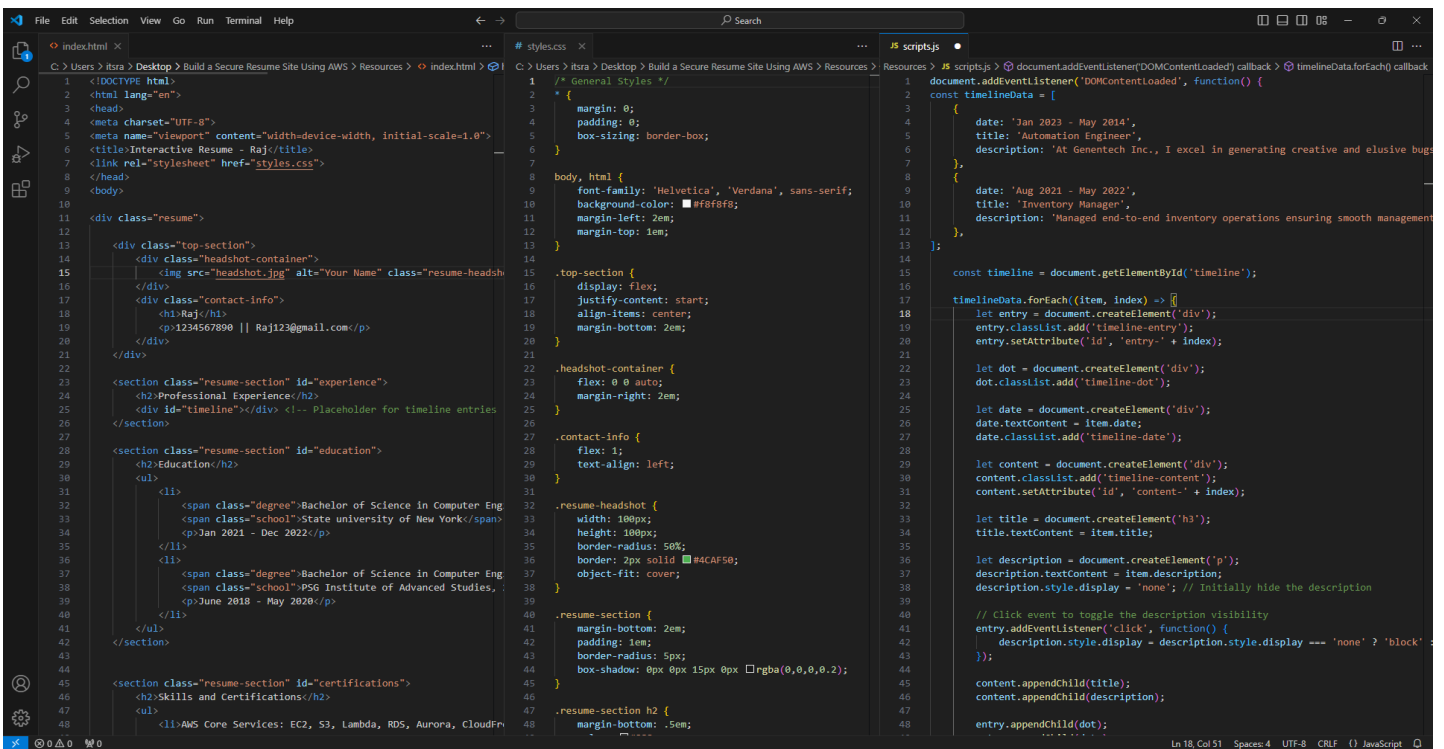


# AWS Project

## Build a Secure Resume Site Using AWS

### Steps to Implement:-

1. First create the code (HTML, CSS, JavaScript) for Your Resume.  
➔ I used Chat-GPT to generate the code and I modified the content.



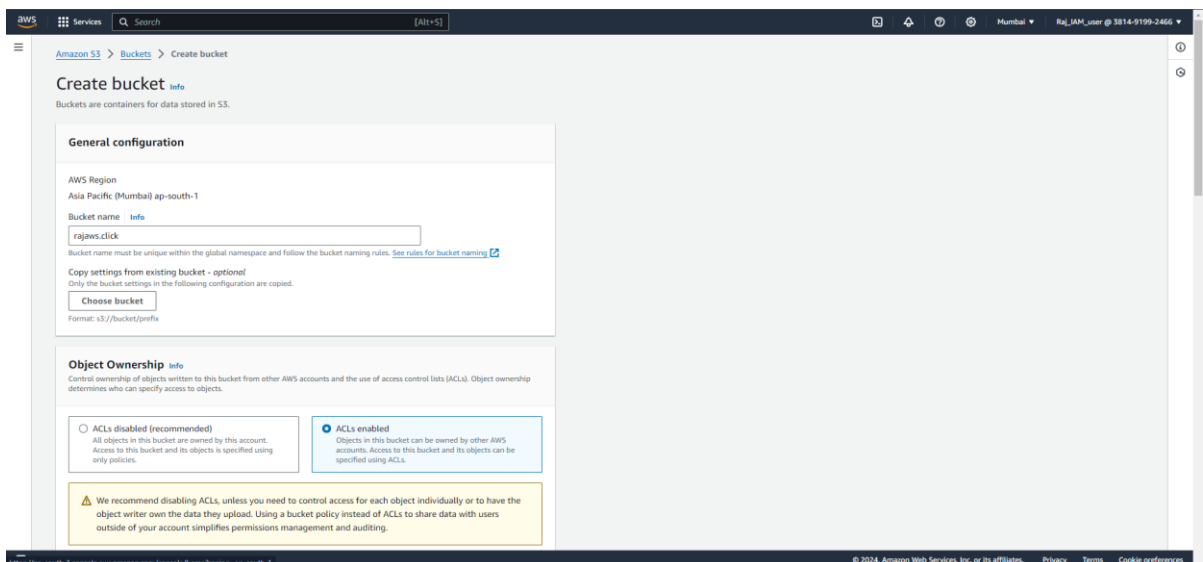
The screenshot displays a code editor with three files: index.html, styles.css, and scripts.js. The index.html file contains a basic HTML structure with a resume section, experience section, education section, and certifications section. The styles.css file defines the layout and styling for the resume, including a top section, headshot container, contact info, and various sections. The scripts.js file contains JavaScript code for handling the timeline data, creating elements, and toggling the description visibility.

```
index.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Interactive Resume - Raj</title>
7 <link rel="stylesheet" href="styles.css">
8 </head>
9 <body>
10
11 <div class="resume">
12
13 <div class="top-section">
14 <div class="headshot-container">
15 
16 </div>
17 <div class="contact-info">
18 <div>Raj</div>
19 <div>+91234567890 || Raj123@gmail.com</div>
20 </div>
21 </div>
22
23 <section class="resume-section" id="experience">
24 <h2>Professional Experience</h2>
25 <div id="timeline"></div> <!-- Placeholder for timeline entries -->
26 </section>
27
28 <section class="resume-section" id="education">
29 <h2>Education</h2>
30 <ul>
31 <li>
32 <span class="degree">Bachelor of Science in Computer Eng</span>
33 <span class="school">State university of New York</span>
34 <p>Jan 2021 - Dec 2022</p>
35 </li>
36 <li>
37 <span class="degree">Bachelor of Science in Computer Eng</span>
38 <span class="school">PSG Institute of Advanced Studies</span>
39 <p>June 2018 - May 2020</p>
40 </li>
41 </ul>
42 </section>
43
44 <section class="resume-section" id="certifications">
45 <h2>Skills and Certifications</h2>
46 <ul>
47 <li>AWS Core Services: EC2, S3, Lambda, RDS, Aurora, CloudF</li>
48 </ul>
49 </section>
50 </div>
51 </body>
52 </html>
```

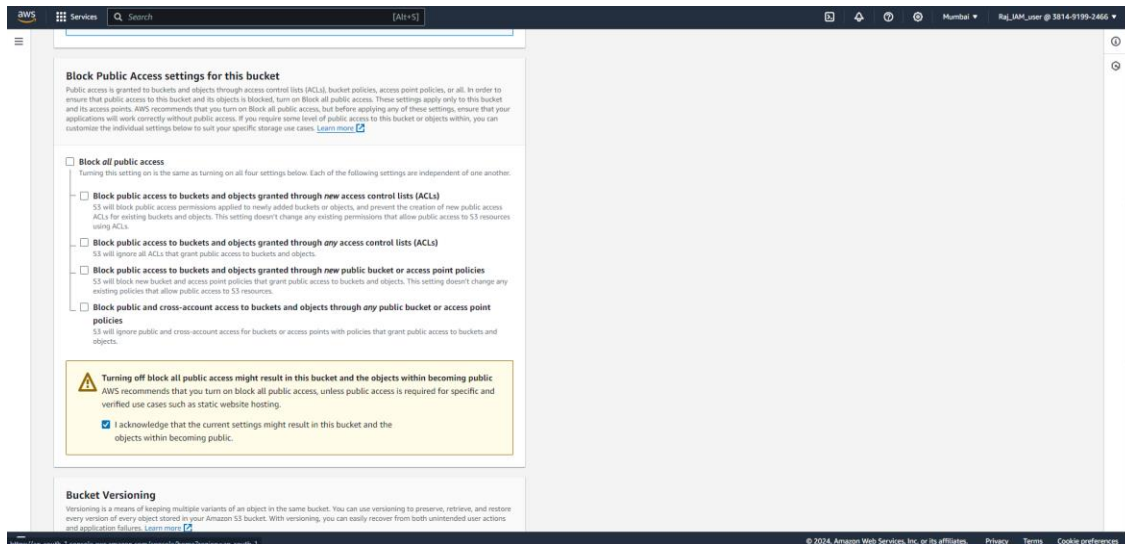
```
styles.css
1 /* General Styles */
2 * {
3   margin: 0;
4   padding: 0;
5   box-sizing: border-box;
6 }
7
8 body, html {
9   font-family: 'Helvetica', 'Verdana', sans-serif;
10  background-color: #f8f8f8;
11  margin-left: 2em;
12  margin-top: 1em;
13 }
14
15 .top-section {
16   display: flex;
17   justify-content: start;
18   align-items: center;
19   margin-bottom: 2em;
20 }
21
22 .headshot-container {
23   flex: 0 0 auto;
24   margin-right: 2em;
25 }
26
27 .contact-info {
28   flex: 1;
29   text-align: left;
30 }
31
32 .resume-headshot {
33   width: 100px;
34   height: 100px;
35   border-radius: 50%;
36   border: 2px solid #4CAF50;
37   object-fit: cover;
38 }
39
40 .resume-section {
41   margin-bottom: 2em;
42   padding: 1em;
43   border-radius: 5px;
44   box-shadow: 0px 0px 15px 0px rgba(0,0,0,0.2);
45 }
46
47 .resume-section h2 {
48   margin-bottom: .5em;
49 }
```

```
scripts.js
1 document.addEventListener('DOMContentLoaded', function() {
2   const timelineData = [
3     {
4       date: 'Jan 2023 - May 2024',
5       title: 'Automation Engineer',
6       description: 'At Genentech Inc., I excel in generating creative and elusive bug'
7     },
8     {
9       date: 'Aug 2021 - May 2022',
10      title: 'Inventory Manager',
11      description: 'Managed end-to-end inventory operations ensuring smooth management'
12    }
13  ];
14
15  const timeline = document.getElementById('timeline');
16
17  timelineData.forEach((item, index) => {
18    let entry = document.createElement('div');
19    entry.classList.add('timeline-entry');
20    entry.setAttribute('id', 'entry-' + index);
21
22    let dot = document.createElement('div');
23    dot.classList.add('timeline-dot');
24
25    let date = document.createElement('div');
26    date.textContent = item.date;
27    date.classList.add('timeline-date');
28
29    let content = document.createElement('div');
30    content.classList.add('timeline-content');
31    content.setAttribute('id', 'content-' + index);
32
33    let title = document.createElement('h3');
34    title.textContent = item.title;
35
36    let description = document.createElement('p');
37    description.textContent = item.description;
38    description.style.display = 'none'; // Initially hide the description
39
40    // Click event to toggle the description visibility
41    entry.addEventListener('click', function() {
42      description.style.display = description.style.display === 'none' ? 'block' : 'none';
43    });
44
45    content.appendChild(title);
46    content.appendChild(description);
47
48    entry.appendChild(dot);
49  });
50
51  timeline.appendChild(entry);
52
53  // Append the timeline entries to the timeline container
54  timeline.appendChild(timeline);
55
56  // Append the timeline entries to the timeline container
57  timeline.appendChild(timeline);
58
59  // Append the timeline entries to the timeline container
60  timeline.appendChild(timeline);
61
62  // Append the timeline entries to the timeline container
63  timeline.appendChild(timeline);
64
65  // Append the timeline entries to the timeline container
66  timeline.appendChild(timeline);
67
68  // Append the timeline entries to the timeline container
69  timeline.appendChild(timeline);
70
71  // Append the timeline entries to the timeline container
72  timeline.appendChild(timeline);
73
74  // Append the timeline entries to the timeline container
75  timeline.appendChild(timeline);
76
77  // Append the timeline entries to the timeline container
78  timeline.appendChild(timeline);
79
80  // Append the timeline entries to the timeline container
81  timeline.appendChild(timeline);
82
83  // Append the timeline entries to the timeline container
84  timeline.appendChild(timeline);
85
86  // Append the timeline entries to the timeline container
87  timeline.appendChild(timeline);
88
89  // Append the timeline entries to the timeline container
90  timeline.appendChild(timeline);
91
92  // Append the timeline entries to the timeline container
93  timeline.appendChild(timeline);
94
95  // Append the timeline entries to the timeline container
96  timeline.appendChild(timeline);
97
98  // Append the timeline entries to the timeline container
99  timeline.appendChild(timeline);
100 }
```

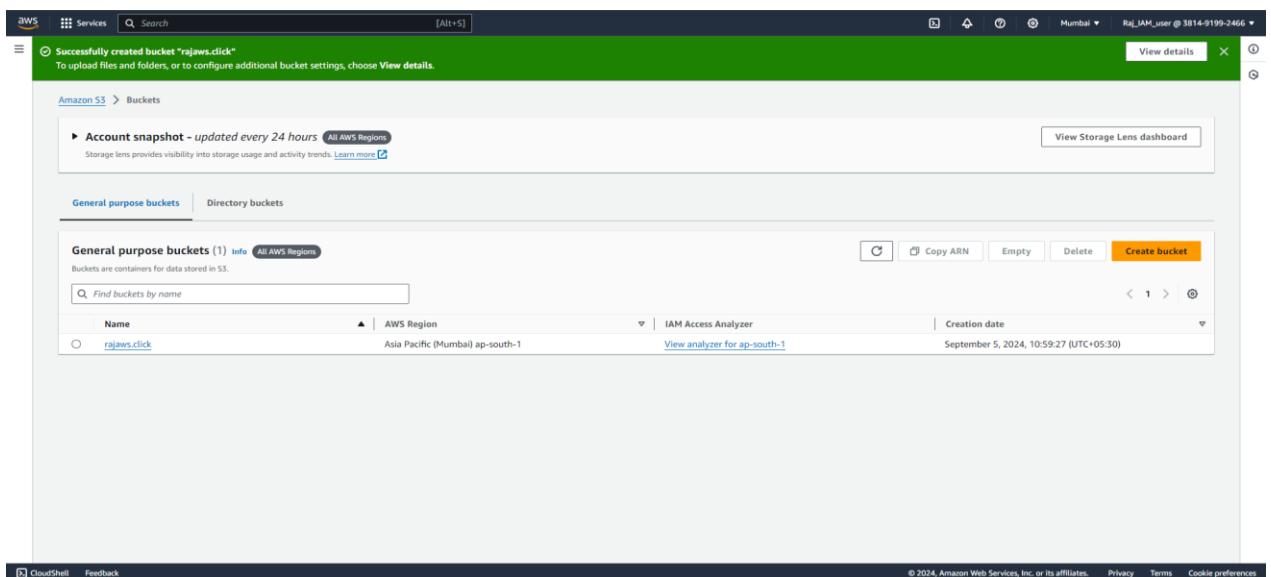
2. Create an S3 bucket.



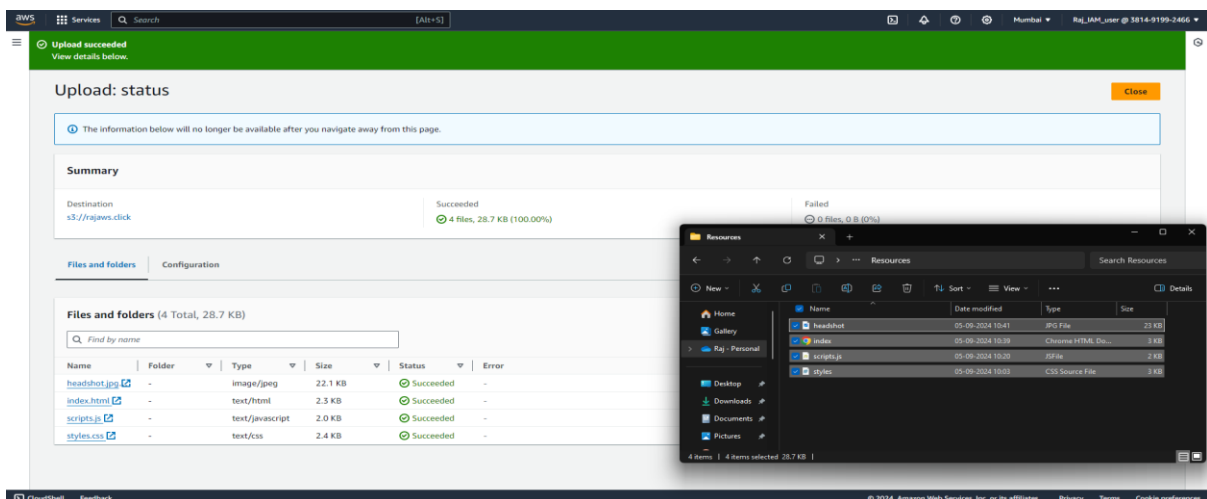
### 3. Deselect the setting for Block all public access.



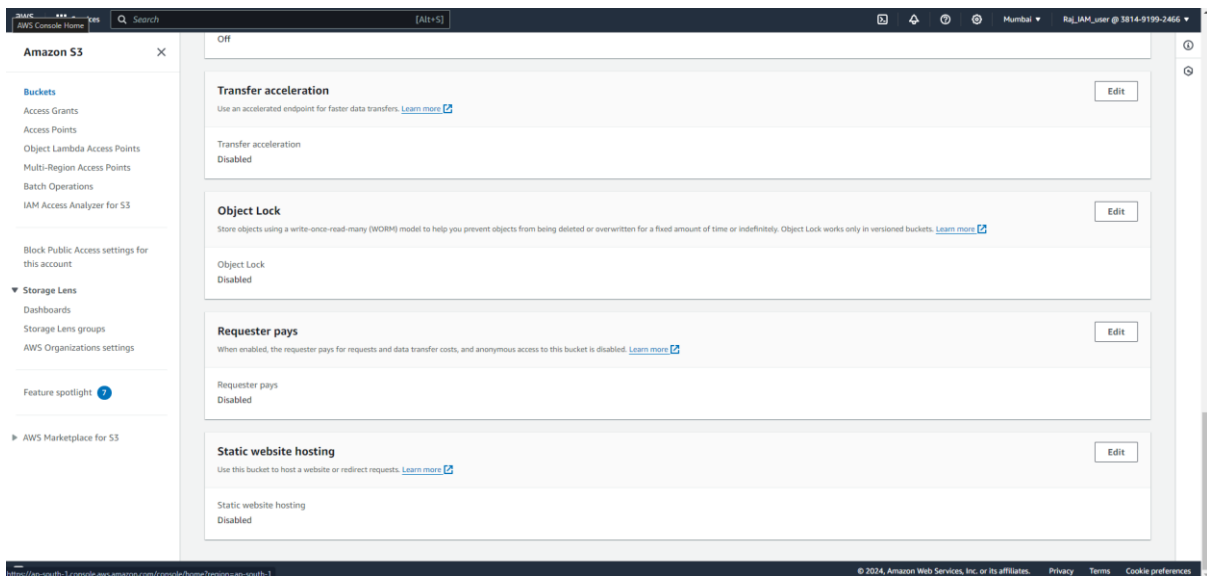
### 4. S3 bucket was created



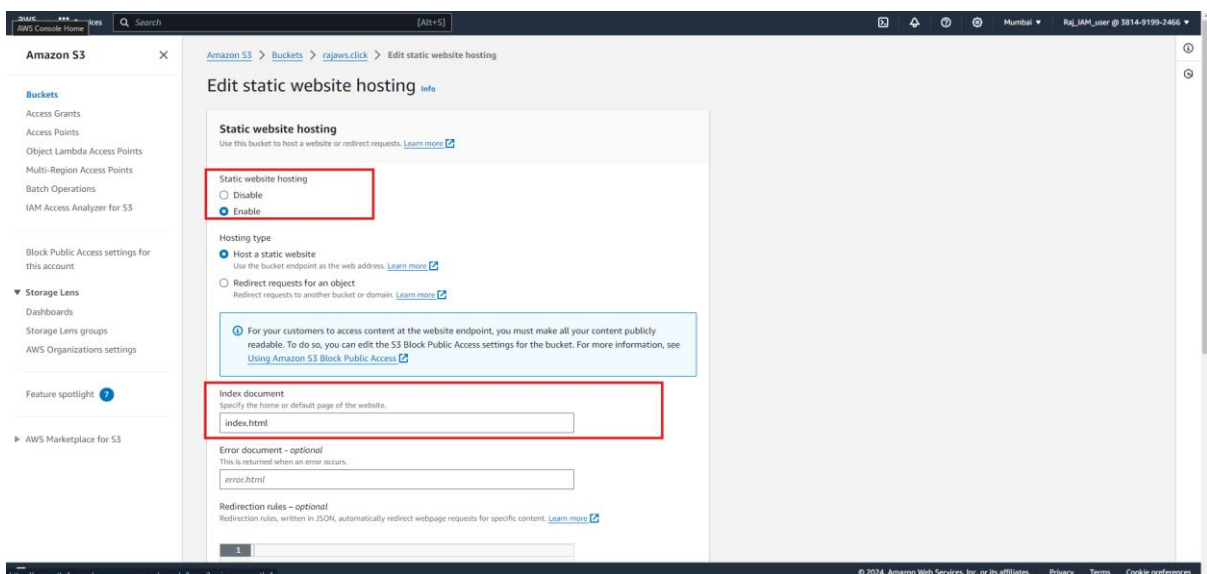
### 5. Uploading my resources into my S3 bucket



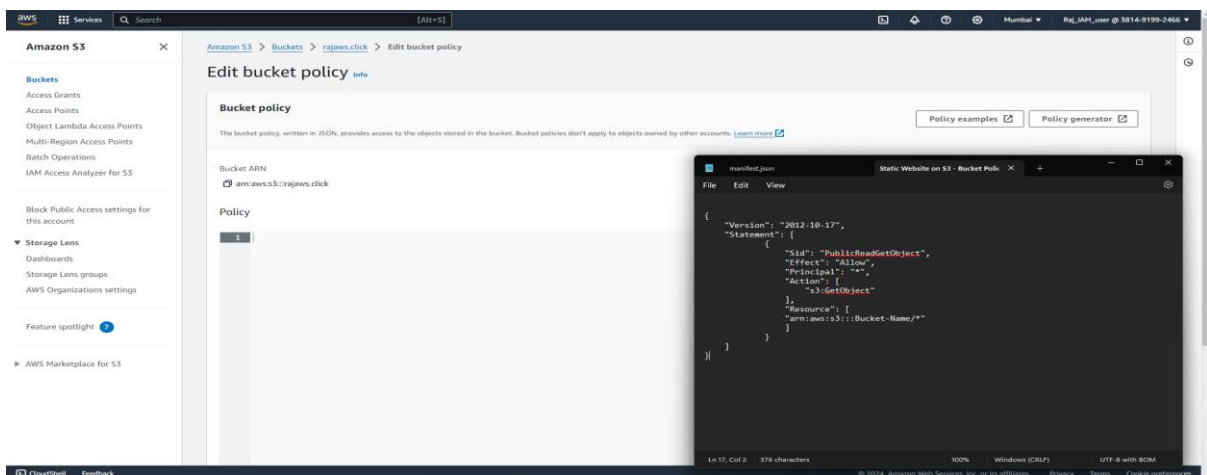
## 6. Enable Static Website Hosting in the properties section.



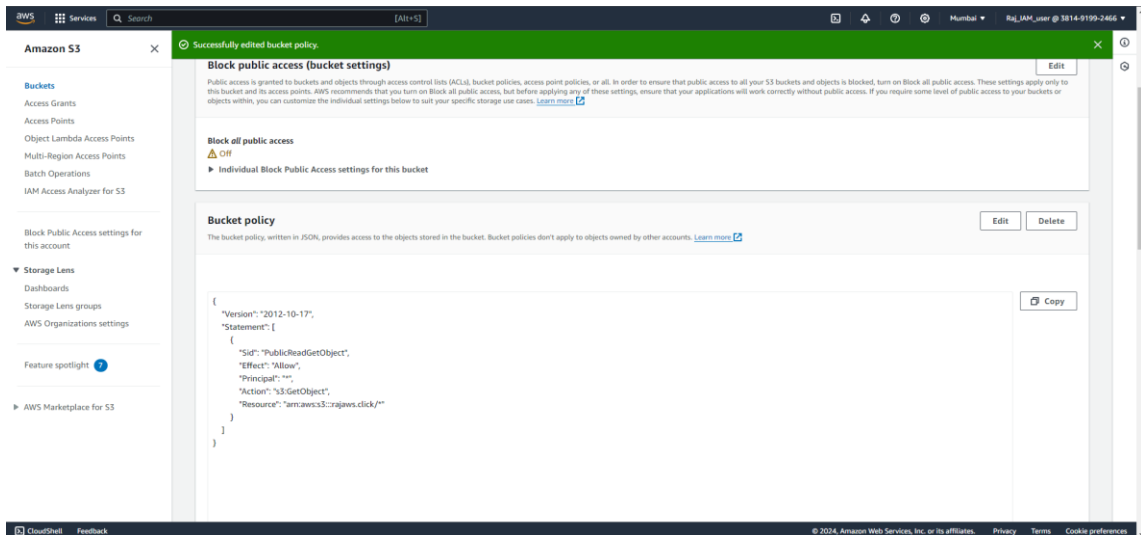
## 7. For the Index document, enter index.html. This specifies the default home page for the site (your HTML code for your resume). Then click Save changes.



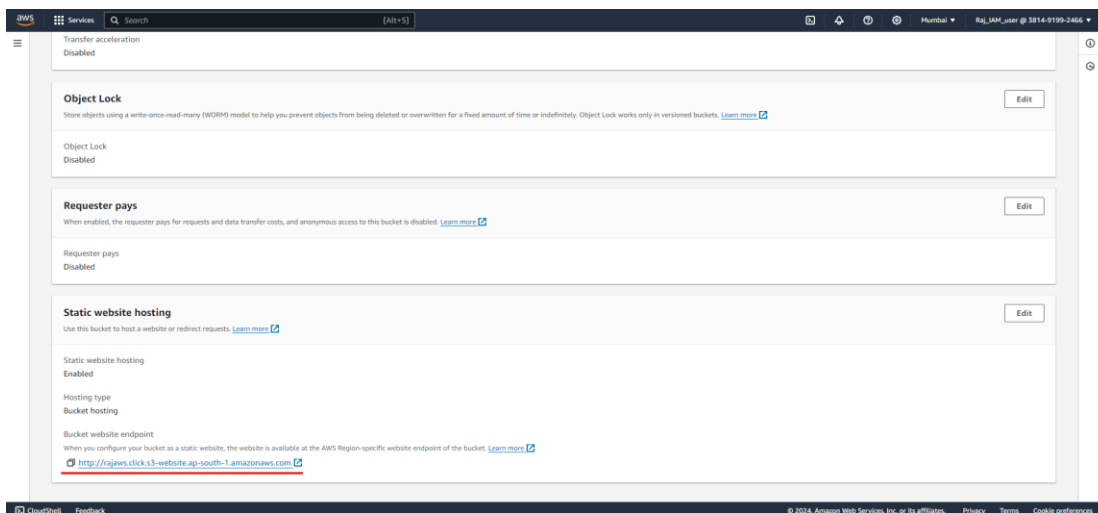
## 8. Copy the S3 bucket policy and paste it into the policy editor.



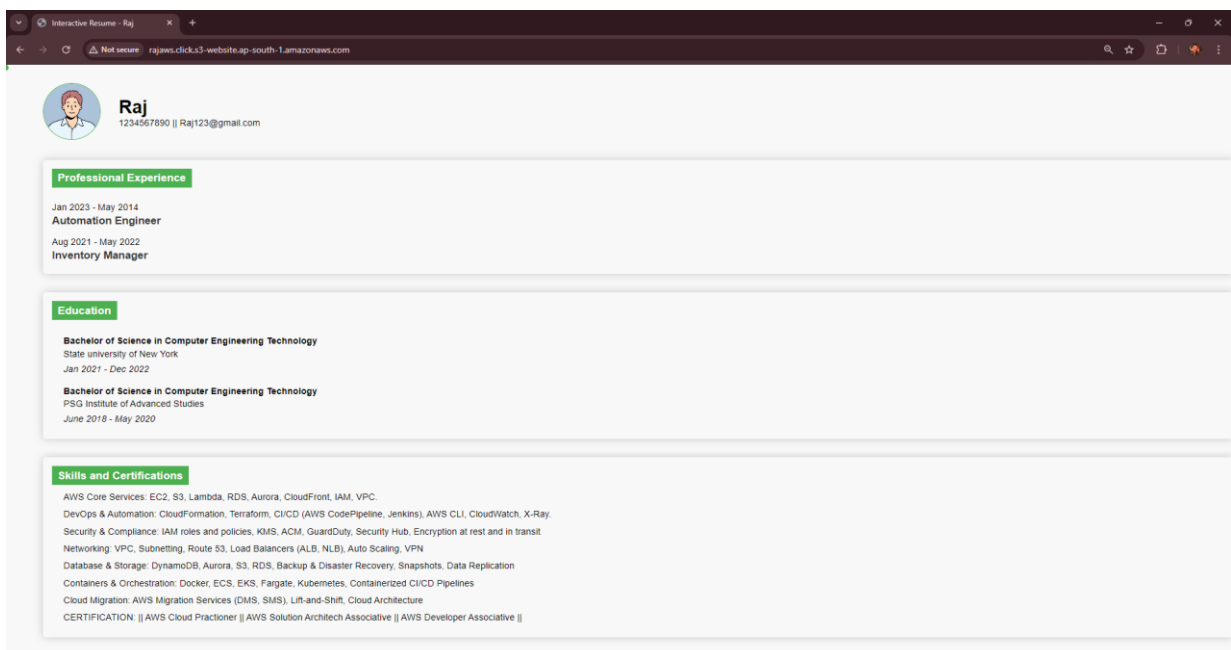
## 9. Bucket policy was added with correct bucket ARN.



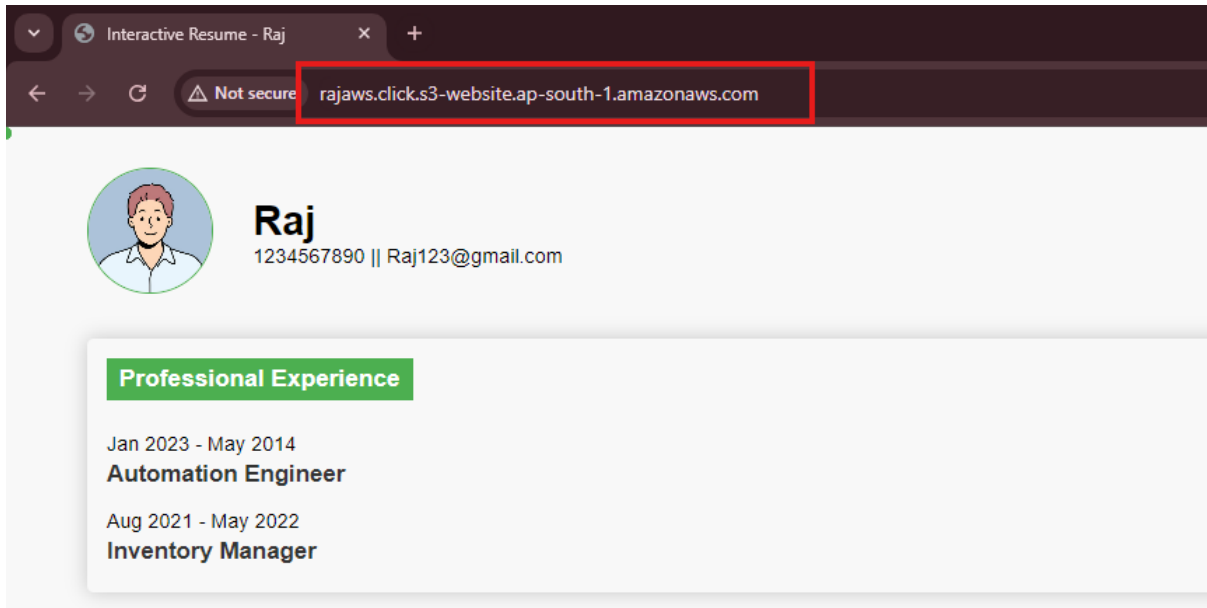
## 10. Click the bucket website endpoint to view your resume.



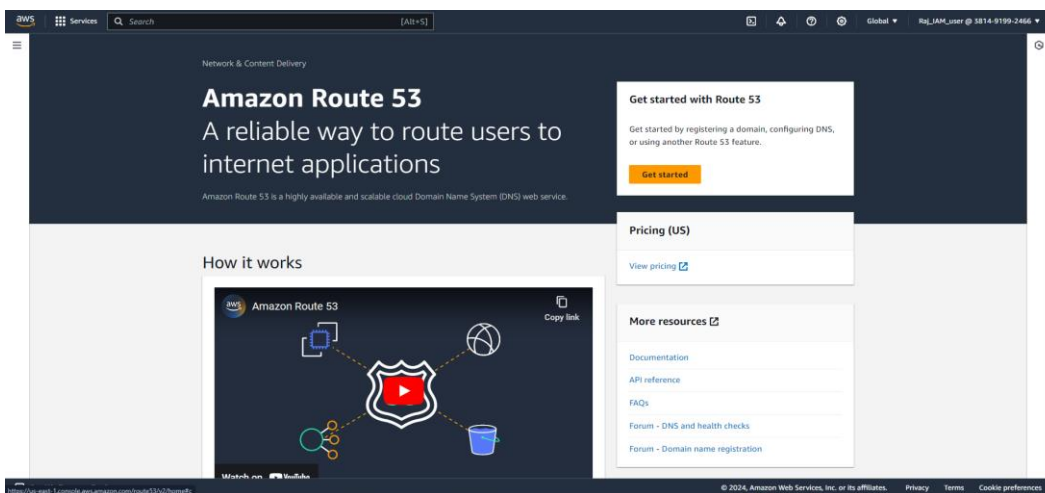
## 11. It works! Resume being served up by S3.



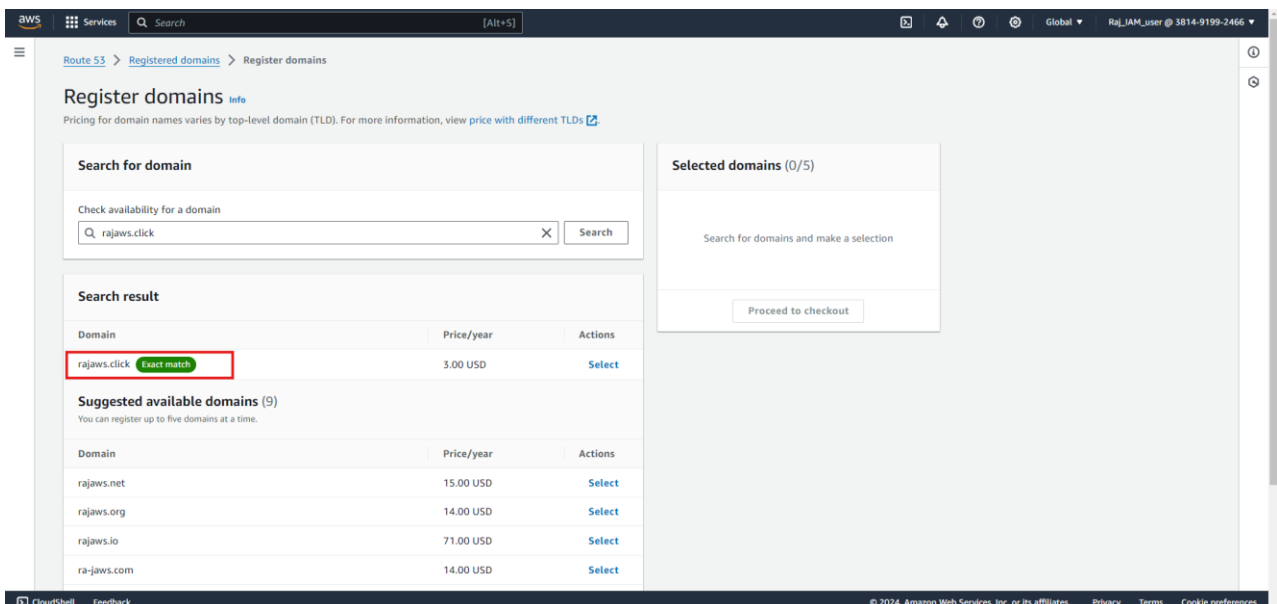
12. The resume would be much more impressive with a custom domain name of its own.



13. Now I'm Register a New Domain Name with Route 53



14. I got the exact match of the domain name that I have mentioned.



15. Now I'm purchasing a domain by clicking "proceed to checkout"

The screenshot shows the AWS 'Register domains' page. On the left, under 'Search for domain', the text 'rajaws.click' is entered in the search box. Below this, the 'Search result' table shows 'rajaws.click' with an 'Exact match' tag, a price of '3.00 USD', and an 'Actions' column with a 'Selected' button. On the right, the 'Selected domains (1/5)' section shows 'rajaws.click' with a '3.00 USD' price and a 'Remove' button. Below this, the 'Subtotal' is '3.00 USD'. A note states: 'The domain registration fee displayed is for 1 year. You can change duration on the next page.' At the bottom right, there is a prominent orange 'Proceed to checkout' button.

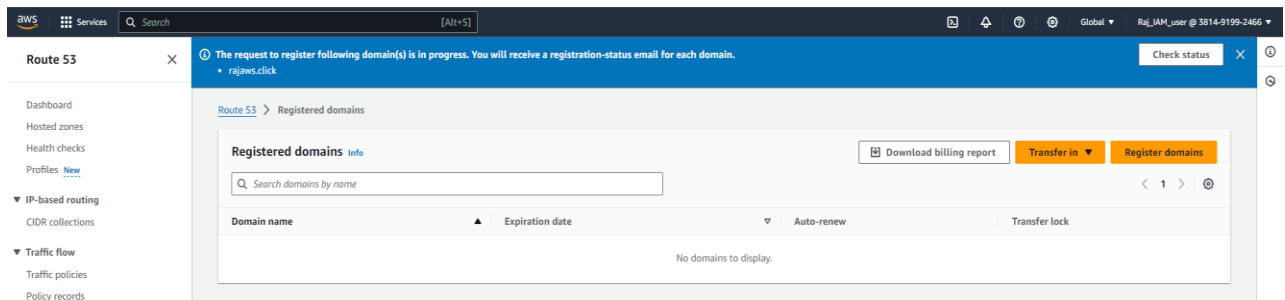
16. I have untick checkbox because I don't want to do auto-renew

The screenshot shows the AWS 'Pricing' page for domain registration. The 'Domain pricing options' section shows 'rajaws.click' as the domain name and '1 year (3.00 USD)' as the duration. The 'Auto-renew' checkbox is set to 'Off'. A blue information box states: 'Auto-renew is turned off for this domain and the registration will expire after the selected registration period. For more information, see [Renewing Registration for a Domain](#).' The 'Subtotal' is '3.00 USD'. At the bottom right, there are 'Cancel' and 'Next' buttons.

17. Filling up the contact information

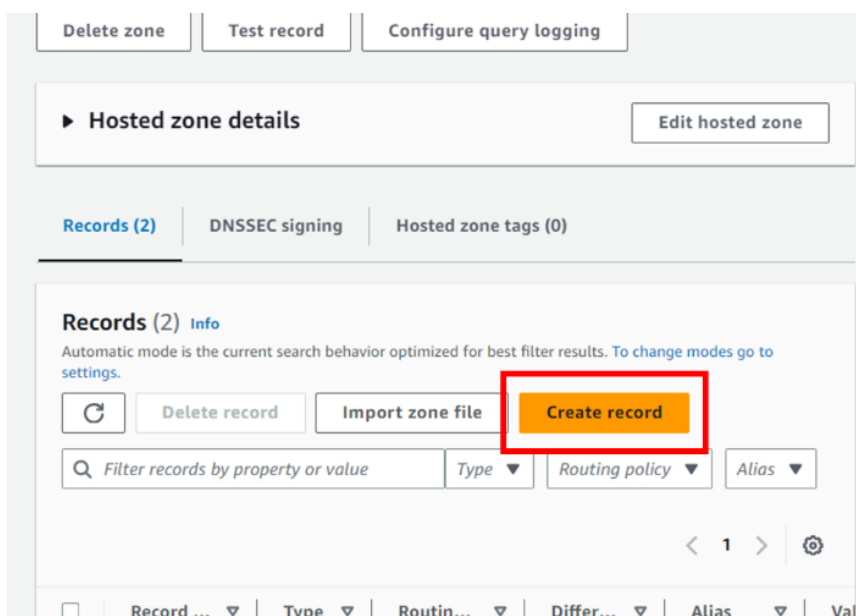
The screenshot shows the AWS 'Contact information' page. It is divided into two main sections: 'Registrant contact' and 'Admin contact'. The 'Registrant contact' section includes fields for 'General information' (Contact type, Organization, First name, Last name, Email) and 'Address information' (Address 1, Address 2 - optional, Country, State / Province, City, Zip code / Postal code). The 'Admin contact' section has a checkbox labeled 'Same as the registrant contact'. The page is part of a multi-step process, with 'Pricing' as Step 1 and 'Contact information' as Step 2.

18. I have registered the domain.

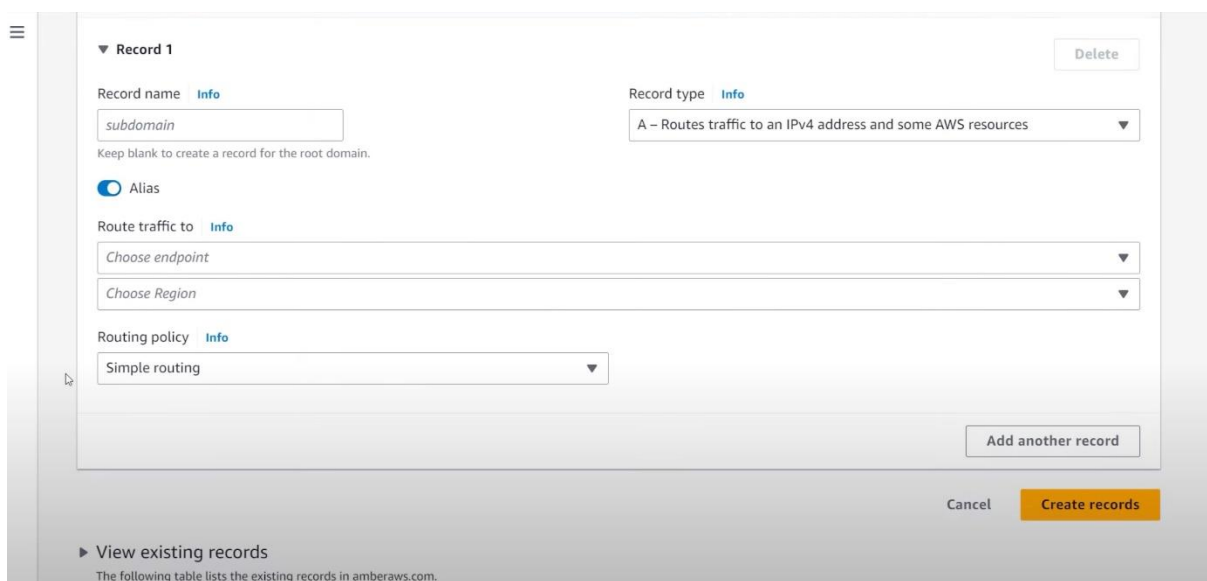


19. Create an A Record with an Alias to Point to the S3 Website.

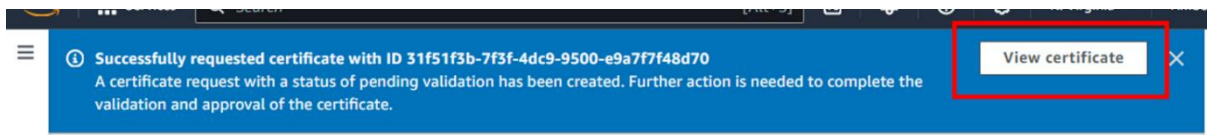
➔ Click into your hosted zone, and then click Create record.



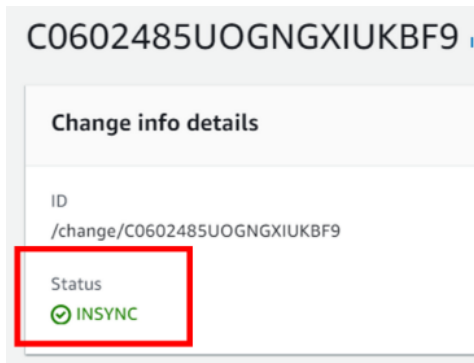
20. Now fill in the details of where to route traffic.



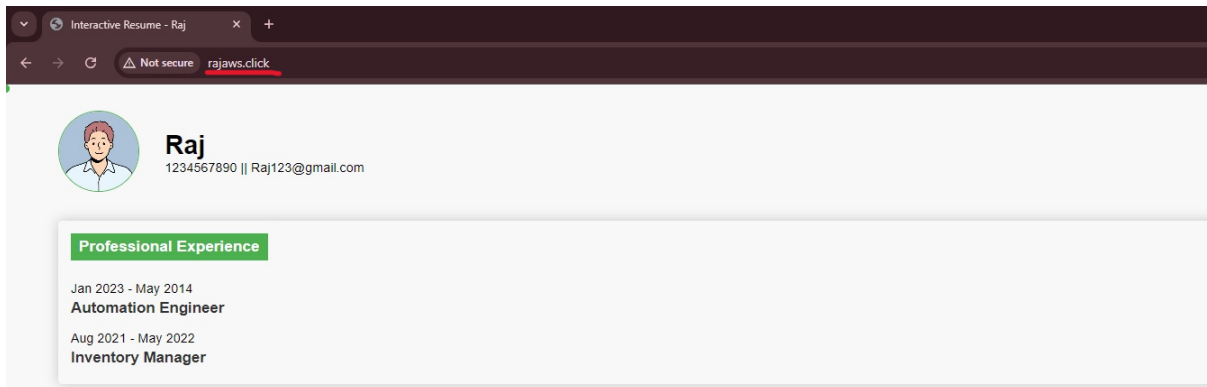
21. You can view the status of propagation by clicking on the handy View status.



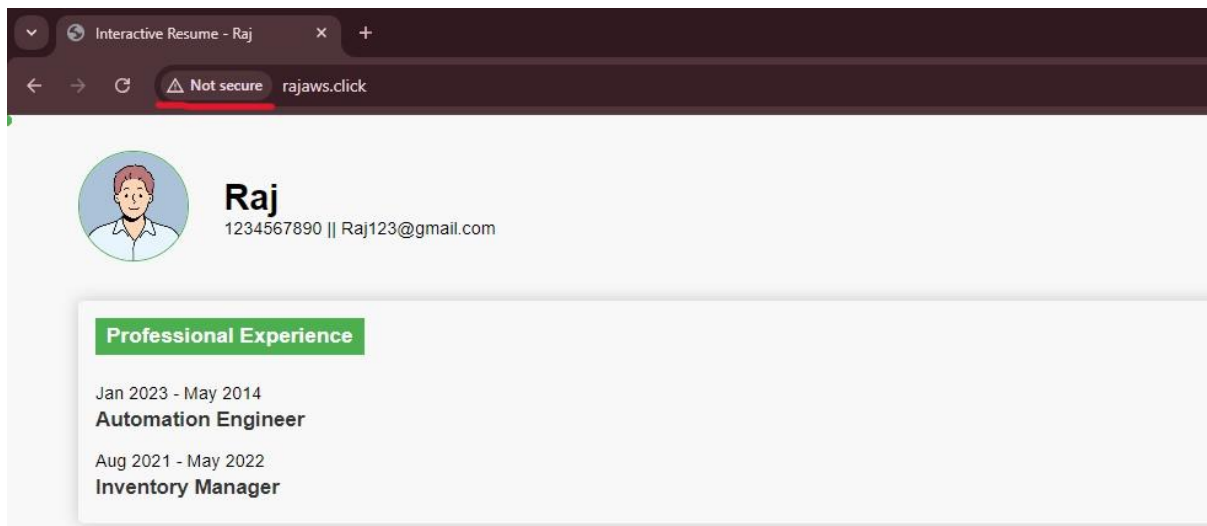
22. After the Status changes from PENDING to INSYNC, then you should be good to test out your changes.



23. My domain name is now displaying with my resume.

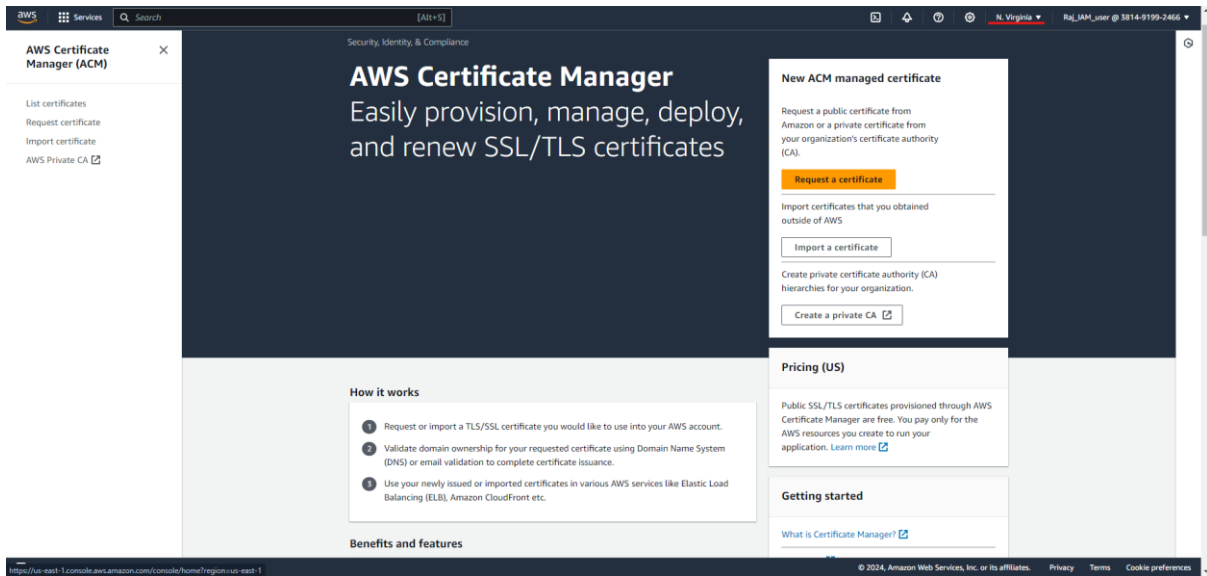


24. But this not a "Not secure" domain.

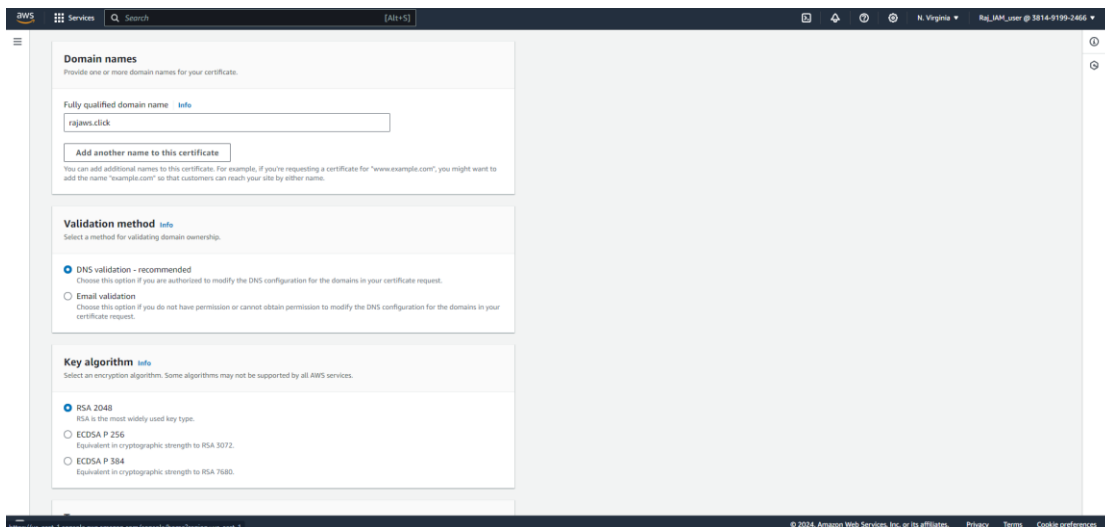




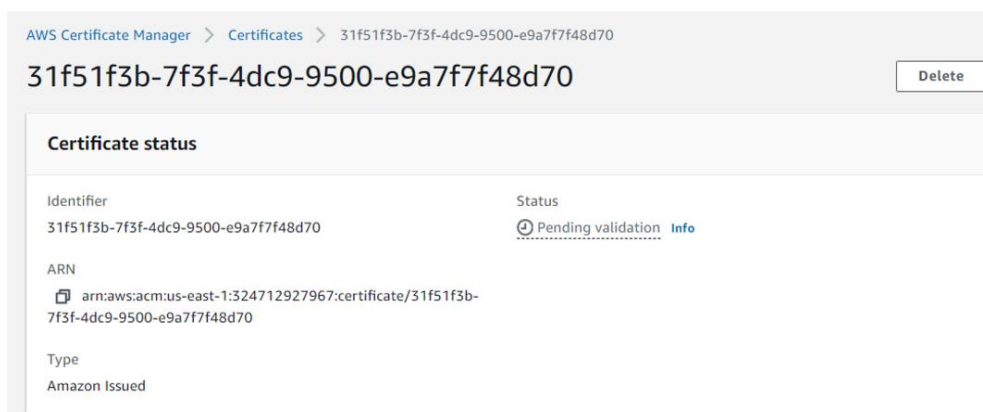
25. So, now I'm Creating a Public TLS/SSL Certificate using AWS Certificate Manager.
- If you create a certificate in another region, you won't be able to use it with CloudFront so use (N. Virginia) region.



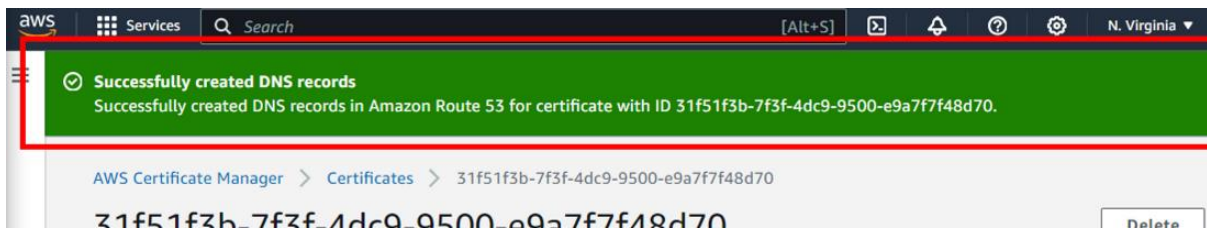
26. I'm Entering my domain name and leave the rest of the options as defaults, then click Request.



27. The request was successful, but it's in a "pending validation" status

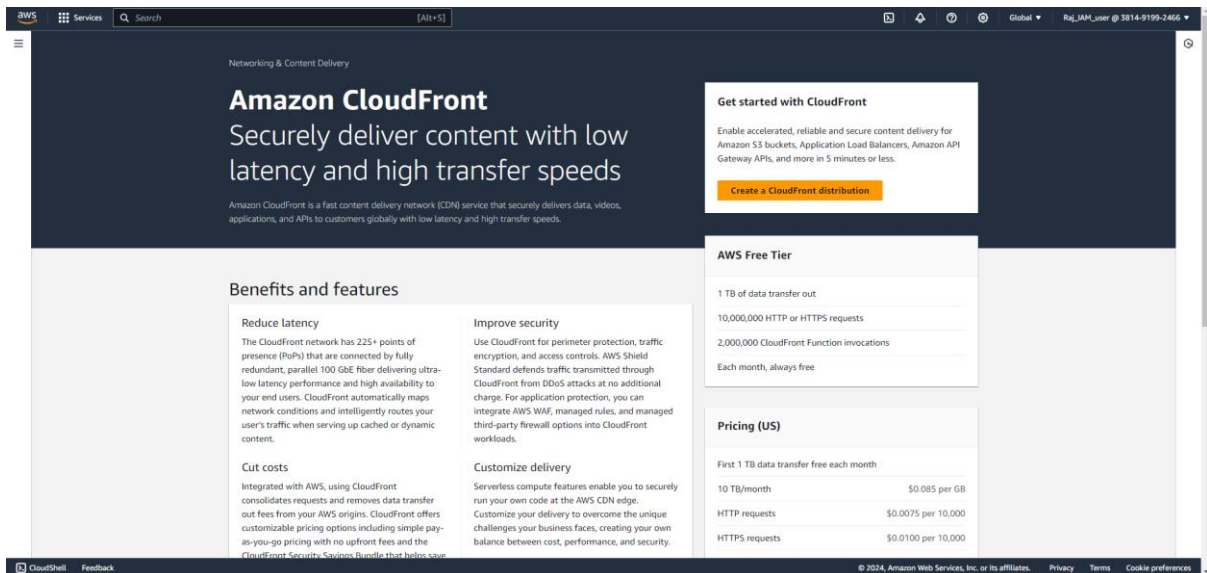


28. Now I'm Creating a records in Route 53 to validate DNS  
→ Successfully created a record in Route 53 to validate DNS

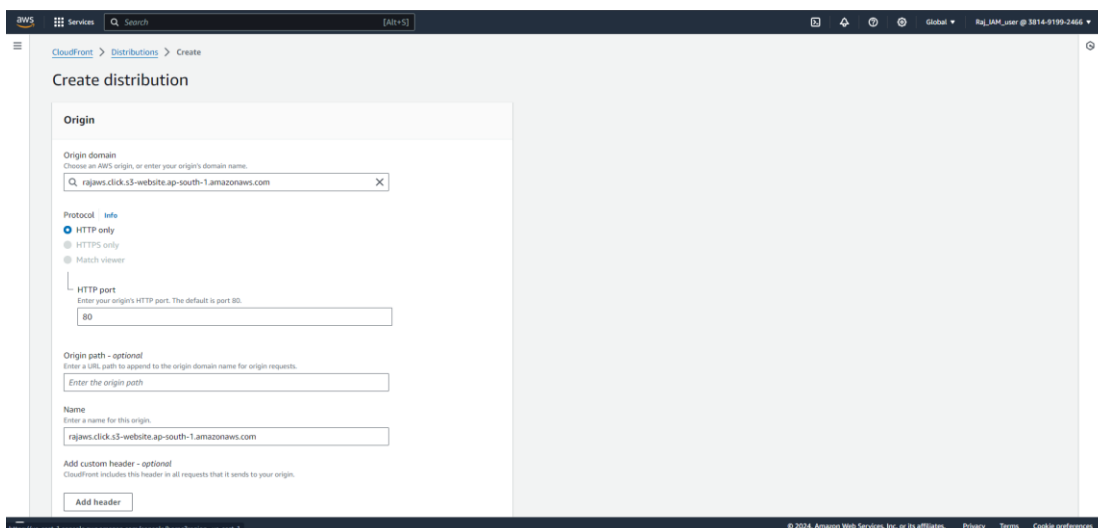


**NOTE:** My website files are currently hosted in S3, but unfortunately, you can't use a certificate on an S3 bucket. So I'm using CloudFront distribution that points to the S3 bucket. And then the certificate is applied to the CloudFront distribution.

29. Creating a CloudFront Distribution.



30. The origin domain is where your website files live, which is in S3. If you type in S3 to filter, it should pull up your bucket.



31. Scroll down to the Default cache behavior section, then under Viewer, select Redirect HTTP to HTTPS.

Compress objects automatically [Info](#)

☐ No

☒ Yes

**Viewer**

Viewer protocol policy

☐ HTTP and HTTPS

☒ Redirect HTTP to HTTPS

☐ HTTPS only

Allowed HTTP methods

☒ GET, HEAD

☐ GET, HEAD, OPTIONS

☐ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

Restrict viewer access

If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.

☒ No

☐ Yes

32. For Alternate domain name (CNAME), entering my domain name and using Custom SSL certificate.

Alternate domain name (CNAME) - *optional*

Add the custom domain names that you use in URLs for the files served by this distribution.

rajaws.click [Remove](#)

[Add item](#)

[To add a list of alternative domain names, use the bulk editor.](#)

Custom SSL certificate - *optional*

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

Choose certificate [Request certificate](#)

Supported HTTP versions

Add support for additional HTTP versions. HTTP/1.0 and HTTP/1.1 are supported by default.

☒ HTTP/2

☐ HTTP/3

33. Successfully created CloudFront Distribution.

Successfully created new distribution.

[CloudFront](#) > [Distributions](#) > E7ERFVR8EXXXJ

**E7ERFVR8EXXXJ**

[General](#) | [Origins](#) | [Behaviors](#) | [Error pages](#) | [Geographic restrictions](#) | [Inv.](#)

**Details**

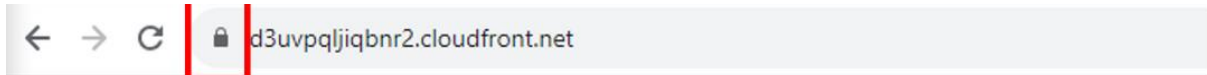
Distribution domain name

[d3uypqljqbnr2.cloudfront.net](#)

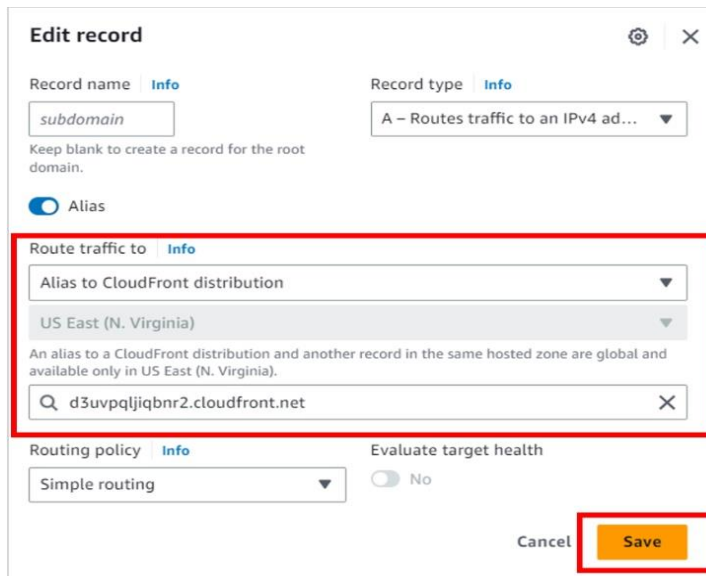
ARN

[arn:aws:cloudfront::324712927967:distribution/E7ERFVR8EXXXJ](#)

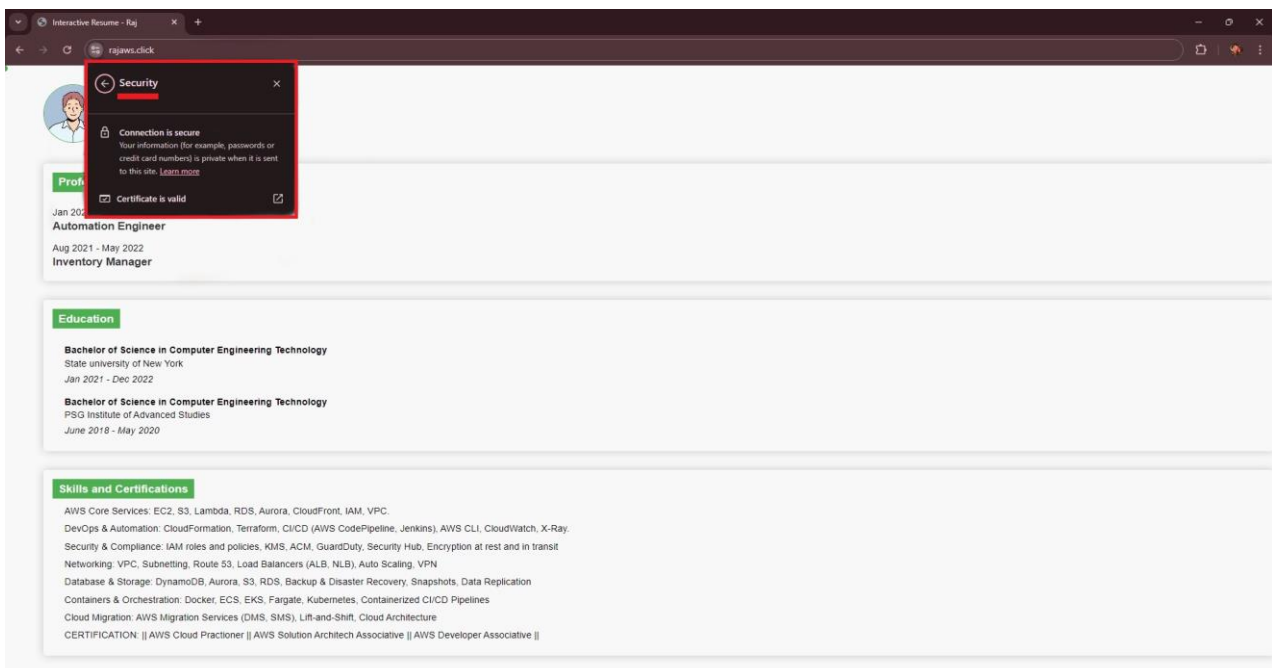
34. The TLS/SSL certificate is working with CloudFront.



35. Now updating Route 53 to Point to the CloudFront Distribution.



36. I'm able to navigate to my custom domain name and have it load my resume on a secure connection.



*The resume files, served from S3 through CloudFront, load on a custom domain configured in Route 53, secured with a TLS/SSL certificate provided by Certificate Manager.*