WAPH-Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Student

Name: Shivani Eli

Email: elisi@mail.uc.edu

Figure 1: Shivani Eli

Lab 2 - Front-end Web Development

Overview

In this lab I done two tasks, overall I have learnt and covered the Basic HTML forms,Javascript for the front-end, as part of Task-1. I have also used the Ajax and jQuery basics to create simple backend GET POST http requests, with echo.php file and some basic CSS template as part of Task-2. In this lab2, as part of task2 have done the basic Web API integration, to load anddd display a programming joke, as well as a simple age guess.

https://github.com/ShivaniEli/waph-elisi/blob/main/labs/lab1/README.md

Task 1: Basic HTML with forms, and JavaScript

In this task-1 I have done the basic HTML forms, analog-clock, digital, clock with some javascript.

a. HTML In this task, I have created a HTML file - 'waph-elisi.html'.It has the basic html tags and my headshot image is placed.

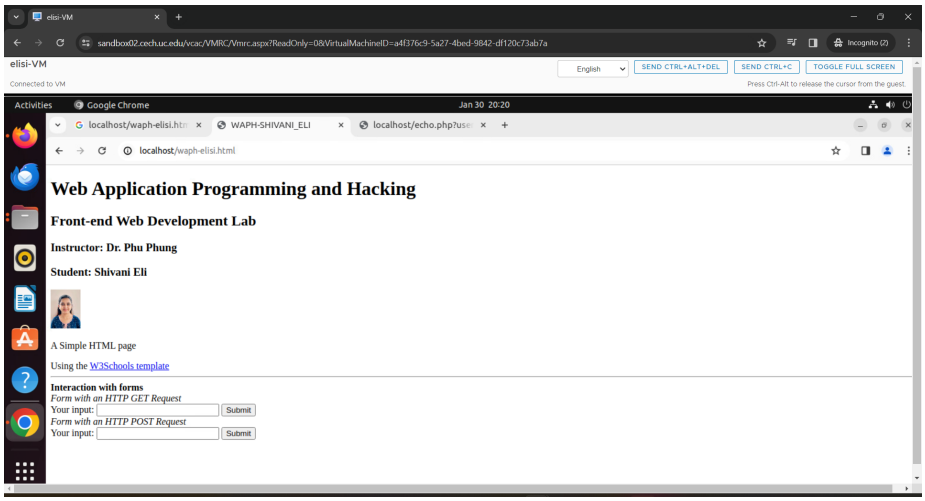


Figure 2: Simple HTML webpage with headshot

b. Simple JavaScript In this part b task-1, I have done the following using javaScript:

- For the web-page, I have done a javascript code for displaying current data-time when the key is pressed. I have included the javascript in the html file, within
  tag (inline JS code). After saving the html file, in the terminal ran the command “sudo cp waph-elisi.html /var/www/html/” within labs/lab2. Then accessed the link “https://localhost/waph-elisi.html” for opening the created webpage.
- In the top displayed the digital-clock with some inlin e JS code, a code included in html file within
  tag.
- Then wrote JS code for show/hide email when onclicked, this JS code was stored in an external email.js file and linked this external js code file in the .html file using
  . For the email, to be displayed in the same terminal, pass the email.js file -> “sudo cp email.js /var/www/html/” as the file is external need to be passed to web server along with the updated waph-elisi.html file.

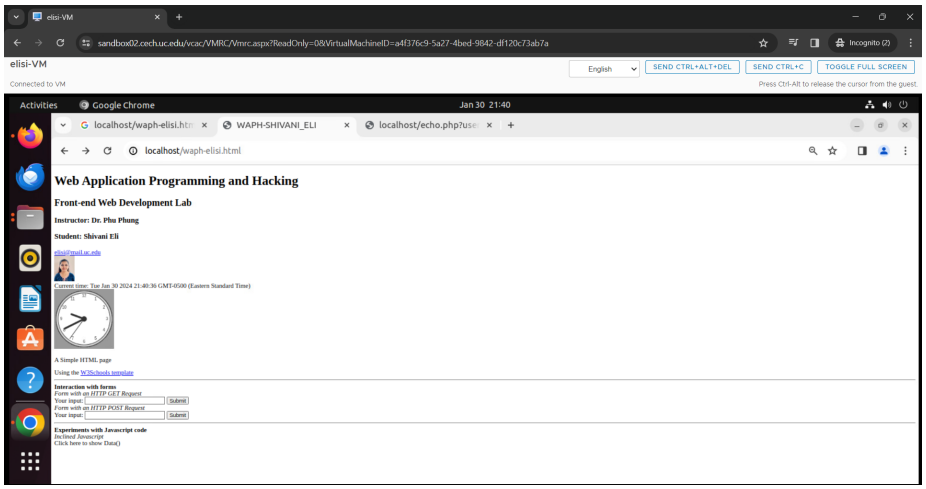


Figure 3: In line javascript for displaying data/time after key pressed

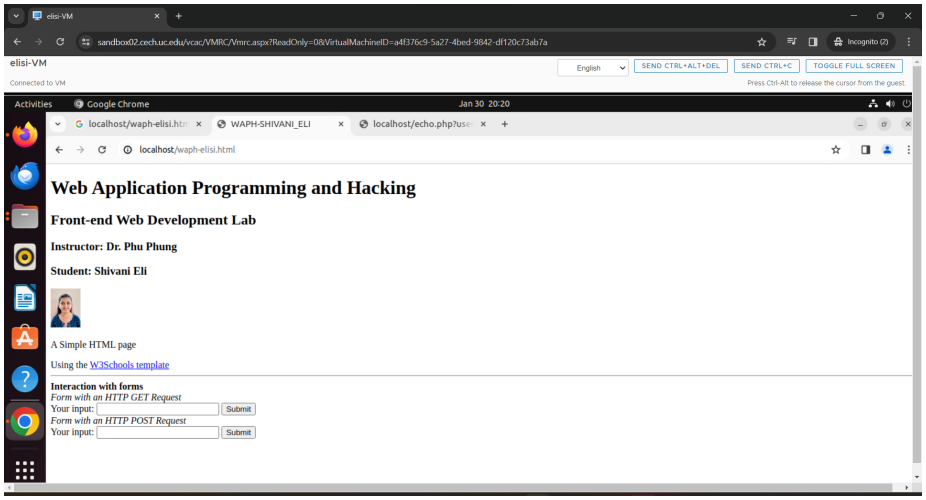
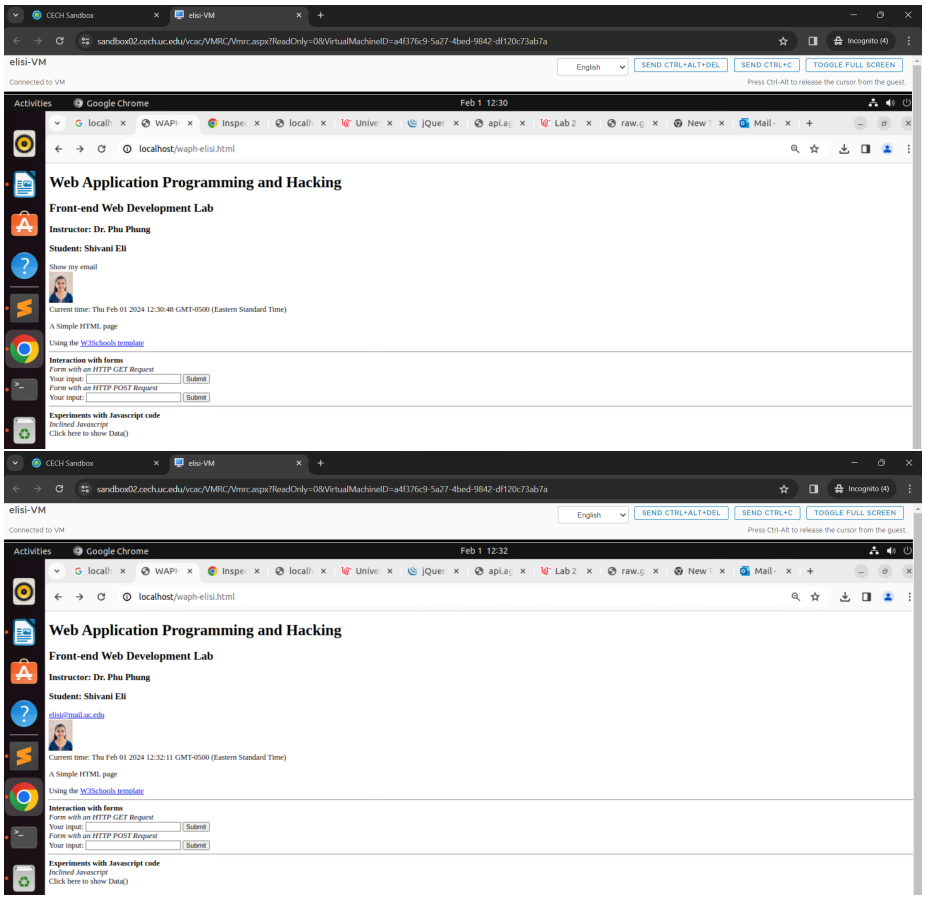


Figure 4: Simple HTML webpage with headshot



- With external JS code, developed a 2d analog clock with the given as the script scourse to draw the clock Face, the Numbers and the JS code to show the time (clock hands) and along with some inline JS code in the script tag. After scripting, and saving the code, again ran the "sudo cp waph-elisi.html /var/www/html/" command to push the update file to web-server, done this in every changes made to the file till the end of the lab-2 tasks. Then reloaded the webpage in the chrome browser, to see the analog-clock.

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

In the Task-2 have done some basic Ajax Echo,with html forms, and CSS, jQuery code and Web API integration.

a. Ajax For task-2, new html template added, with

, a form with a button, and element for taking the user input, and used the echo.php file in lab1, as a back-end file for this lab2 which cointains php echo to

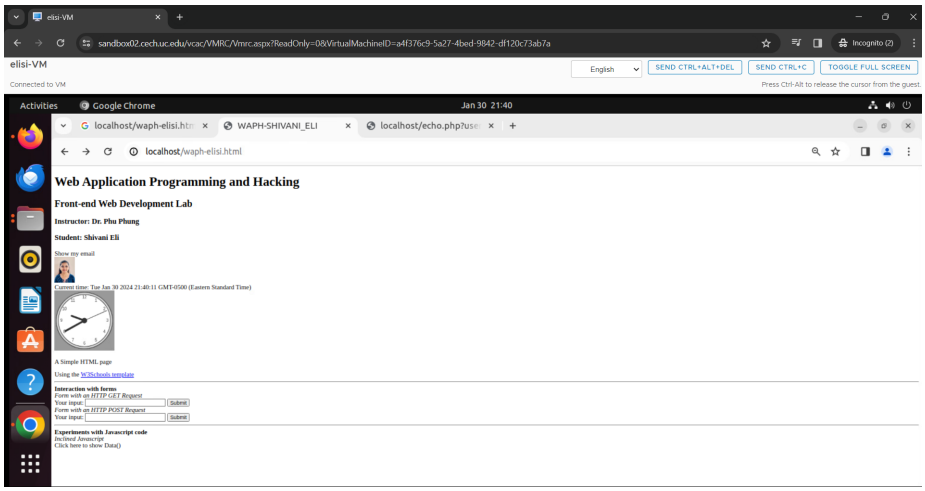


Figure 5: Analog clock 2D developed with external javascript

display the user input.:

- In the .html file created a new button in the so after getting input from user, need to click new button; submit, displays the user input.And logged in keypressed.
- Then created a text box in the html for, tag to get user input and a then included the action="/echo.php", here the echo.php file is used from lab1, as backend. Then an Ajax GET request is sent, with some ajax code, within
  tag, to the echo.php with the useeeeeeeer input. This displayed the user input along with echo.php content in a new tab, "loaclhost/echo.php?user="+input_value.

So, for above, in the devloper tools, in the network tab I obserevd the Ajax requests, where it is observed that the ajax http request is sent to server successfully, (echo.php?user=input) and its response, is recorded in the server, with simple echo statement.

b. CSS For css, used inline css with style attribute in for color to be blue.

For internal css the css for button class, round class, and response id elements were added, within the

tag in the

part of waph-elisi.html file.

Then linked an external css template number 2, with

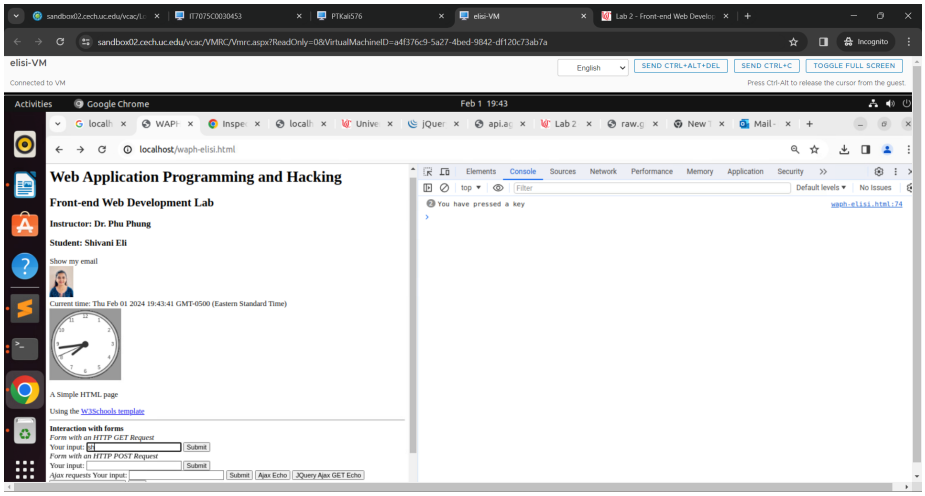


Figure 6: webpage when keypressed in the input

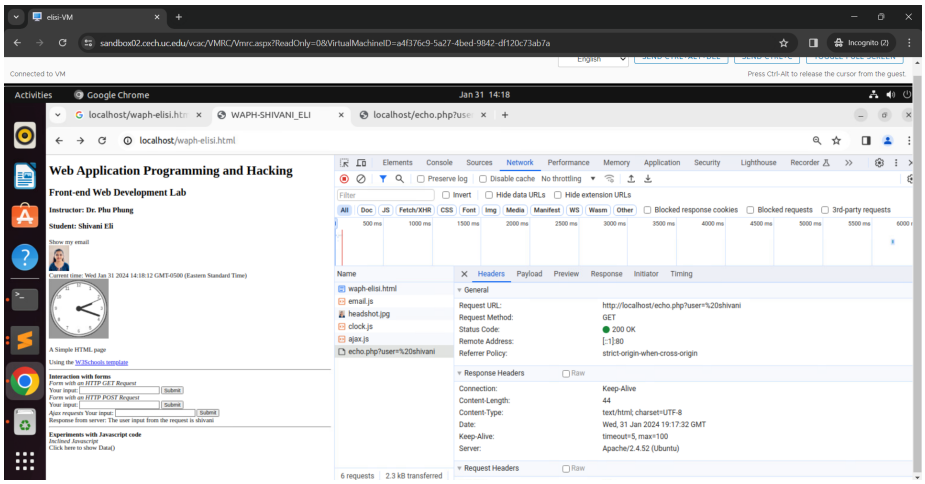


Figure 7: Ajax GET request

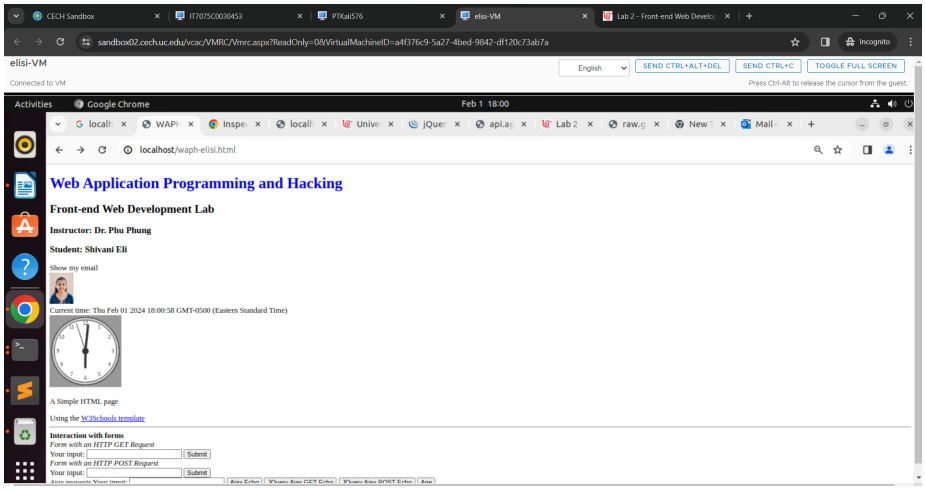


Figure 8: inline css

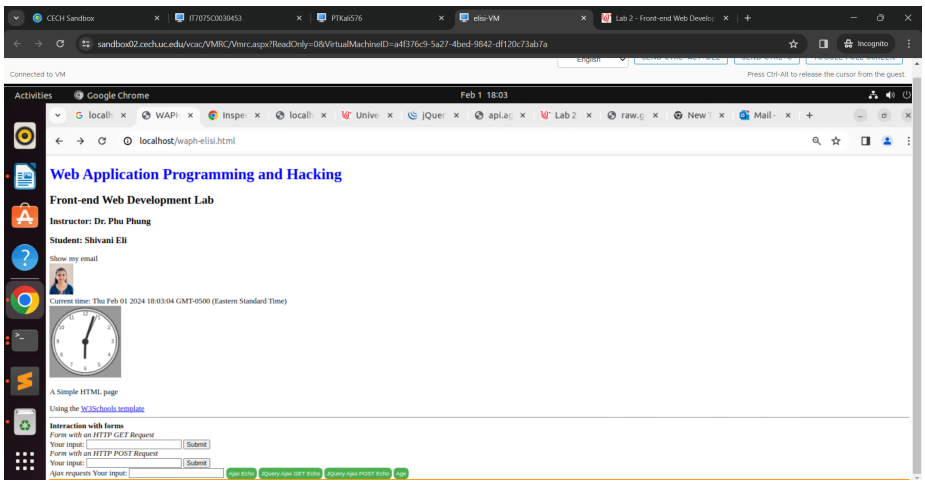


Figure 9: internal css

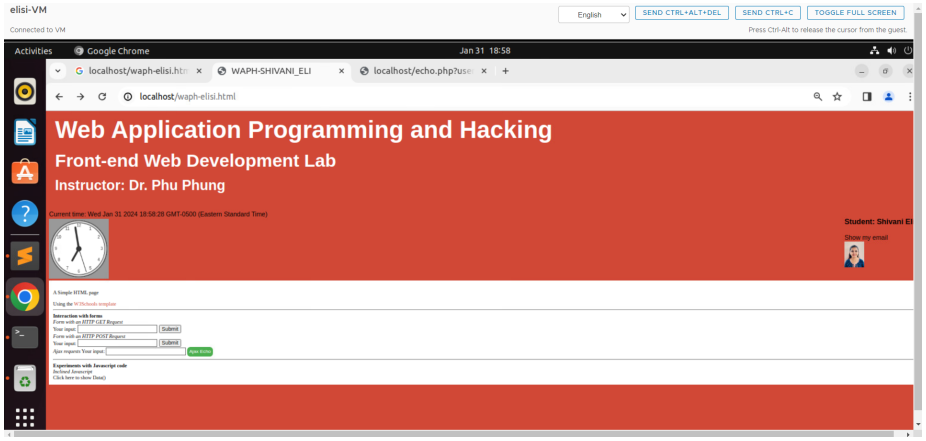


Figure 10: external css

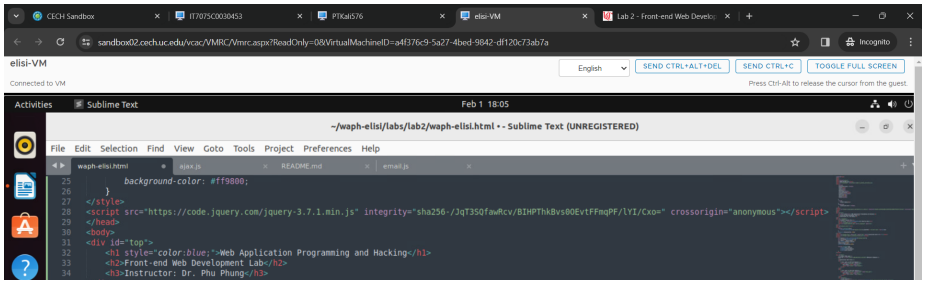


Figure 11: inline css code

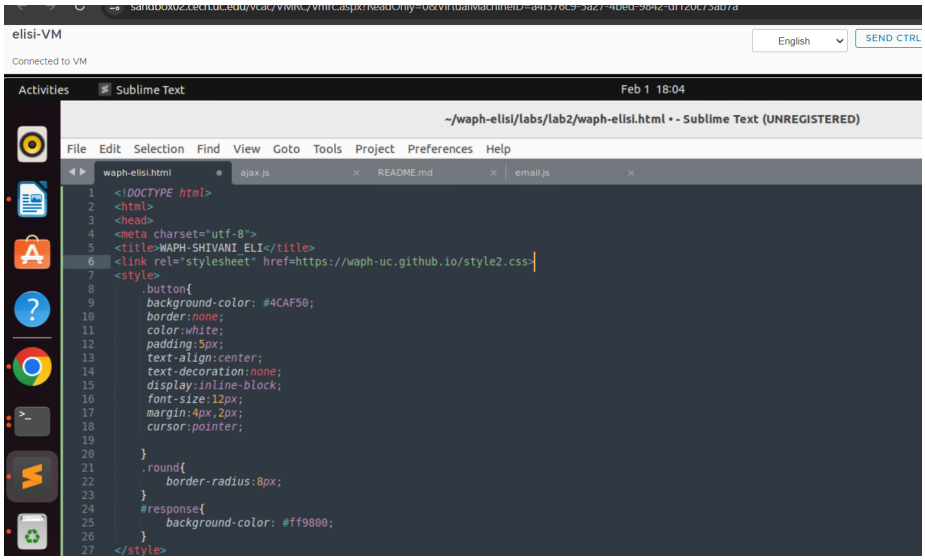


Figure 12: internal and external css code

c. jQuery A jQuery library file was installed to use the jQuery. And performed the following:

i. In the input user, added another button as "jQuery Ajax GET" for sending an Ajax GET request to the echo.php with user input, using iQuery $.get(), in the getjQueryAjax function within the

tag in the html file, when clickedon to the button jQuery Ajax GET.This displays the Server response within the

response. In the network tab of developer settings of the webpage, it is observed that the request sent to echo.php server is the GET request.

ii. So in the same way, another button added to send the Ajax POST request with jQuery to the same echo.php, with $.post() jQuery postjQueryAjax function when clicked on to button "jQuery Ajax POST". The result is displayed in the div elemnt with id = "response". In thhhhhe network tab it is observed that teh Ajax POST request is sent, with response same.

d. Web API integration **i.** In this last task I have performed the API integration for dsplaying a joke, in the div element in html, with the api link - in the

tag. https://v2.jokeapi.dev/joke/Programming?type=single]

The above is written with the jQuery Ajax code to send request to display the response of the http request to this random joke in the webpage, it changes whenever the page is loaded.

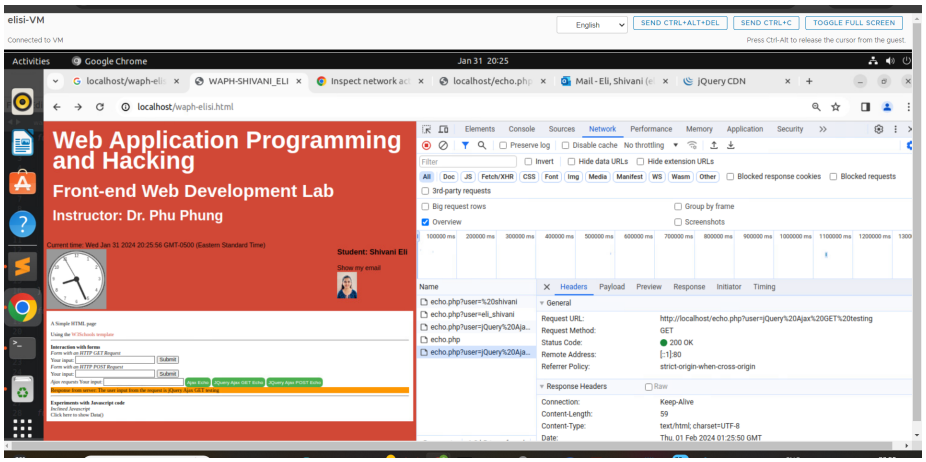


Figure 13: jQuer ajax GET

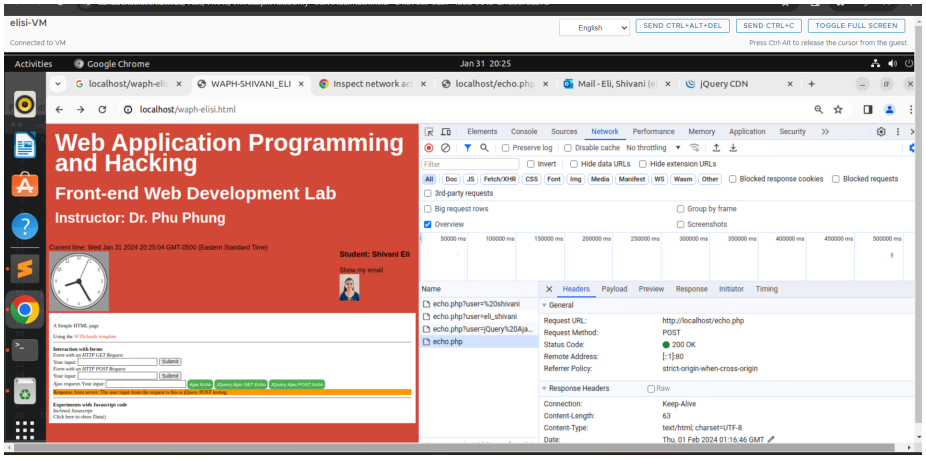


Figure 14: jQuery POST

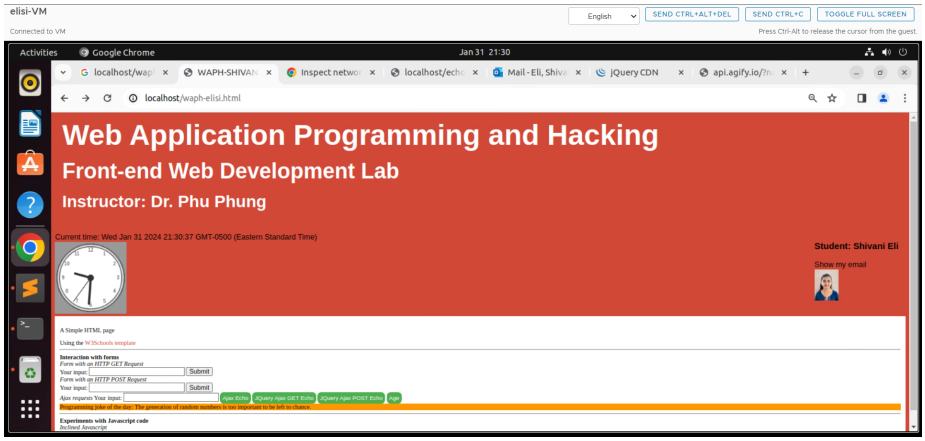


Figure 15: joke api

ii. For integrating the API for guess age with fetch in the jQuery Ajax code with api - https://api.agify.io/?name=input

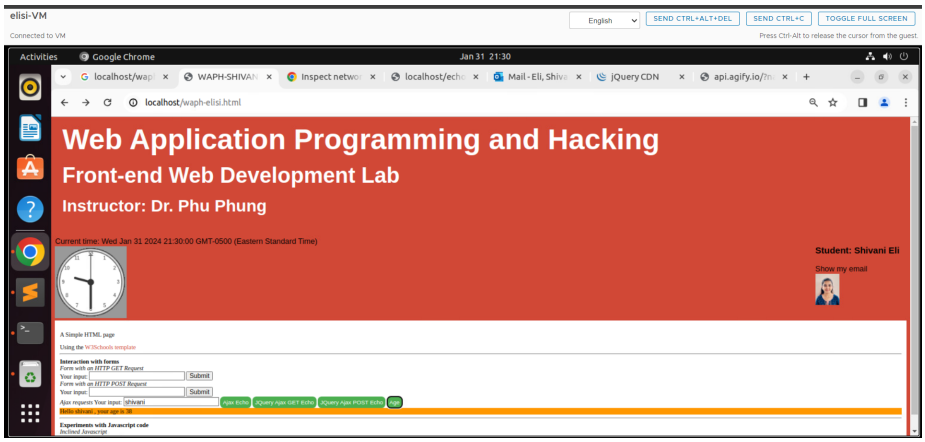


Figure 16: fetch agify

For this, when teh api method is called the the response is displayed in the div response, for this new button age is created, when the user enter the input and click on to this button then the jQuery ajax, guessAge function is called, where the fetch is used to get the api content, to displaye the results in the div response element. The browser network tab is used to see the jQuery ajax requests to api, and its response.

.

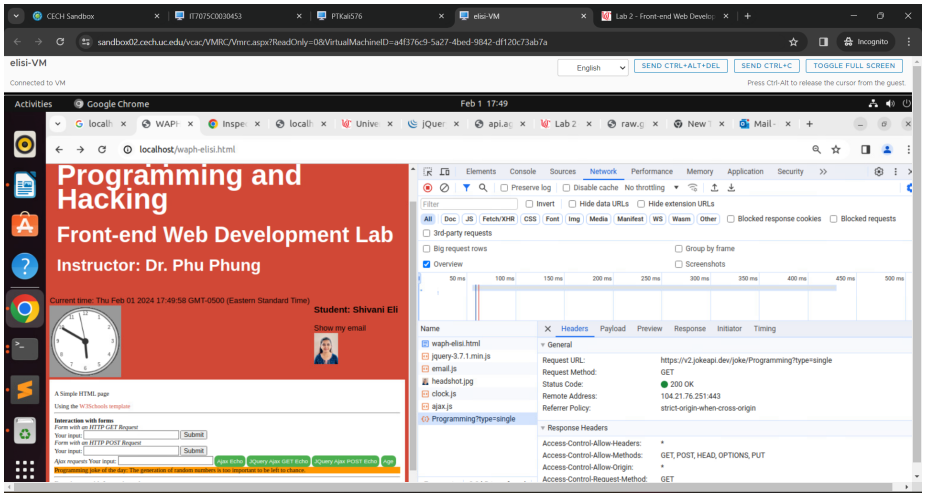


Figure 17: network server for api