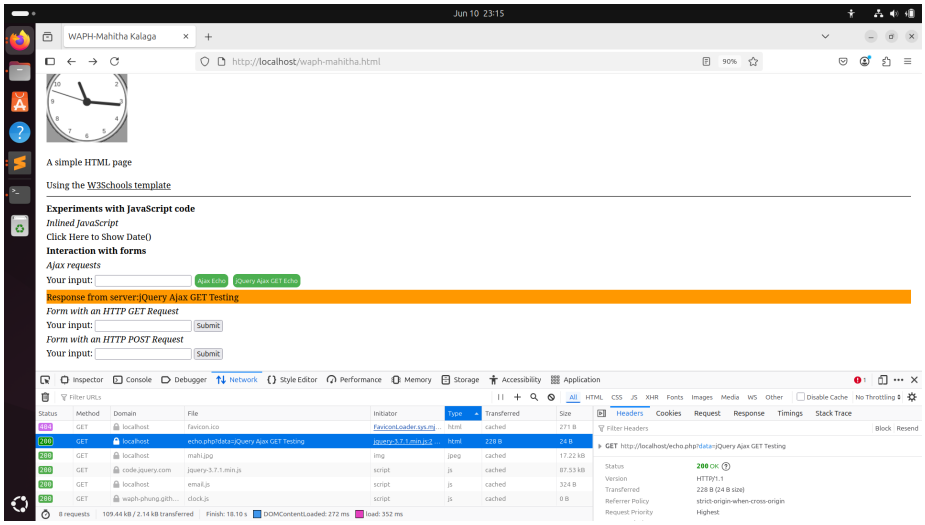
Demonstrated an understanding of different CSS application methods: - Inline CSS was used directly within HTML tags for quick styling. - Internal CSS was defined within a style block in the head for layout consistency. - External CSS was applied by linking to a remote stylesheet (https:/waph-phung.github.io/style3.css).

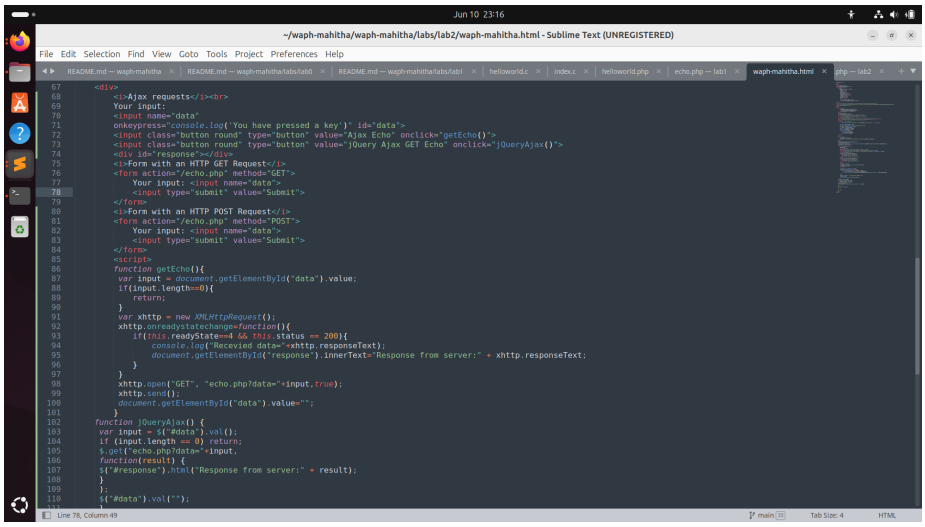


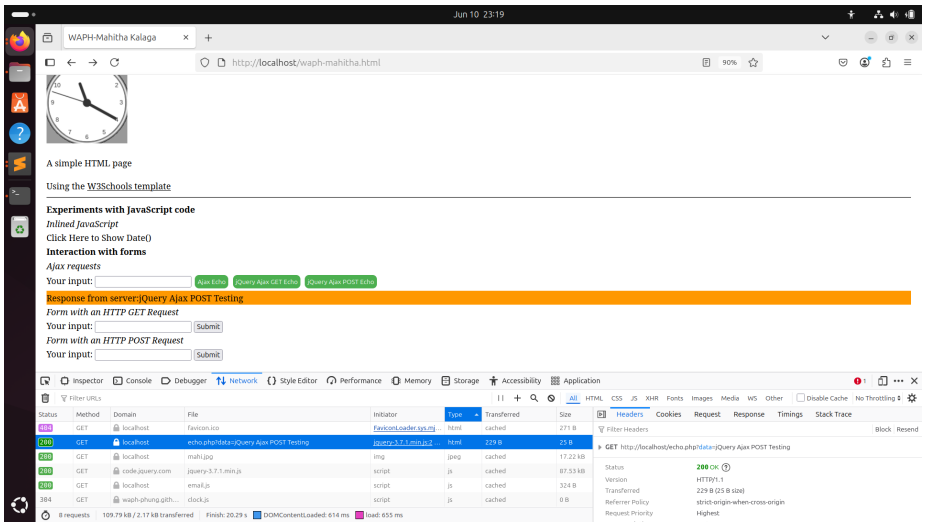


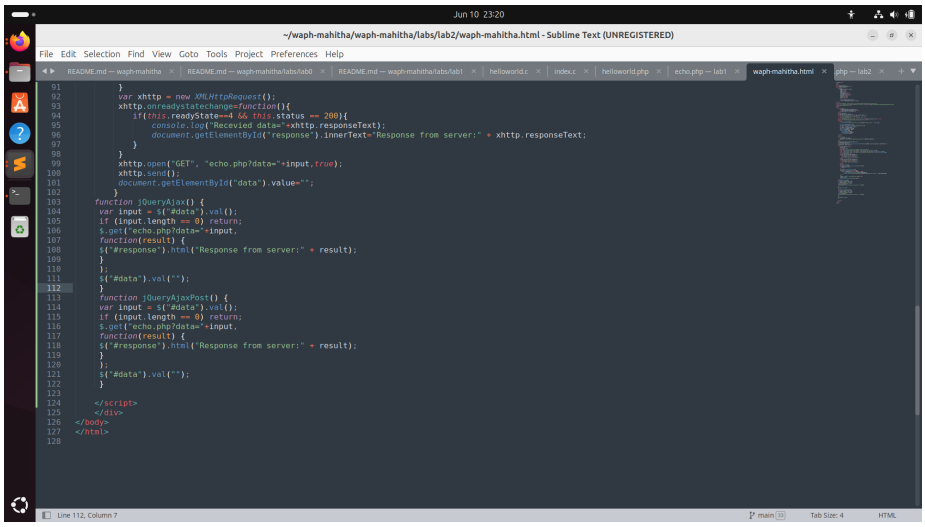
Task 2.C: jQuery

Included the jQuery library via CDN and developed Ajax functions using both $get() and $post() methods to interact with the echo.php endpoint. The responses were dynamically injected into the DOM.


Jun 10 23:15
WAPH-Mahitha Kalaga
http://localhost/waph-mahitha.html
A simple HTML page
Using the W3Schools template
Experiments with JavaScript code
Inlined JavaScript
Click Here to Show Date()
Interaction with forms
Ajax requests
Your input:
Ajax Echo
jQuery Ajax GET Echo
Response from server:jQuery Ajax GET Testing
Form with an HTTP GET Request
Your input:
Submit
Form with an HTTP POST Request
Your input:
Submit
Inspector
Console
Debugger
Network
Style Editor
Performance
Memory
Storage
Accessibility
Application
Status
Method
Domain
File
Initiator
Type
Transferred
Size
GET http://localhost/echo.php?data=jQuery Ajax GET Testing


Jun 10 23:16
~/waph-mahitha/waph-mahitha/labs/lab2/waph-mahitha.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
README.md — waph-mahitha
README.md — waph-mahitha/labs/lab0
README.md — waph-mahitha/labs/lab1
helloworld.c
index.c
helloworld.php
echo.php — lab1
waph-mahitha.html
Line 78, Column 49
Tab Size: 4
HTML

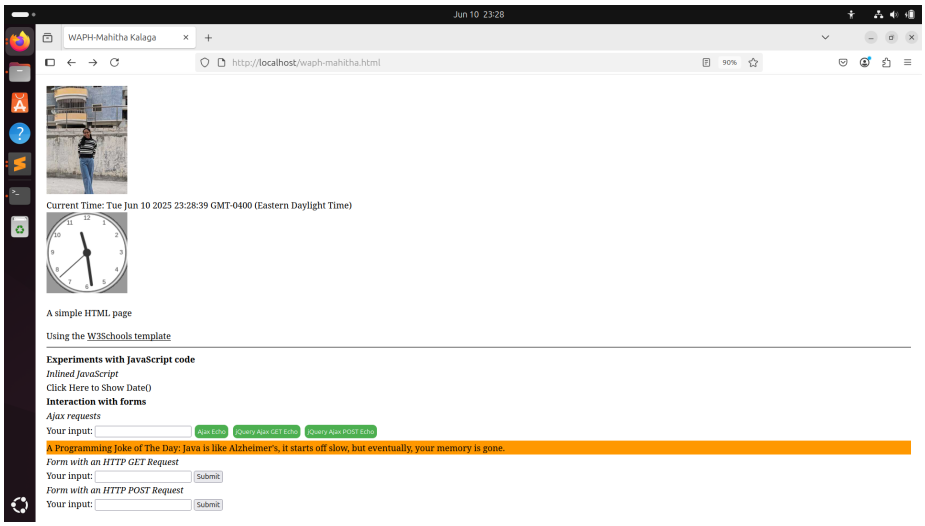


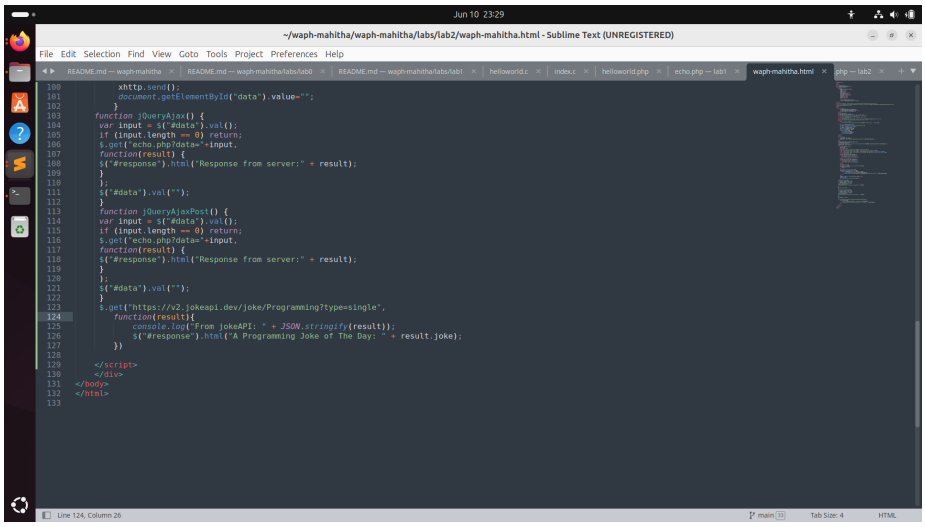


Task 2.D: Web API Integration

- i. Joke API

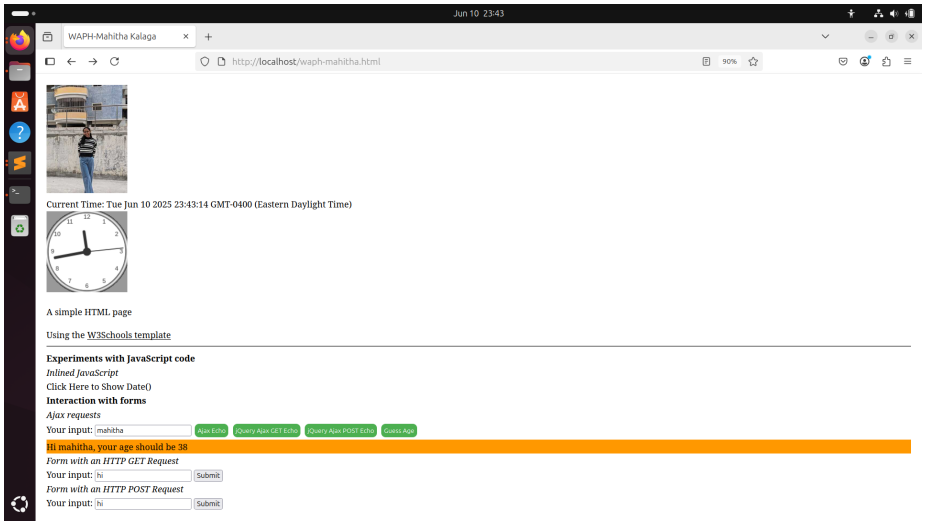
Used jQuery to fetch a random programming joke from https:/v2.jokeapi.dev/joke/Programming?type=single on page load. The returned JSON was parsed, and the joke was displayed in a div element. This integration demonstrated the practical use of external APIs to enhance user engagement.

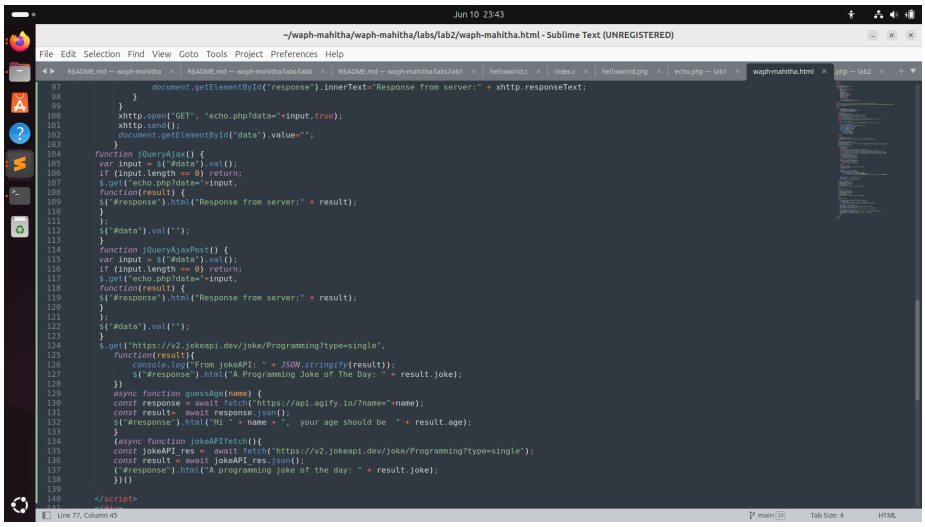




- ii. Agify API with fetch()

Used JavaScripts modern fetch() API to retrieve age prediction data from https:/api.agify.io/?name=... based on user input. The results were processed asynchronously and rendered within the page, providing a hands-on example of modern JavaScript promises and external API interaction.


Jun 10 23:43
WAPH-Mahitha Kalaga
http://localhost/waph-mahitha.html
90%
Current Time: Tue Jun 10 2025 23:43:14 GMT-0400 (Eastern Daylight Time)
A simple HTML page
Using the W3Schools template
Experiments with JavaScript code
Inlined JavaScript
Click Here to Show Date()
Interaction with forms
Ajax requests
Your input: mahitha
Ajax Echo
jQuery Ajax GET Echo
jQuery Ajax POST Echo
Guess Age
Hi mahitha, your age should be 38
Form with an HTTP GET Request
Your input: hi
Submit
Form with an HTTP POST Request
Your input: hi
Submit


Jun 10 23:43
~/waph-mahitha/waph-mahitha/labs/lab2/waph-mahitha.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
README.md — waph-mahitha
README.md — waph-mahitha/labs/lab0
README.md — waph-mahitha/labs/lab1
helloworld.c
index.c
helloworld.php
echo.php — lab1
waph-mahitha.html
php — lab2
document.getElementById("response").innerText="Response from server:" + xhttp.responseText;
xhttp.open("GET", "echo.php?data="+input,true);
xhttp.send();
document.getElementById("data").value="";
function jQueryAjax() {
var input = $("#data").val();
if (input.length == 0) return;
$.get("echo.php?data="+input,
function(result) {
$("#response").html("Response from server:" + result);
});
$("#data").val("");
}
function jQueryAjaxPost() {
var input = $("#data").val();
if (input.length == 0) return;
$.get("echo.php?data="+input,
function(result) {
$("#response").html("Response from server:" + result);
});
$("#data").val("");
}
$.get("https://v2.jokeapi.dev/joke/Programming?type=single",
function(result){
console.log("From jokeAPI: " + JSON.stringify(result));
$("#response").html("A Programming Joke of The Day: " + result.joke);
})
async function guessAge(name) {
const response = await fetch("https://api.agify.io/?name="+name);
const result= await response.json();
$("#response").html("Hi " + name + ", your age should be " + result.age);
}
(async function jokeAPIfetch(){
const jokeAPI_res = await fetch("https://v2.jokeapi.dev/joke/Programming?type=single");
const result = await jokeAPI_res.json();
("#response").html("A programming joke of the day: " + result.joke);
})()
</script>
Line 77, Column 45
main
Tab Size: 4
HTML