

# **Web Application Programming and Hacking**

**Instructor : Dr Phu Phung**

**Student**

**Name :** Likhitha Tirumalareddy

**Email :** tirumala@mail.uc.edu

**MyShort-bio:** I possess a keen enthusiasm for developing interactive and responsive webpages.

## **Individual Project 1**

The project revolves around crafting a professional portfolio that encompasses details such as education, experience, employment history, skills, and other relevant information. You can review my portfolio by following the link provided below.

My Portfolio

Repositoy URL

A link to this course page <https://likhithatr.github.io/index.html> was provided in my education section in portfolio , you can see screenshot for reference.

### **Non-technical requirements (20 pts)**

Utilize an open-source CSS template or framework like Bootstrap for styling. I opted for Tailwind CSS to design the entire page, employing a CDN link for convenience. The specific CDN link used is `<script src="https://cdn.tailwindcss.com"></script>`. Tailored towards a potential employer, this profile is subject to evaluation as part of a job application. My skills and expertise, tailored to meet the needs of my target employer, are detailed in both my portfolio and the accompanying screenshot..

Include a page tracker, for example: <https://analytics.google.com/analytics/web/?pli=1#/p428016586/reports/in>

### **Technical requirements (50 pts)**

Implement basic JavaScript code worth 20 points by incorporating jQuery and an additional open-source JavaScript framework or library. This code should cover the Lab 2 functionalities, comprising a digital clock, an analog clock, a show/hide mechanism for your email, and one more functionality of your choosing (each worth 5 points).



Figure 1: install Likhetha

```

<div class="card-education h-auto my-10 p-4 one w-1/2 bg-green-500 rounded-2xl">
  <h3 class="font-bold text-gray-700 text-xl">Education</h3>
  <div class="Masters" id="Masters">
    <p class="font-text-white">University of Cincinnati</p>
    <p class="font-text-white">Masters in Information Technology</p>
    <p class="text-white" href="/uaph.html" targets="_blank"><span class="text-white">Course: </span><span class="text-blue-500">Web Programming and Ethical Hacking</span></p>
    <p class="text-gray-700"> 2023 Aug - 2025 Apr</p>
  </div>
  <div class="Education" id="Education" px-2 = "16">
    <p class="font-bold text-gray-700 text-xl">Education</p>
    <div class="Masters" id="Masters">
      <p class="font-text-white">Rvr&Ic college of Engineering</p>
      <p class="text-white">Masters</p>
      <p class="text-white">Masters in Computer Application's</p>
      <p class="text-white" href="#" targets="_blank"> </a></p>
      <p class="text-gray-700"> 2017 June - 2020 June</p>
    </div>
  </div>

```

Figure 2: install Integration

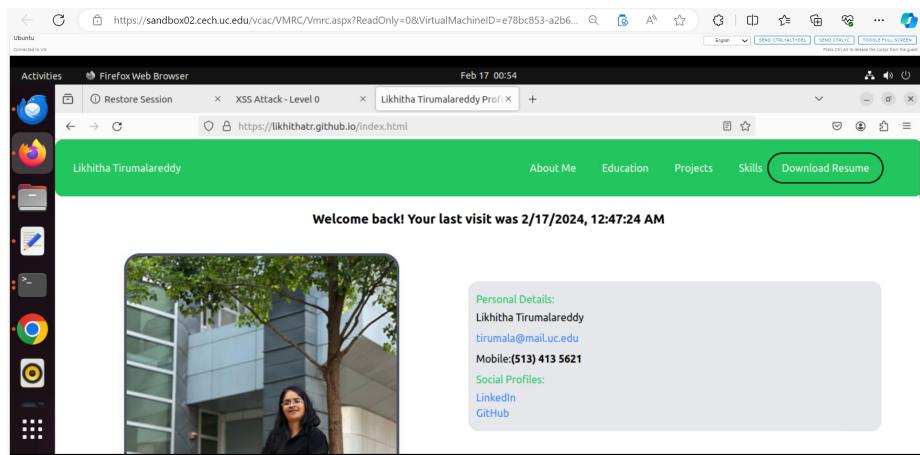


Figure 3: install FirstScreen

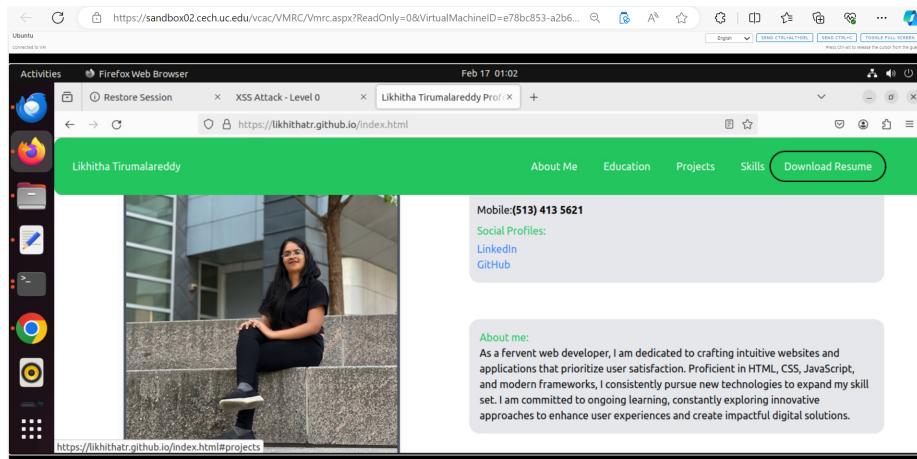


Figure 4: install SecondScreen

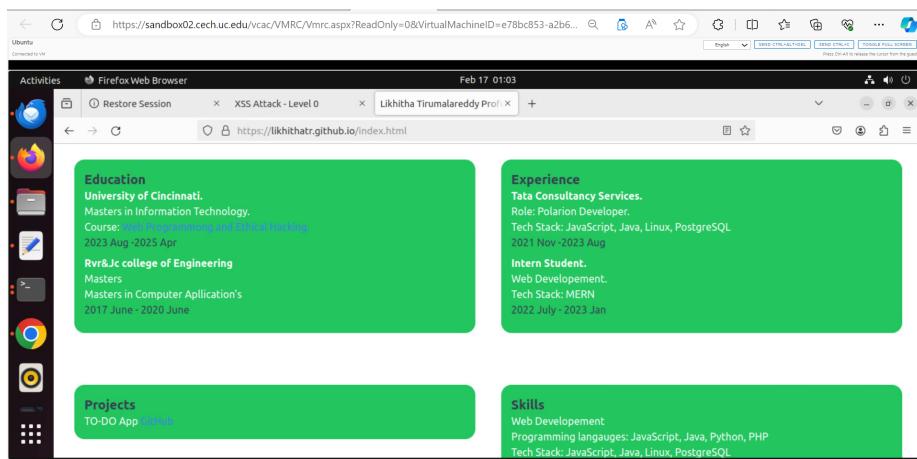


Figure 5: install ThridScreen

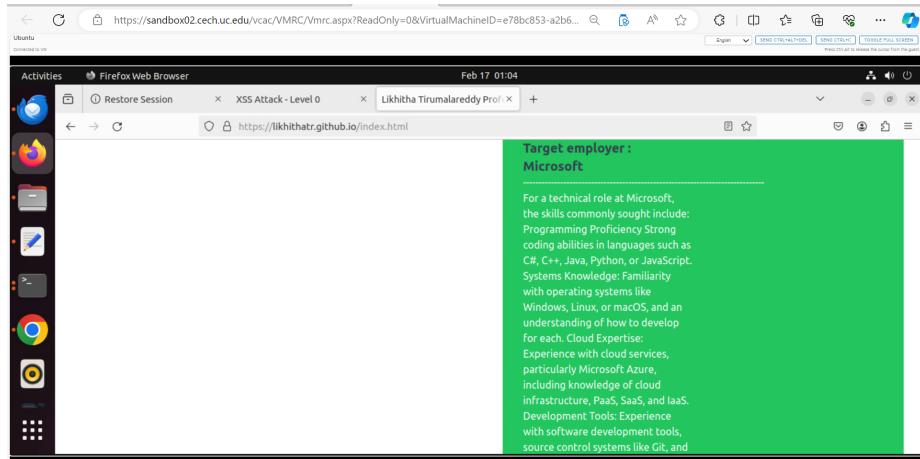


Figure 6: install FourthScreen

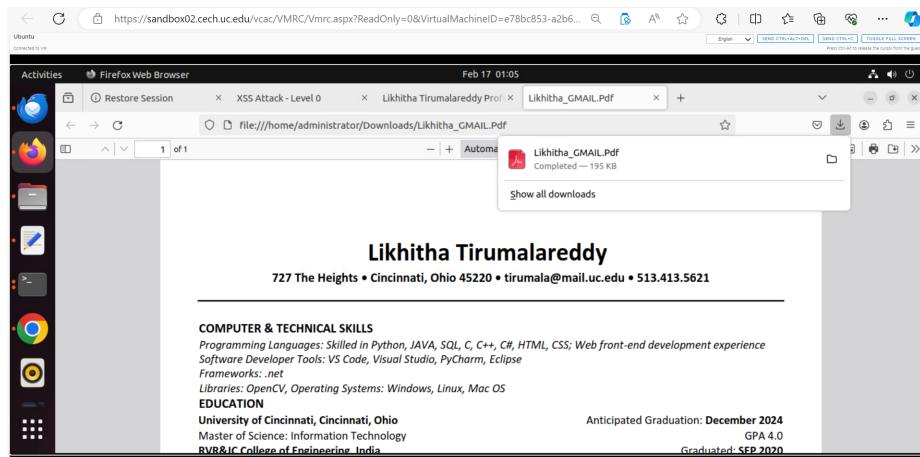


Figure 7: install FiveScreen

```

<!DOCTYPE html>
<html lang="en">
<head>
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-PB6QQY2GH"></script>
<script>
window.dataLayer = window.dataLayer || [];
function gtag() {dataLayer.push(arguments);}
gtag('js', new Date());
gtag('config', 'G-PB6QQY2GH');
</script>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Likitha Tirumalareddy Professional </title>
<script src="https://cdn.tailwindcss.com"></script>
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css" integrity="sha512-DTOQ9RwCH3ppGqchEA1BIIZOC6xxalwEsw9c2QqeAifj" integrity="sha512-JqT3QfawRcv8IHPTk8v80EvtFFmqPF/lYI/Cxa=" crossorigin="anonymous">
<script>
function setCookie(name, value, days) {
    var expires = "";
    if (days) {
        var date = new Date();
        date.setTime(date.getTime() + (days * 24 * 60 * 60 * 1000));
        expires = " "; expires += date.toUTCString();
    }
    document.cookie = name + "=" + value + expires + "; path=/";
}
function getCookie(name) {
    var nameEQ = name + "=";
    var ca = document.cookie.split(';');
    for (var i = 0; i < ca.length; i++) {
        var c = ca[i];
    }
}

```

Figure 8: install Googlecode

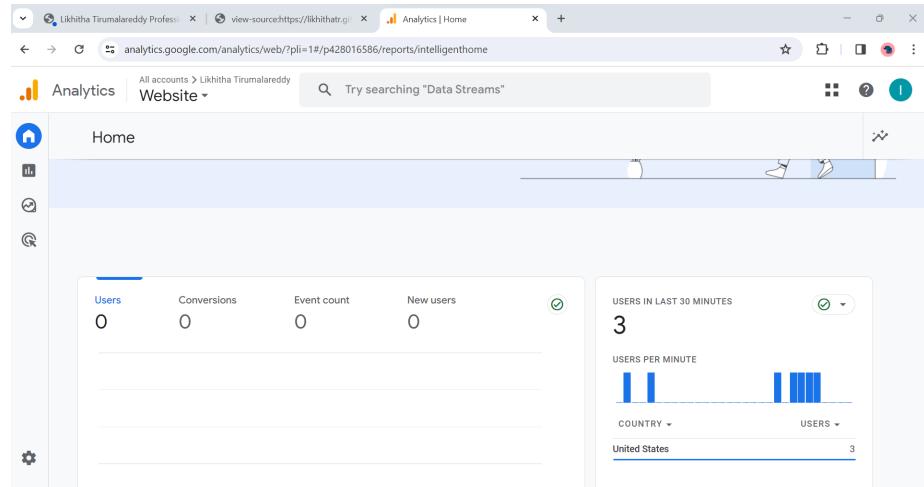


Figure 9: install GoogleAnalytics

```
32 <div class="p-1w">
33   <span class="text-x1 font-bold"> Joke API intergration with programming and dark humor, tow part (set up and delivery)</span>
34   <div class="joke p-4" id="jokeContainer">
35     ...
36   </div>
37 </div>
38 <script src="email.js"></script>
39 </div>
40   Using External JS file (email.js) to run function on this page.
41 </div>
42 <div id="show_hide_email" style="cursor: pointer; text-decoration: underline; color: blue;" onclick="show_or_hide_email()">
43   Click here to see my email/ hide email
44 </div>
45 <div class="new_func flex flex-col py-5 my-5">
46   <h1 class="text-x1 font-bold">ASCII Sum Calculator</h1>
47   <form class="flex flex-col md:flex-col my-5">
48     <input placeholder="Enter a string" class="border-1 border-black rounded-2xl p-2 my-2" type="text" id="inputString" name="inputString">
49     <button class="text-white my-1 bg-orange-500 p-1 hover:text-black hover:bg-white rounded-2xl" type="button" onclick="sumAsciivalue()>Calculate</button>
50   </form>
51   <div id="result"></div>
52 </div>
53 <!-- canvas id="analog_clock" -->
54 <div class="clocks flex flex-col items-center my-5 justify-between">
55   <h1 class="text-x2l font-bold">Digital Clock (24hr format)</h1>
56   <div class="my-3 py-3" id="digital-clock"></div>
57   <h1 class="text-x2l font-bold">Analog Clock</h1>
58   <canvas class="my-3 py-3" id="analog-clock" width="200" height="200"></canvas>
59 </div>
60 
61 
62 
63 
64 
65 
66 
67 </div>
68 </div>
69   ...
70 </div>
```

Figure 10: install emailcode

```
124
125     setInterval(updateDigitalClock, 1000);
126
127
128 });
129 function AnalogClock() {
130     var now = new Date();
131     var canvas = document.getElementById('analog-clock');
132     var context = canvas.getContext('2d');
133     var xCenter = canvas.width / 2;
134     var yCenter = canvas.height / 2;
135     var radius = Math.min(canvas.width / 2, 180);
136     context.clearRect(0, 0, canvas.width, canvas.height);
137     context.beginPath();
138     context.arc(xCenter, yCenter, radius, 0, 2 * Math.PI);
139     context.lineWidth = 10;
140     context.strokeStyle = "#E655100";
141     context.stroke();
142     var hour = now.getHours() % 12;
143     var minute = now.getMinutes();
144     var second = now.getSeconds();
145     var hourAngle = (hour + minute / 60) * 30 * Math.PI / 180;
146     var minuteAngle = (minute + second / 60) * 6 * Math.PI / 180;
147     var secondAngle = second * 6 * Math.PI / 180;
148
149     // Hour hand
150     context.beginPath();
151     context.moveTo(xCenter, yCenter);
152     context.lineTo(xCenter + Math.cos(hourAngle) * (radius * 0.5), yCenter + Math.sin(hourAngle) * (radius * 0.5));
153     context.lineWidth = 6;
154     context.strokeStyle = "#000000";
155     context.stroke();
156     // Minute hand
157     context.beginPath();
158     context.moveTo(xCenter, yCenter);
159     context.lineTo(xCenter + Math.cos(minuteAngle) * (radius * 0.7), yCenter + Math.sin(minuteAngle) * (radius * 0.7));
160     context.lineWidth = 4;
161     context.strokeStyle = "#6666";
162     context.stroke();
163 }
```

Figure 11: install Analogcode

```

91     }
92   }
93 });
94 });
95 // Make an AJAX GET request on page load
96 setInterval(function(){
97   $.get('https://v2.jokeapi.dev/joke/Programming,Dark,Spooky?blacklistFlags=religious,racist,sexist&type=twopart', function(result) {
98     // Handle the response
99     $('#jokeContainer').html('Set Up: ' + result.setup + " <br> Delivery : " + result.delivery);
100   });
101 }, 60000);
102 function sumAsciiValues() {
103   var inputString = document.getElementById("inputString").value;
104   var asciiSum = 0;
105   if (inputString == ''){
106     document.getElementById("result").innerHTML = "<i class='text-red'>Please enter a string</i>";
107   }else{
108     for (var i = 0; i < inputString.length; i++) {
109       asciiSum += inputString.charCodeAt(i);
110     }
111     document.getElementById("result").innerText = "Sum of ASCII values: " + asciiSum;
112     document.getElementById("inputString").value = '';
113   }
114 }
115 $(document).ready(function() {
116   // Digital Clock
117   function updateDigitalClock() {
118     var now = moment().format('HH:mm:ss');
119     $('#digital-clock').text(now);
120   }
121   setInterval(updateDigitalClock, 1000);
122 });
123
124 });
125
126 });
127

```

Figure 12: install Ascii-Digitalclock

Make sure to establish a link from the main file “index.html” to the course page “waph.html.” Additionally, adopt a distinctive approach to the Lab 2-related hands-on exercises within the “waph.html” file.

For the integration of two public Web APIs (20 pts), incorporate the jokeAPI (<https://v2.jokeapi.dev/joke/Any>) by selecting jokes from any category, similar to Lab 2.2.d.i. The goal is to display a new joke on your page every minute. In this instance, I opted for a two-part joke type with dark humor as the genre, excluding certain types of jokes. This choice stems from my previous use of an API in programming, presenting an opportunity to explore different parameters and enhance the variety of jokes displayed. HTML code

```

91     }
92   }
93 });
94 });
95 // Make an AJAX GET request on page load
96 setInterval(function(){
97   $.get('https://v2.jokeapi.dev/joke/Programming,Dark,Spooky?blacklistFlags=religious,racist,sexist&type=twopart', function(result) {
98     // Handle the response
99     $('#jokeContainer').html('Set Up: ' + result.setup + " <br> Delivery : " + result.delivery);
100   });
101 }, 60000);
102 function sumAsciiValues() {
103   var inputString = document.getElementById("inputString").value;
104   var asciiSum = 0;
105   if (inputString == ''){
106     document.getElementById("result").innerHTML = "<i class='text-red'>Please enter a string</i>";
107   }else{
108     for (var i = 0; i < inputString.length; i++) {
109       asciiSum += inputString.charCodeAt(i);
110     }
111     document.getElementById("result").innerText = "Sum of ASCII values: " + asciiSum;
112     document.getElementById("inputString").value = '';
113   }
114 }
115 $(document).ready(function() {
116   // Digital Clock
117   function updateDigitalClock() {
118     var now = moment().format('HH:mm:ss');
119     $('#digital-clock').text(now);
120   }
121   setInterval(updateDigitalClock, 1000);
122 });
123
124 });
125
126 };
127

```

Figure 13: install jokecode

Integrated a public API with graphics, utilizing the Weatherbit API for obtaining details on Cincinnati's location. Displayed the information in a graphical format, as depicted in the screenshot.

```
65
66
67 </div>
68 <script>
69   // Fetch Weather Data
70   fetch('https://api.weatherbit.io/v2.0/current?city=Cincinnati&key=77d23fc983748e6a7ead6e8ebf34531')
71     .then(response => response.json())
72     .then(data => {
73       const weatherData = {
74         labels: ['Temperature', 'Feels Like', 'Wind Speed'],
75         datasets: [
76           {
77             label: 'data',
78             data: [data.data[0].temp, data.data[0].app_temp, data.data[0].wind_spd],
79             backgroundColor: ['rgba(255, 99, 132, 0.2)', 'rgba(54, 162, 235, 0.2)', 'rgba(255, 206, 86, 0.2)'],
80             borderColor: ['rgba(255, 99, 132, 1)', 'rgba(54, 162, 235, 1)', 'rgba(255, 206, 86, 1)'],
81             borderWidth: 1
82           }
83         ];
84
85         const weatherChart = new Chart(document.getElementById('weather-chart'), {
86           type: 'bar',
87           data: weatherData,
88           options: {
89             scales: {
90               y: {
91                 beginAtZero: true
92               }
93             }
94           }
95         });
96       // Make an AJAX GET request on page load
97
98       setInterval(function(){
99         $.get('https://v2.jokeapi.dev/joke/Programming,Dark,Spooky?blacklistFlags=religious,racist,sexist&type=twopart', function(result) {
100           // Handle the response.
101           $('#jokeContainer').html(`Set Up: ${result.setup} <br> Delivery : ${result.delivery}`);
102         })
103       }, 5000);
104     }
105   })
106 </script>
```

Implementing a functionality worth 10 points, I've employed JavaScript cookies to remember the client. If it's the first visit, a message "Welcome to my homepage!" is displayed; otherwise, a message "Welcome back! Your last visit was on 'the date/time of the last visit'" is shown. To achieve this, I've organized the implementation using multiple functions for modularity. These functions handle setting cookies during the first visit, updating the cookie with the number of visits and the last visited time, and ultimately displaying a welcome message on the page.

```
Line wrap □
<!DOCTYPE html>
<html lang="en">
<head>
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-PB86QQ2Y2GH"></script>
<script>
window.dataLayer = window.dataLayer || [];
function gtag(x){dataLayer.push(arguments);}
gtag('js', new Date());
gtag('config', 'G-PB86QQ2Y2GH');
</script>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Likhitha Tirumalareddy Professsor</title>
<script src="https://cdn.tailwindcss.com"></script>
<script src="https://code.jquery.com/jquery-3.7.1.min.js" integrity="sha256-/JqT3s+2wUdR/l+jqzC2oHqEH/dl8V/ptdoJqS0f4c=" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css" integrity="sha512-DTQ9RNWCH3pp6qcwaEAIBIZO6xxalwEsw9c2Q0qAifT..."/>
<script>
    function setCookie(name, value, days) {
        var expires = "";
        if (days) {
            var date = new Date();
            date.setTime(date.getTime() + (days * 24 * 60 * 60 * 1000));
            expires = "expires=" + date.toUTCString();
        }
        document.cookie = name + "=" + value + expires + "; path=/";
    }
    function getCookie(name) {
        var cookieName = name + "=";
        var ca = document.cookie.split(';');
        for (var i = 0; i < ca.length; i++) {
            var c = ca[i];
            var cstr = c.substring(0, c.indexOf('='));
            if (cstr == cookieName) {
                return c.substring(c.indexOf('=') + 1, c.length);
            }
        }
        return null;
    }
</script>
```

Figure 14: install cookiecode

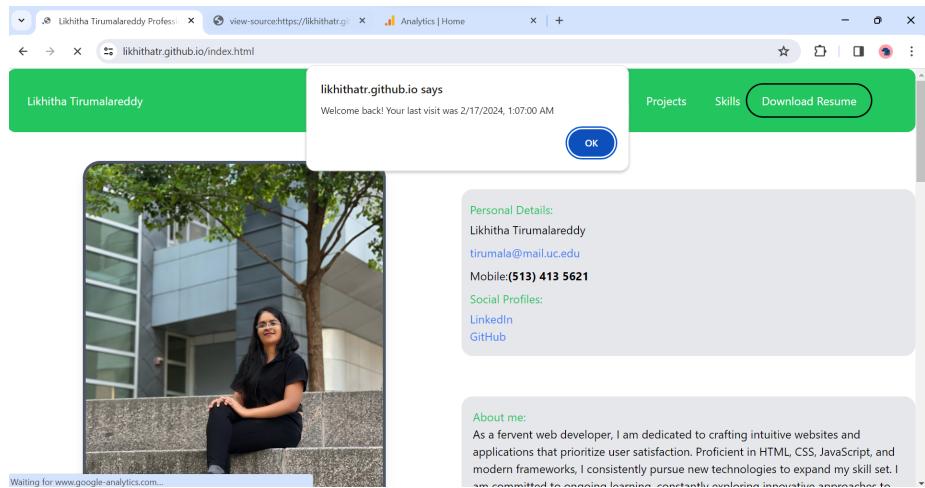


Figure 15: install SixScreen

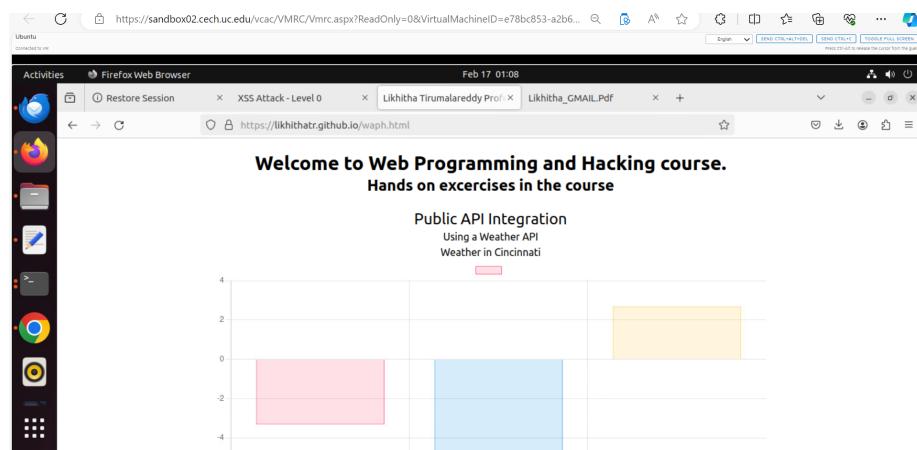


Figure 16: install Figure1

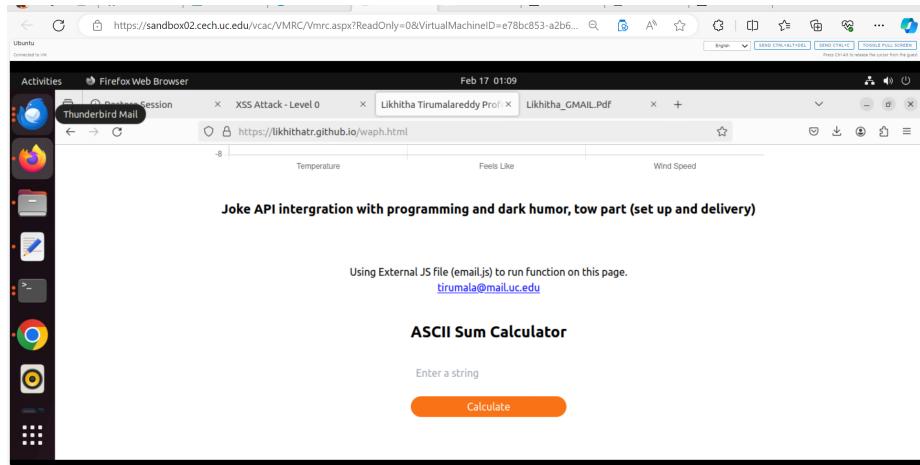


Figure 17: install Figure2

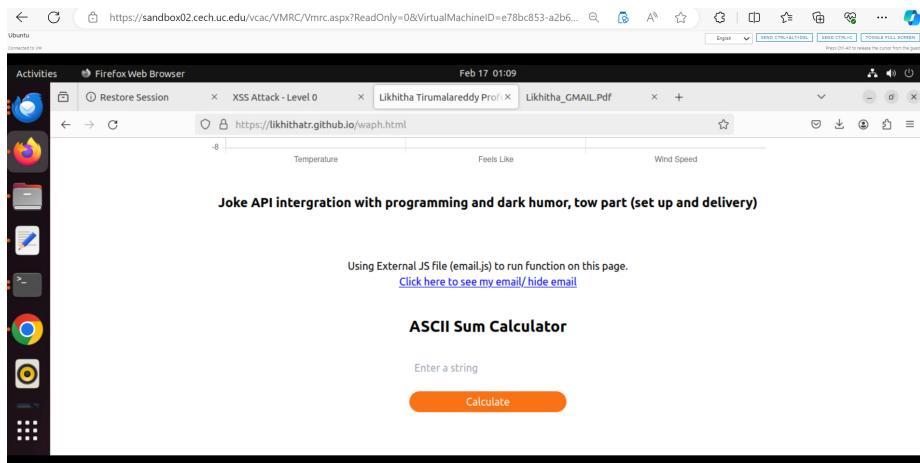


Figure 18: install Figure3

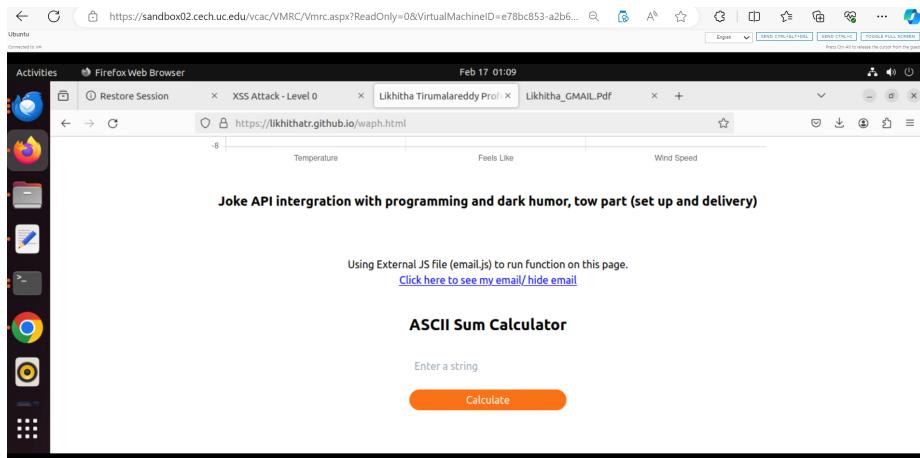


Figure 19: install Figure4

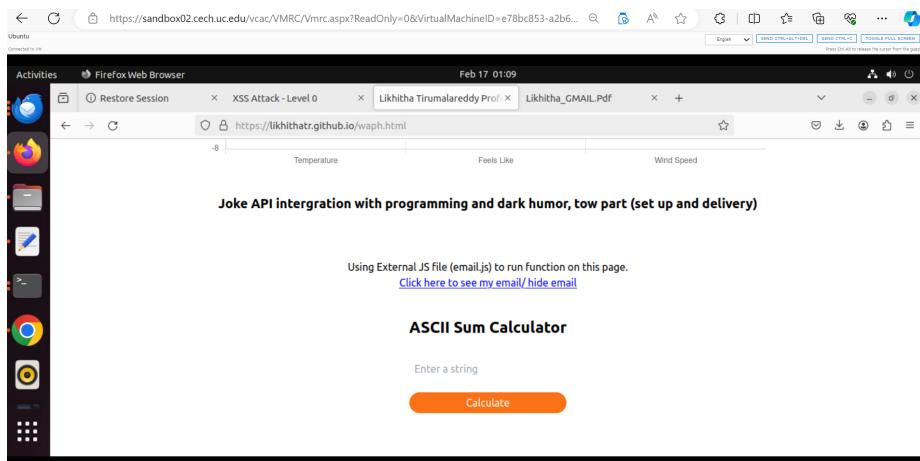


Figure 20: install Figure5

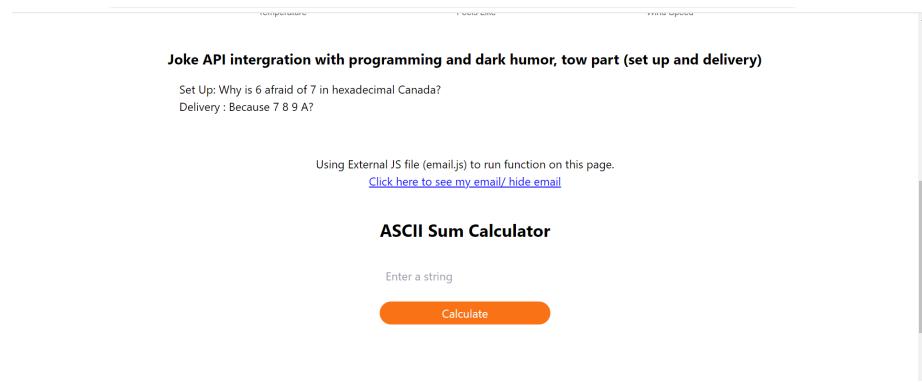


Figure 21: install joke2