**Serverless Reminder App**

**Introduction:**

In this, we will be implementing a serverless reminder application. The Web application will load from an S3 bucket as a static website and run in browser, communicate with Lambda and Step functions via an API Gateway Endpoint. Using the application you will be able to configure reminders to be send using email.

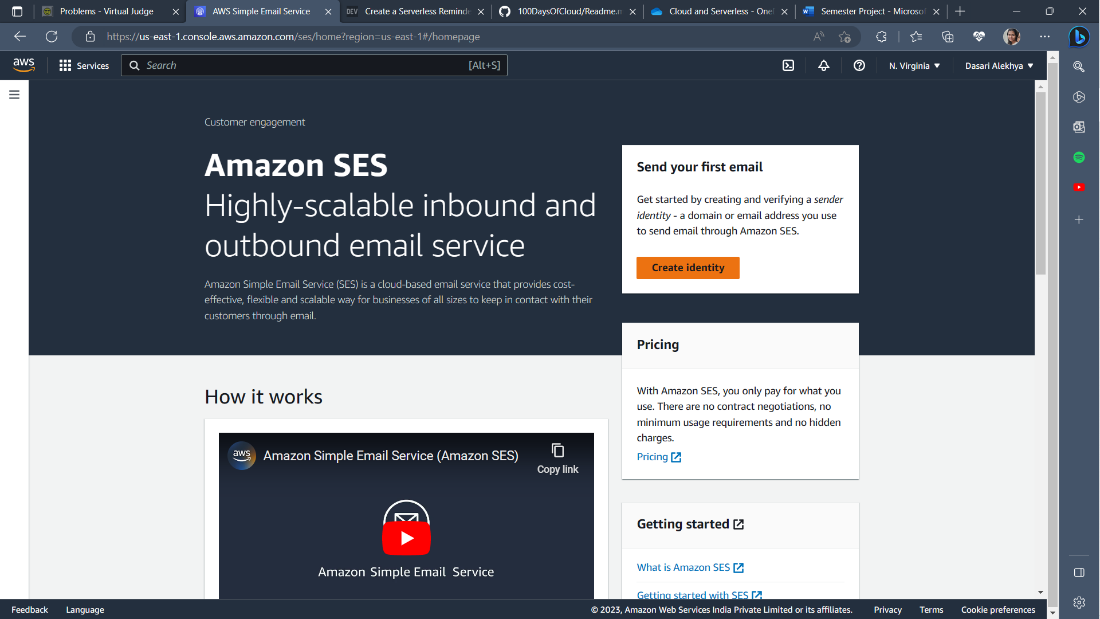
**Services covered:**

* Simple Email service
* CloudFormation
* IAM
* Lambda Function
* Step Functions
* API gateway
* S3

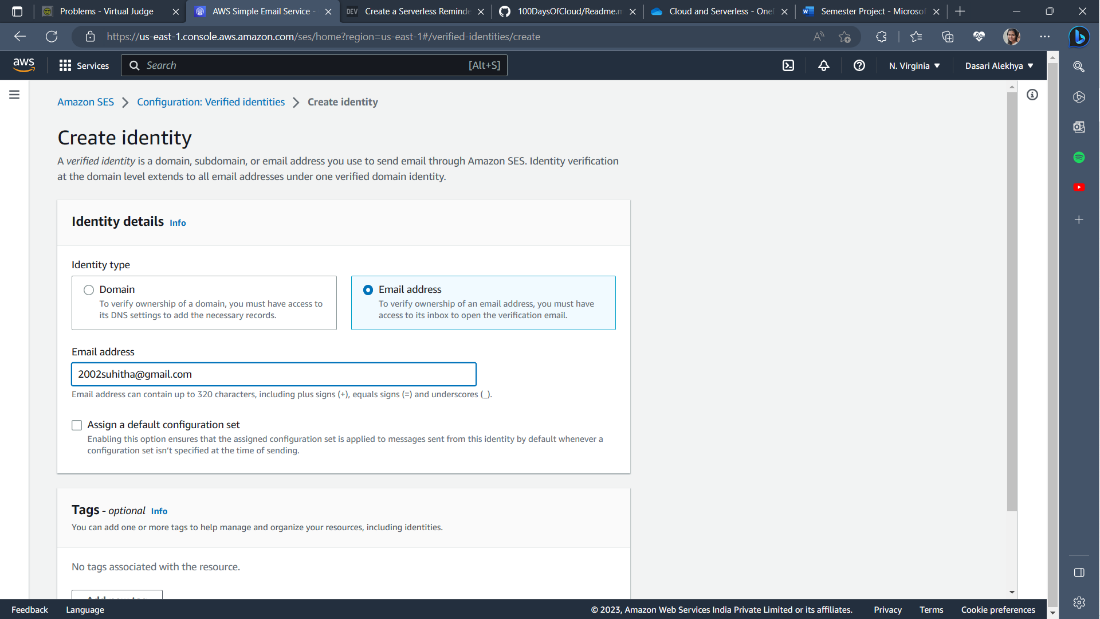
**Step-1: Configure Simple Email Service (SES)**

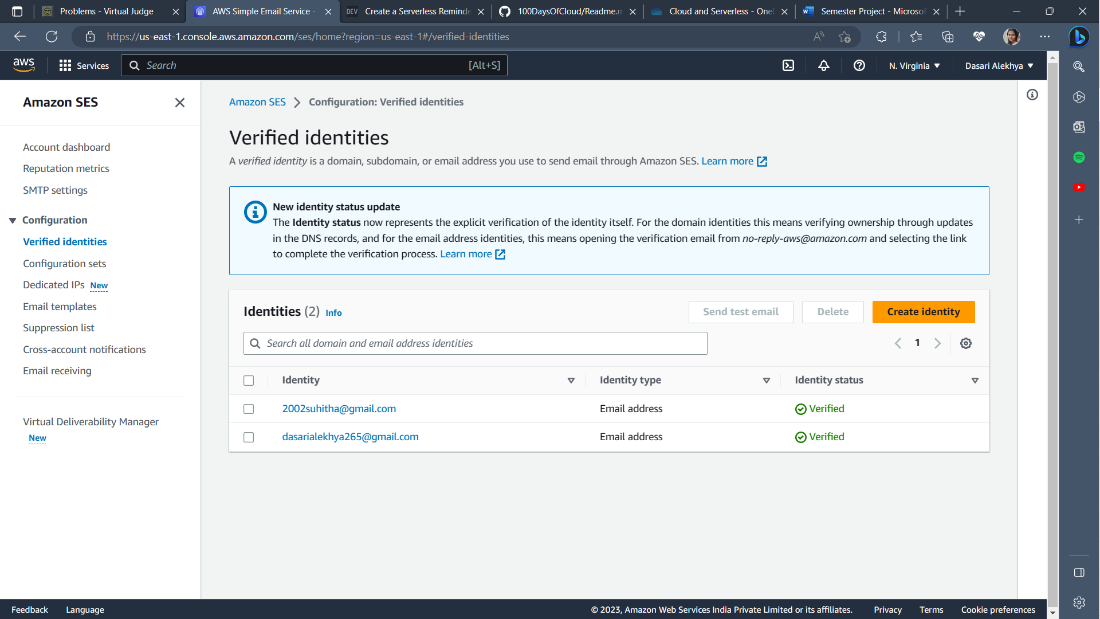
We need two emails for this. One for the web app to send the email from and another for the reminder from the app to be sent to (user).

* Search SES. Click verified identities on left side.
* Create identity.
* Check the 'Email Address' checkbox.
* Enter the email address that you want the application to use to send the reminder.
* Create Identity.



Create one more identity with which you want to receive the reminder





### **Step-2: Add a email Lambda function to use SES to send emails for the serverless application**

### In this stage, we need to create a Lambda function which will be used by the serverless application to create an email and then send it using SES. Before that we need to create an IAM role which Lambda will use to interact with other AWS services.

### **Create a Lambda Role in CloudFormation using the yaml file. Make sure you unzip the file.**

### Search CloudFormation and select Create Stack.

### Select Template is Ready option and select upload a template file and choose the lambdarolecfn.yaml file and upload it and click Next.

### Enter Stack name as LAMBDAROLE, click Next till you reach the last section.

### In the last section, scroll down and check the 'I acknowledge that AWS CloudFormation might create IAM resources' box and then click Create Stack.

### **Create the Lambda Function.**

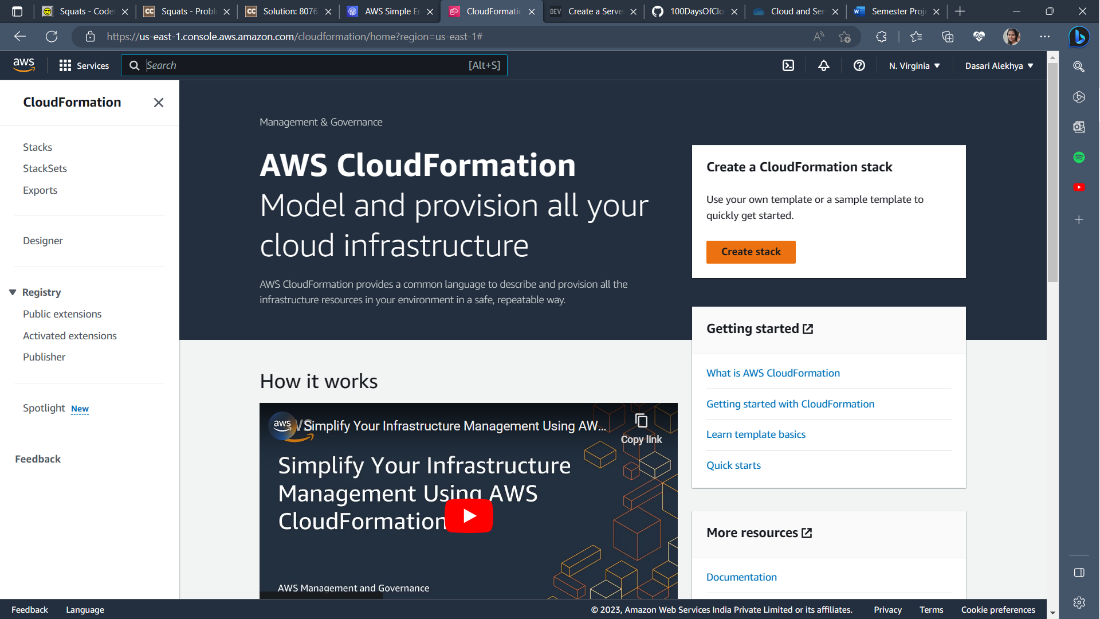
### Search Lambda and click on Create Function.

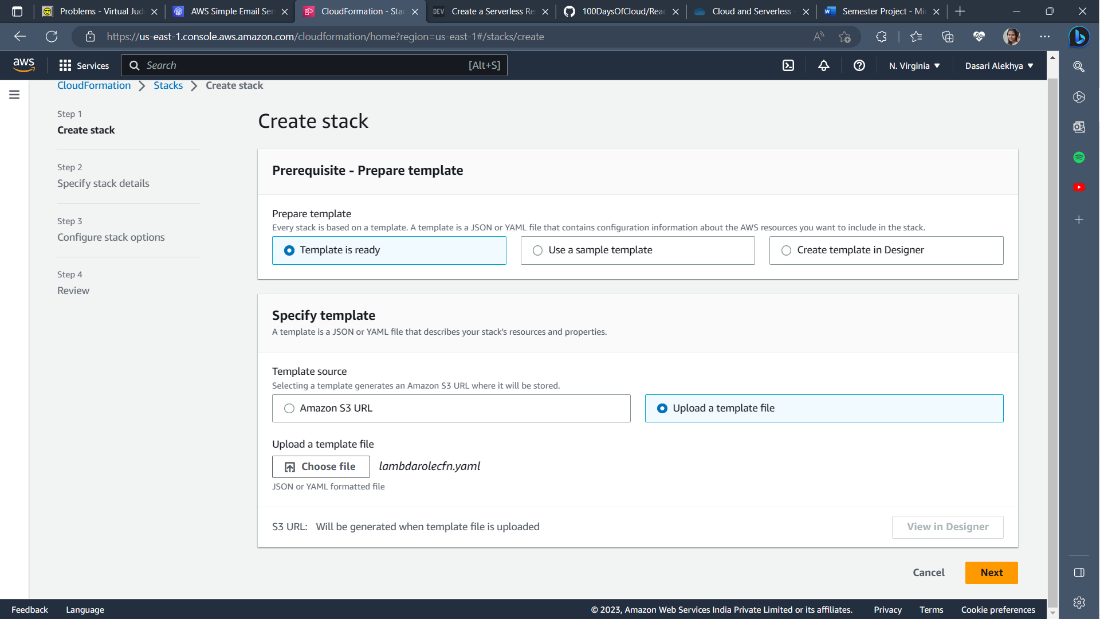
### Select Author from scratch. For Function name enter email\_reminder\_lambda .

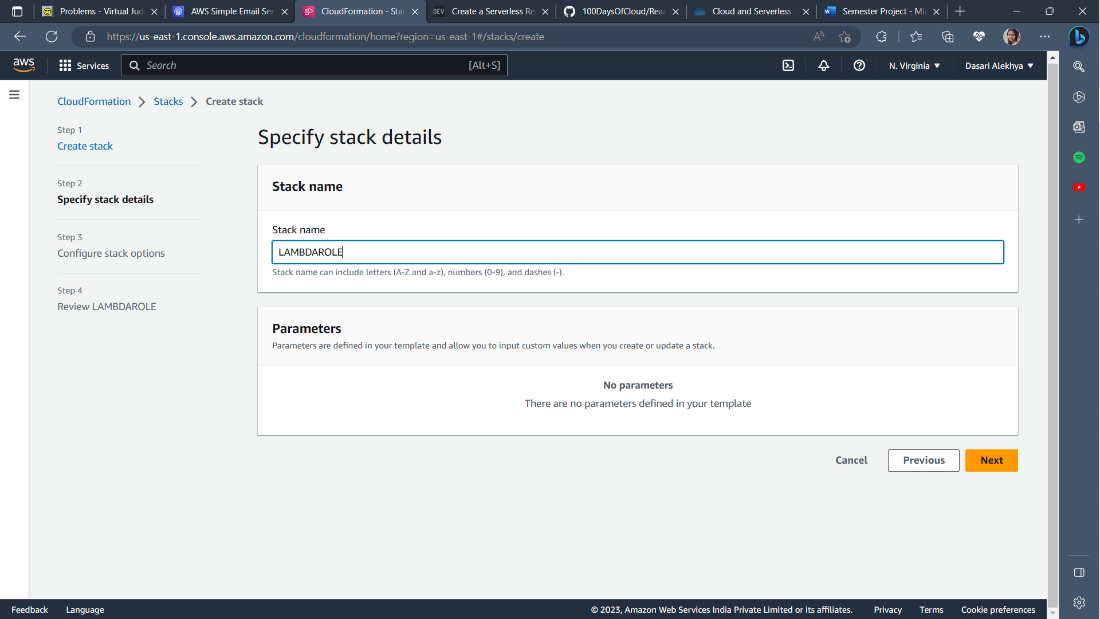
### For runtime click the dropdown and select Python 3.9.

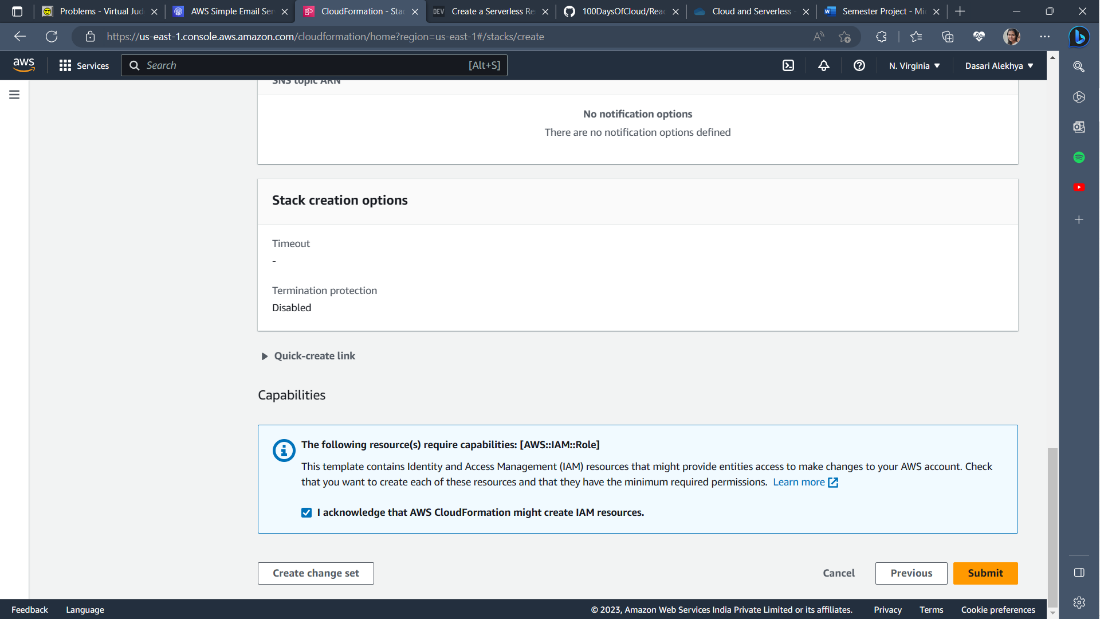
### Expand Change default execution role, select Use an existing role, click the Existing role dropdown and pick LambdaRole.

### Create the function.

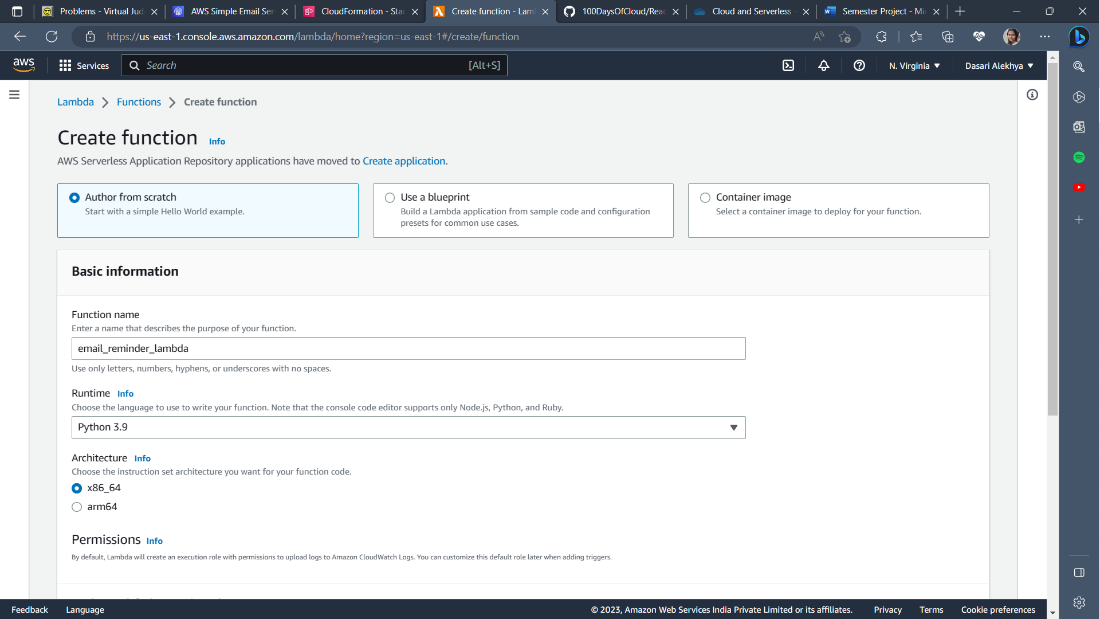


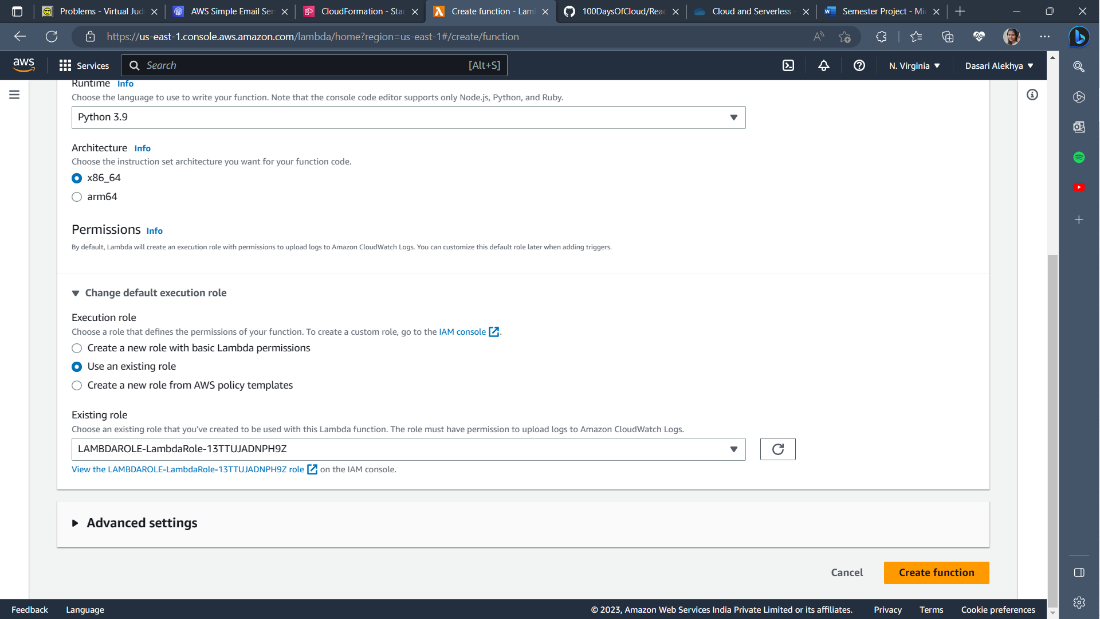


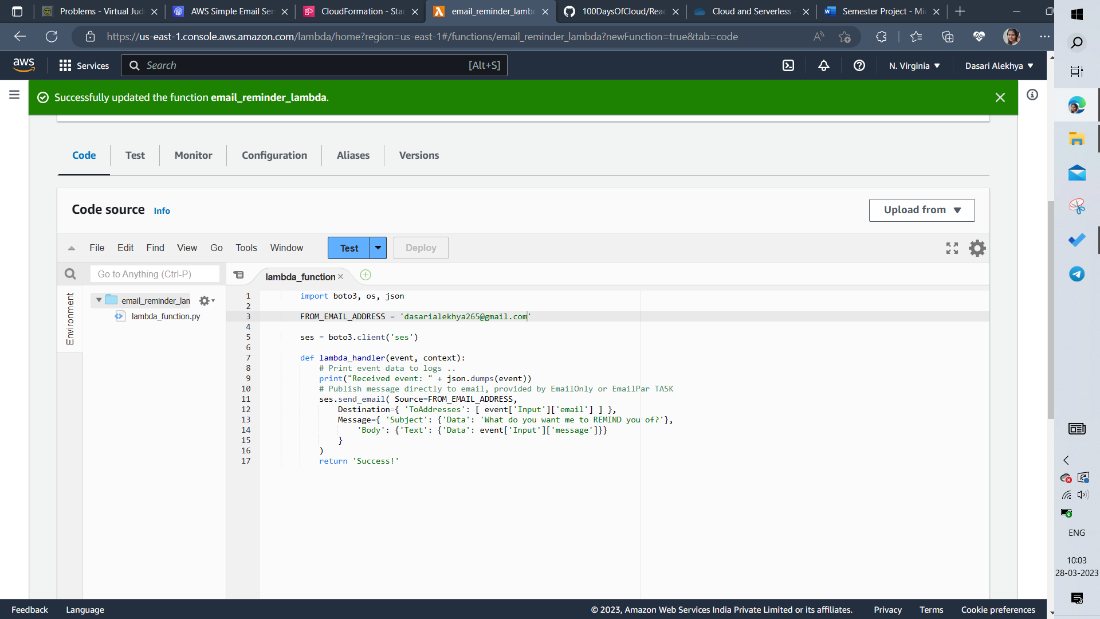




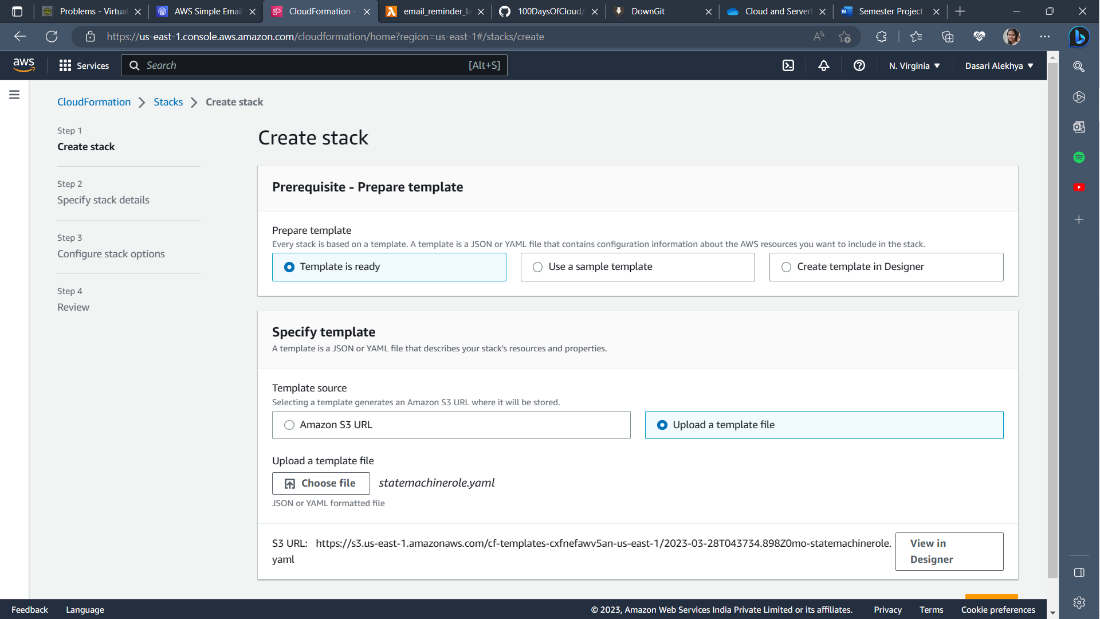
Create a lambda function

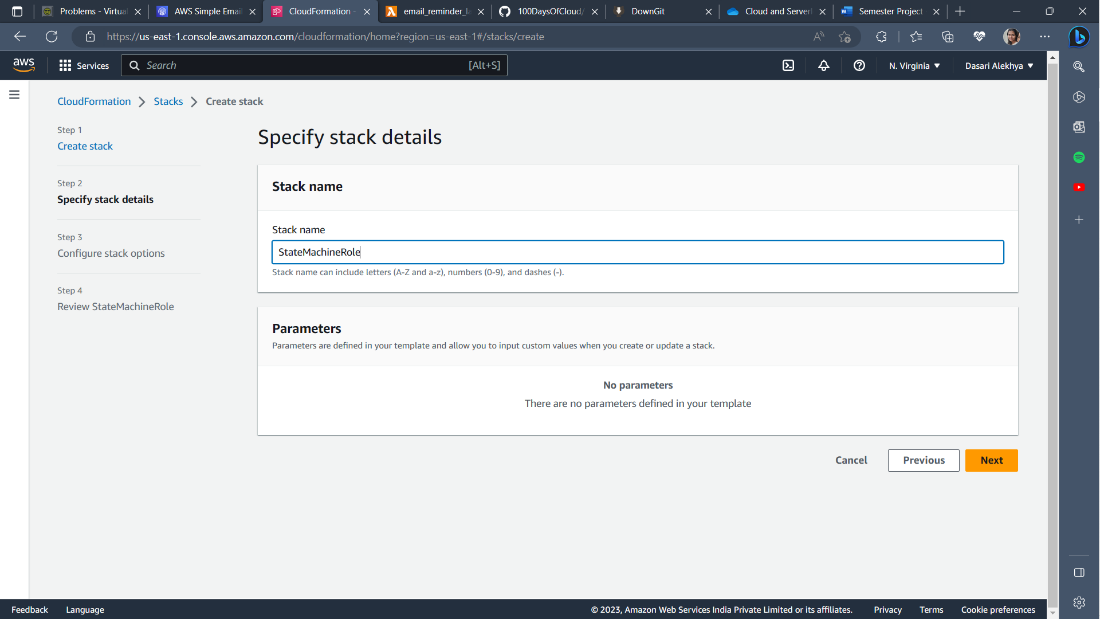






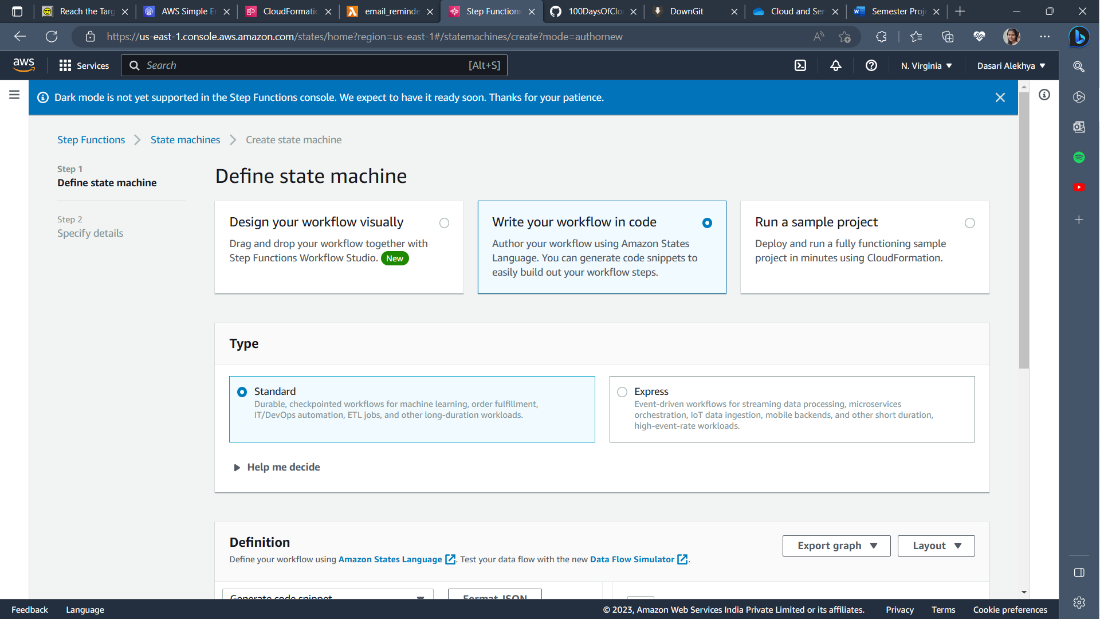
Create a new stack in cloudformation

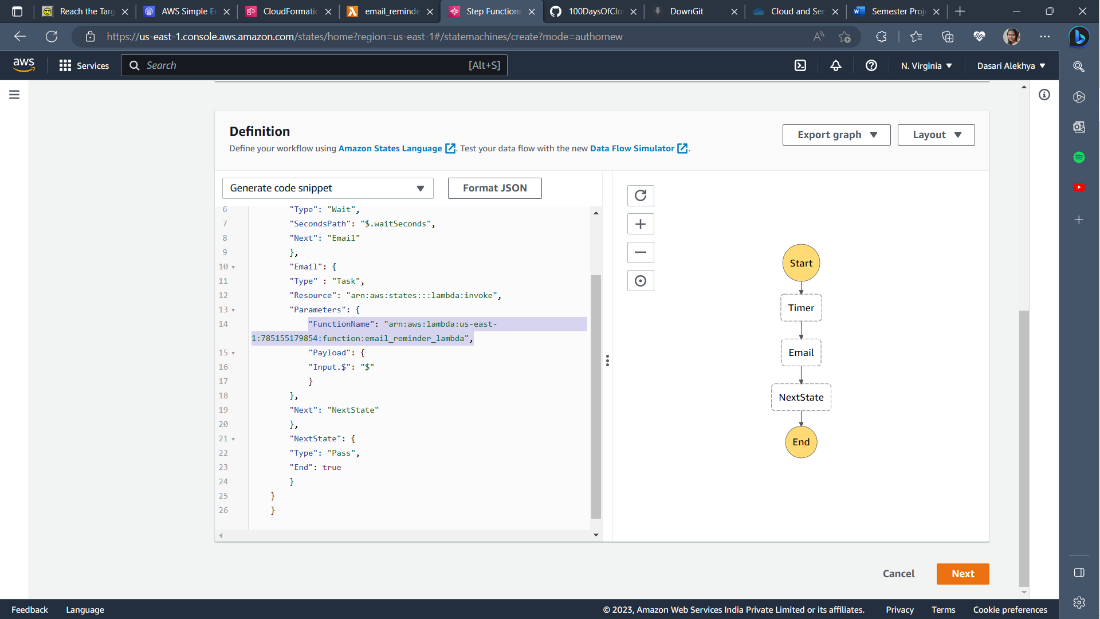


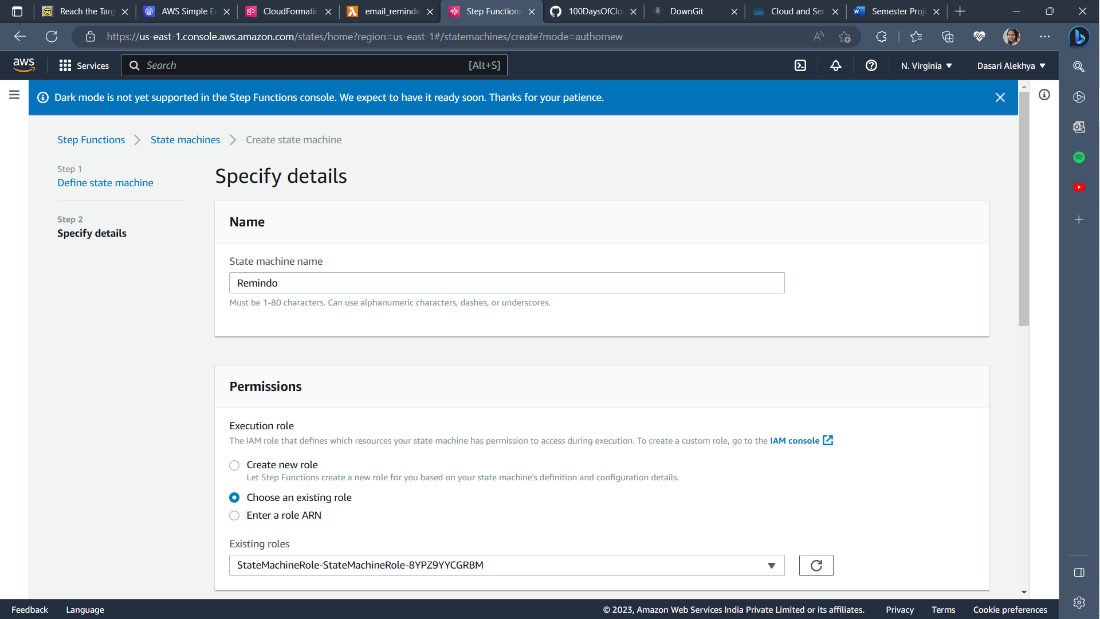


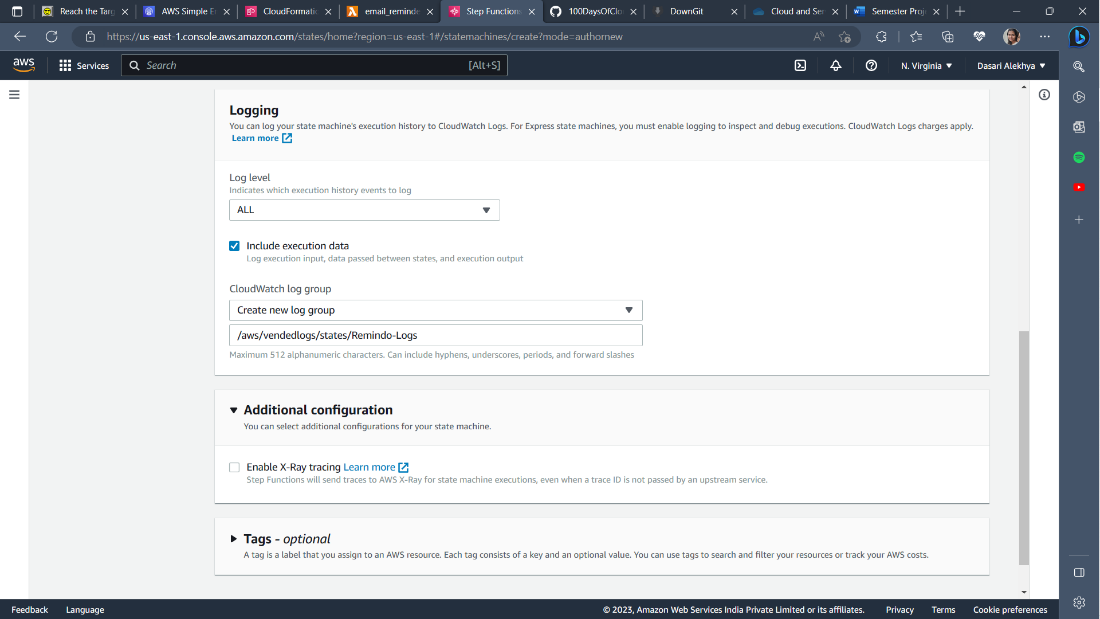
**Step-3: Implement and configure the state machine, the core of the application**

In step functions service, create a new state machine





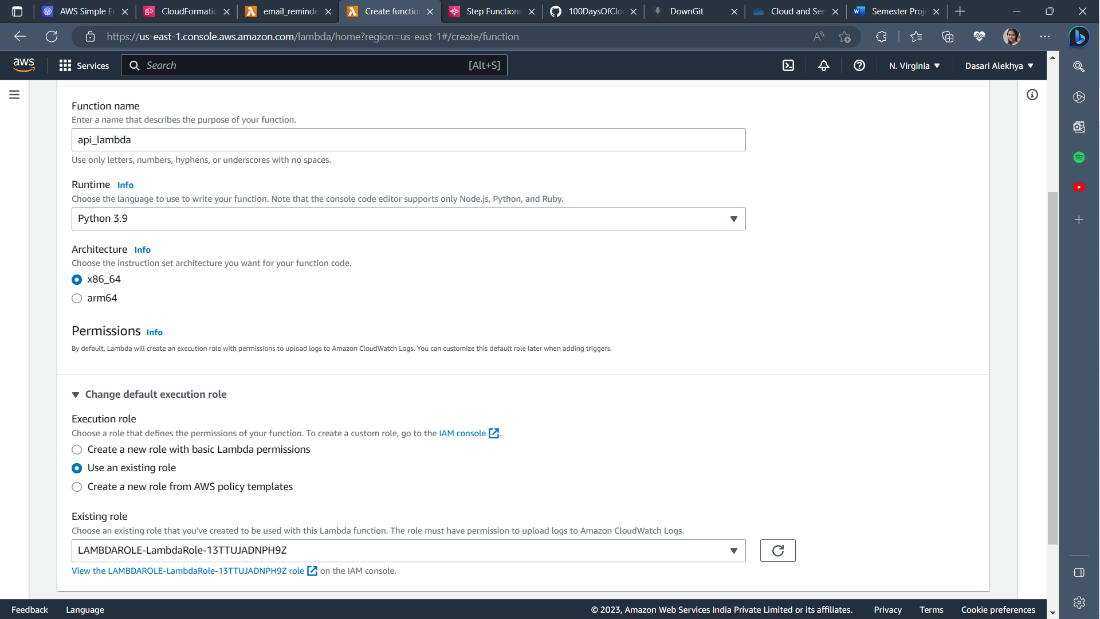


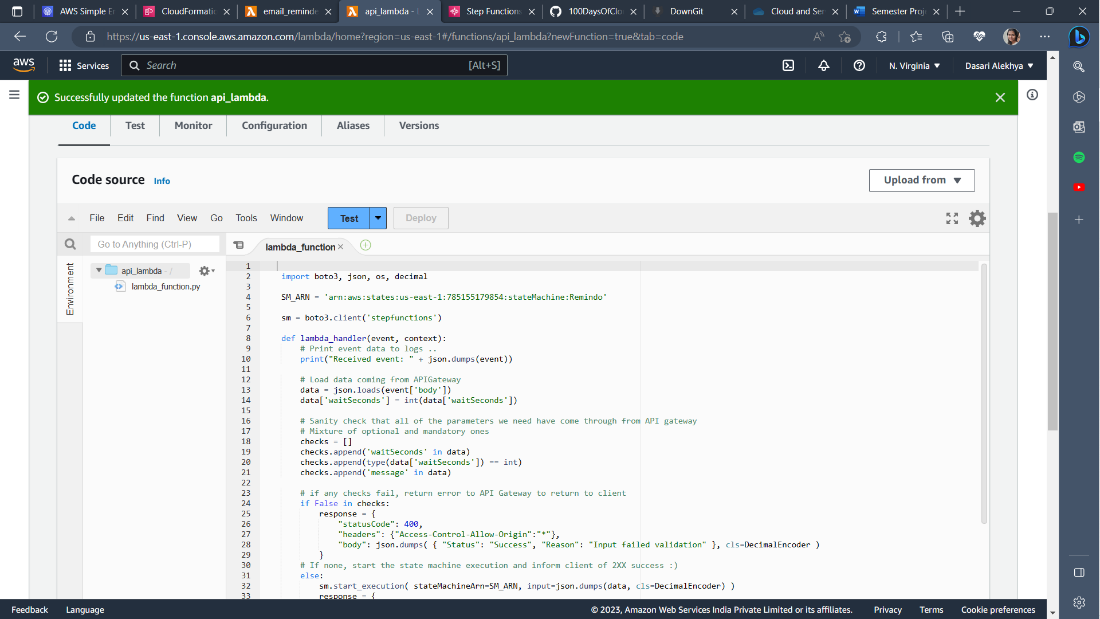


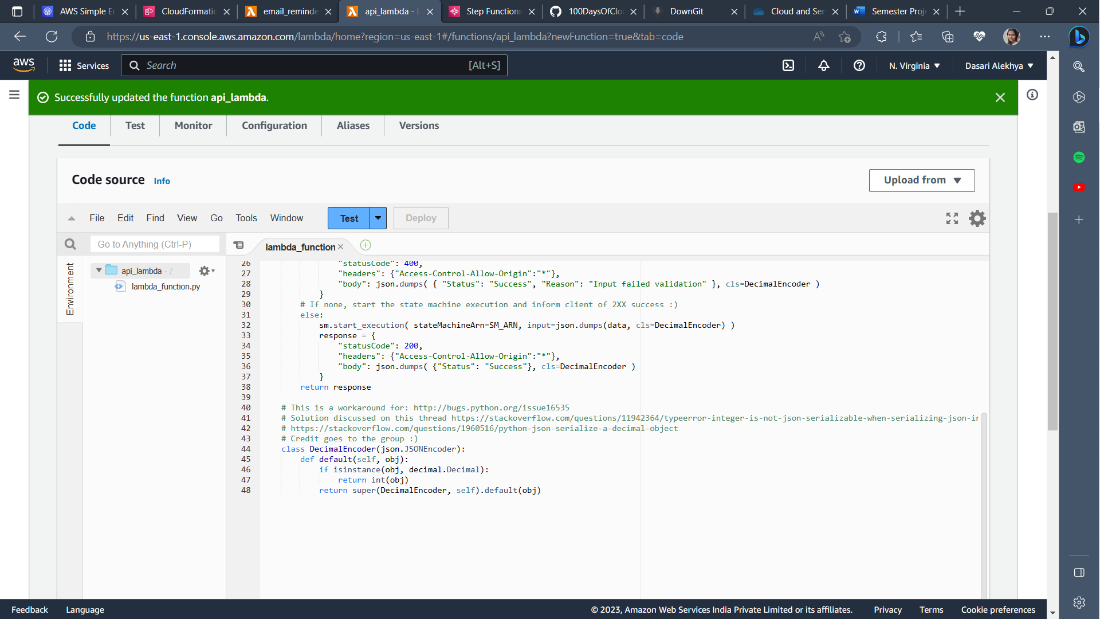
**Step-4: Implement the API Gateway, API and supporting Lambda function**

In this stage you will be creating the front end API for the serverless application. The front end loads from S3, runs in your browser and communicates with this API. It uses API Gateway for the API Endpoint, and this uses Lambda to provide the backing compute. First you will create the supporting Lambda function and then the API Gateway.

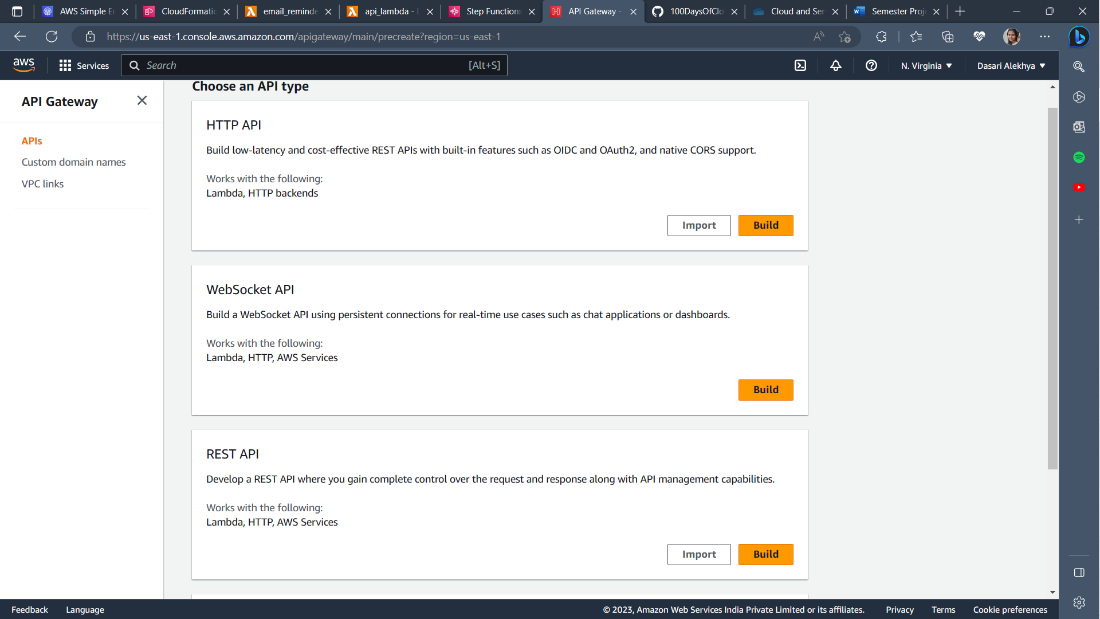
* **Create the Lambda Function.** 
  + Search Lambda and click on Create Function.
  + Select Author from scratch.
  + For Function name enter api\_lambda .
  + For runtime click the dropdown and select Python 3.9.
  + Expand Change default execution role, select Use an existing role, click the Existing role dropdown and pick LambdaRole.
  + Create the function.

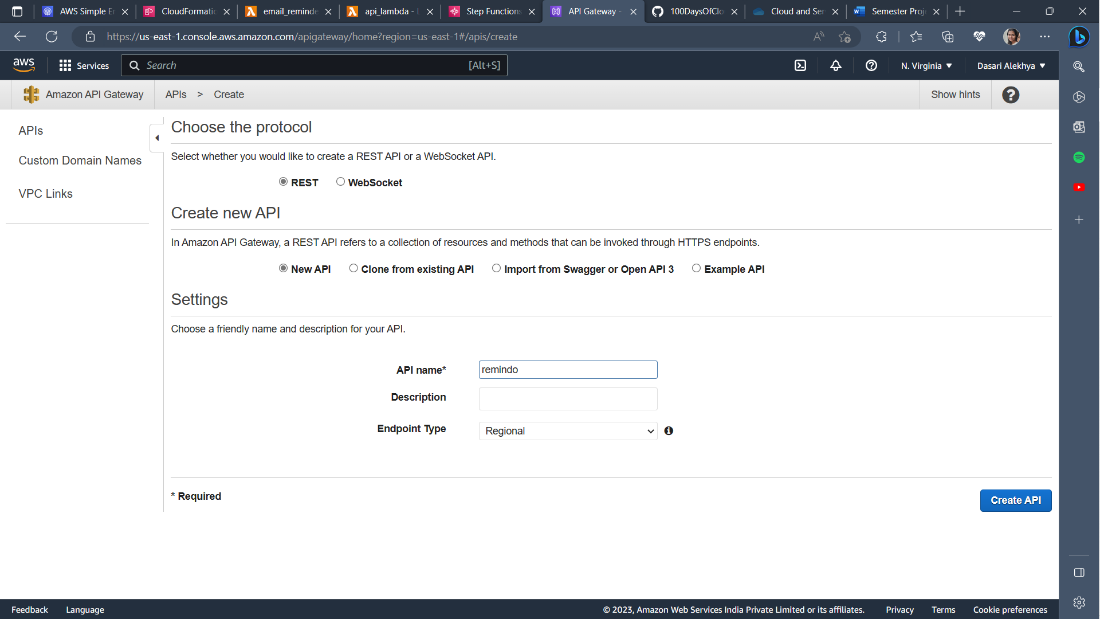


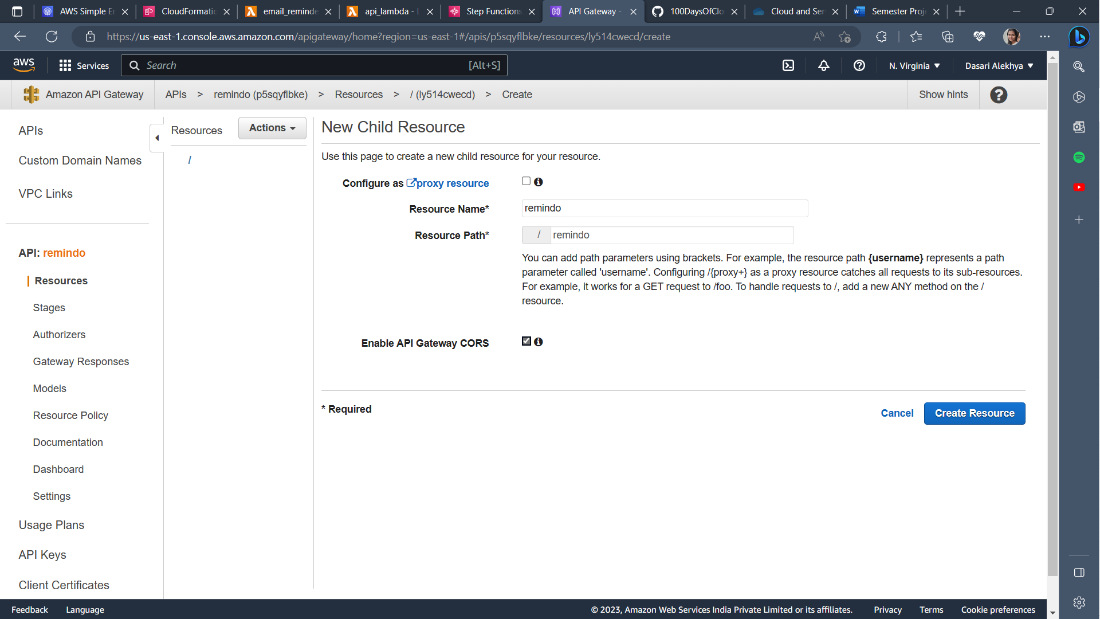


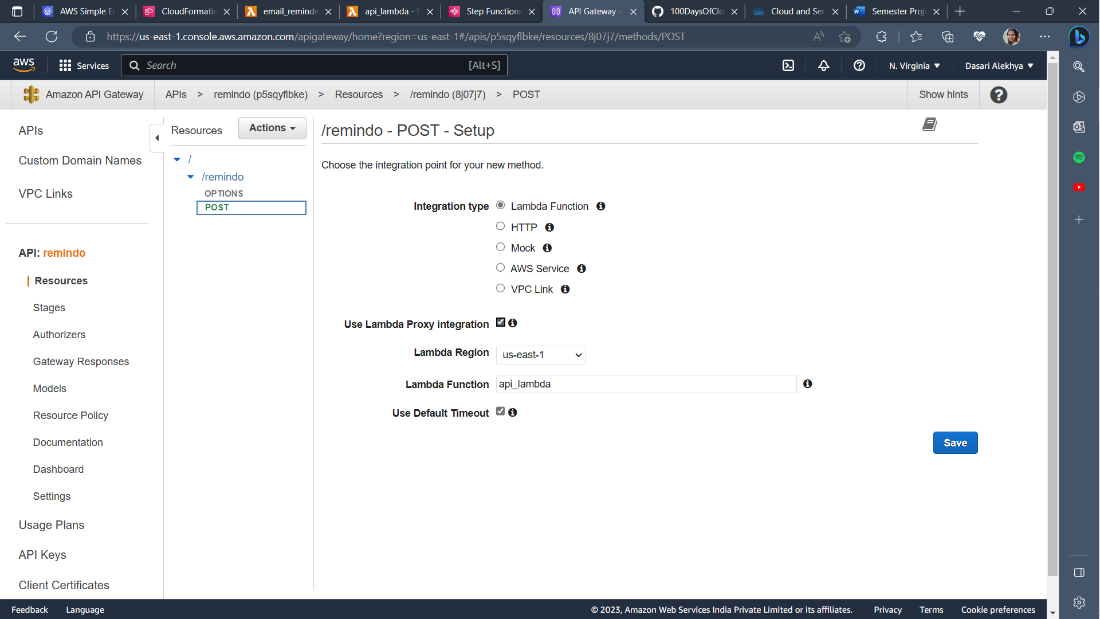


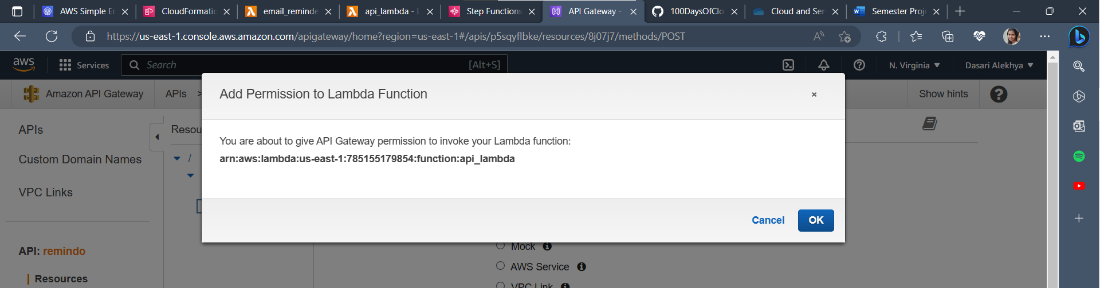
* **Create an API Gateway.** 
  + Click APIs on the menu on the left.
  + Locate the REST API box, and click Build.
  + Under Create new API ensure New API is selected. For API name\* enter remindo
  + For Endpoint Type, pick Regional.
  + Click create API.
  + Click the Actions dropdown and Click Create Resource.
  + Under resource name enter remindo .
  + Make sure that Configure as proxy resource is NOT ticked.
  + Tick Enable API Gateway CORS.
  + Click Create Resource.
  + Ensure you have the /remindo resource selected, click Actions dropdown and click create method. In the small dropdown box which appears below /remindo select POST and click the tick symbol next to it.
  + Ensure for Integration Type that Lambda Function is selected.
  + Make sure us-east-1 is selected for Lambda Region. In the Lambda Function box, start typing api\_lambda and it should autocomplete, click this auto complete (Make sure you pick api\_lambda and not email reminder lambda).
  + Make sure that Use Default Timeout box is ticked.
  + Make sure that Use Lambda Proxy integration box is ticked.
  + Click Save.
  + Click Actions Dropdown and Deploy API.
  + For Deployment Stage select New Stage. -For stage name and stage description enter prod.
  + Click Deploy.

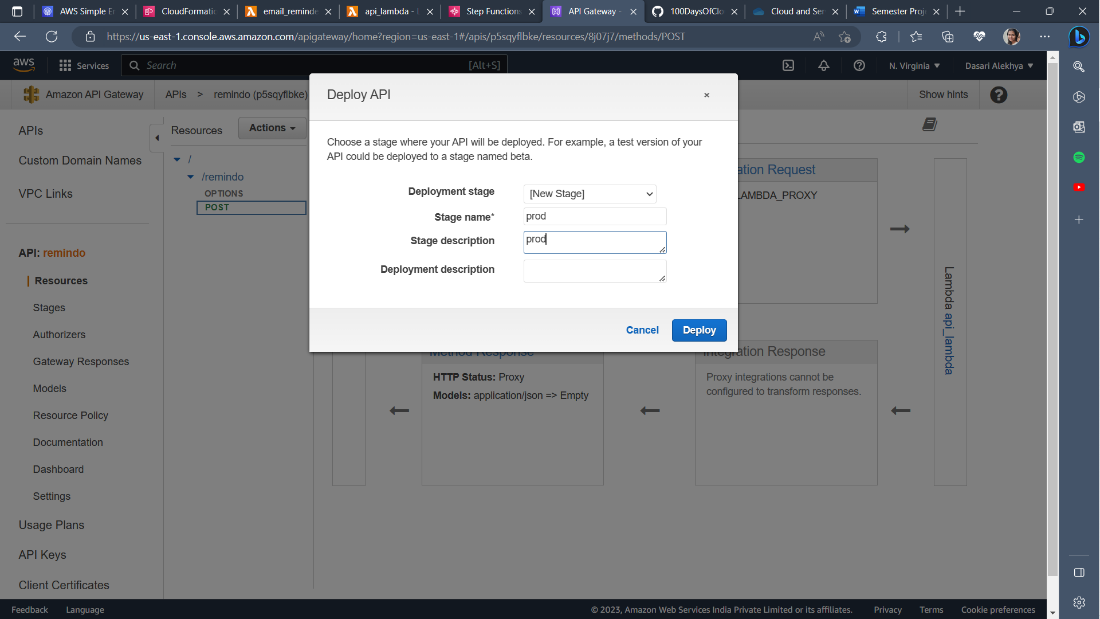










\

**Step-5: Implement the static frontend application and test functionality**

In this stage, you will create an S3 bucket and static website hosting which will host the application front end. You will download the source files for the front end, configure them to connect to your specific API gateway and then upload them to S3. Finally, you will run some application tests to verify its functionality.

* **Create S3 Bucket**.
  + Search S3. Click Create Bucket.
  + Give a unique bucket name and ensure the region is set to US East (N.Virginia) us-east-1 .
  + Uncheck Block all public access. Tick the box under Turning off block all public access might result in this bucket and the objects within becoming public.
  + Create Bucket.
  + Make it publicly accessible using a bucket policy.
  + Go into the bucket you just created. Click the Permissions tab.
  + Scroll down and in the Bucket Policy area, click Edit and in the box, paste the code below :

{

"Version":"2012-10-17",

"Statement":[

{

"Sid":"PublicRead",

"Effect":"Allow",

"Principal": "\*",

"Action":["s3:GetObject"],

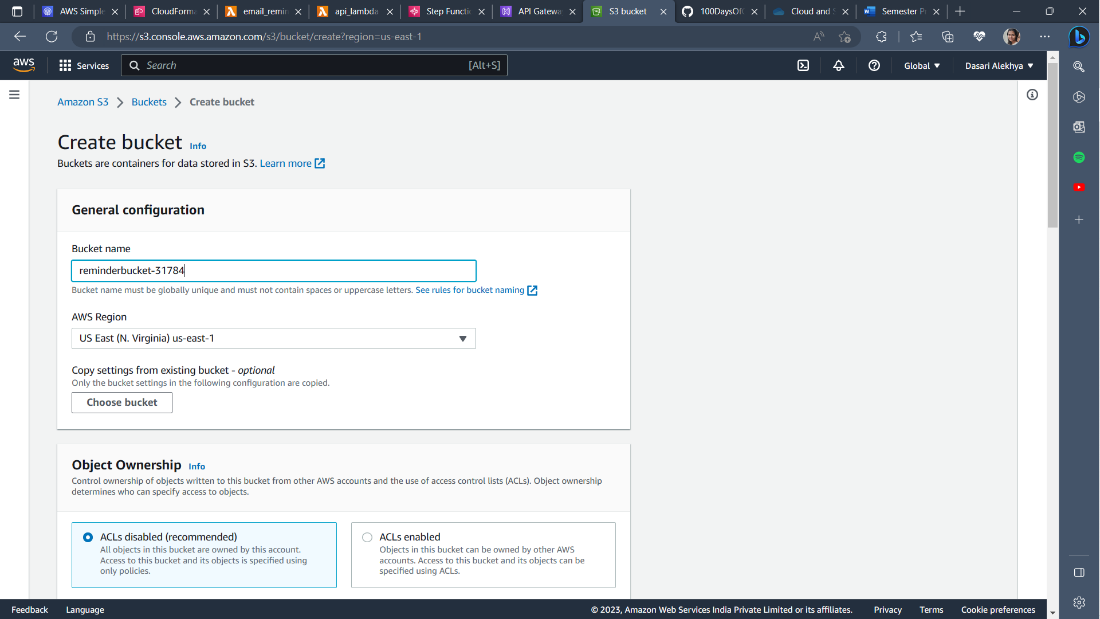
"Resource":["REPLACEME\_BUCKET\_ARN/\*"]

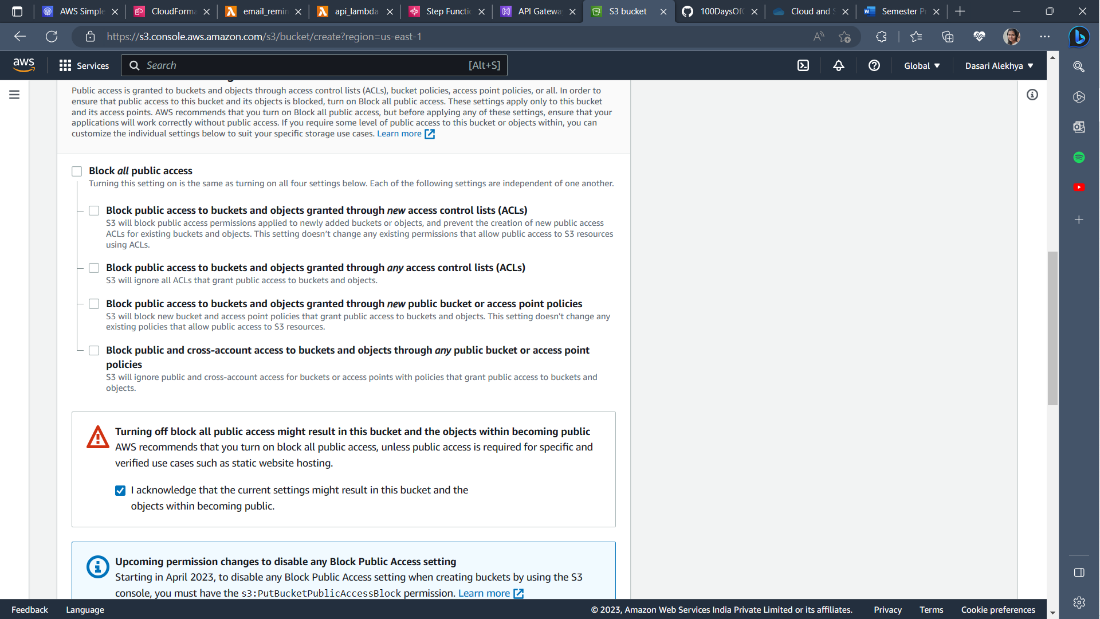
}

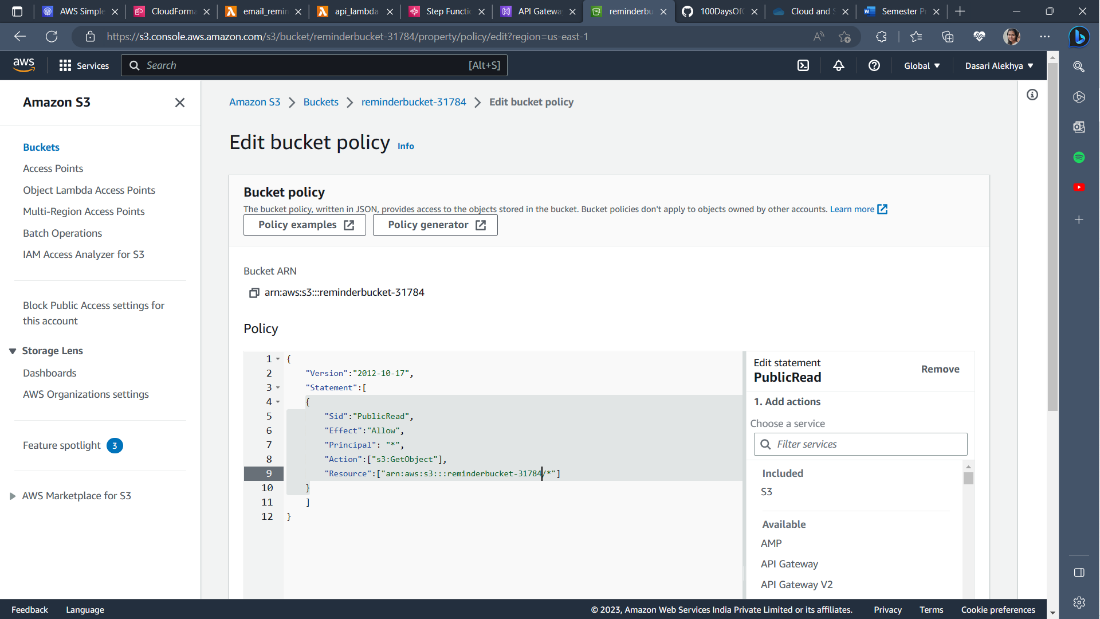
]

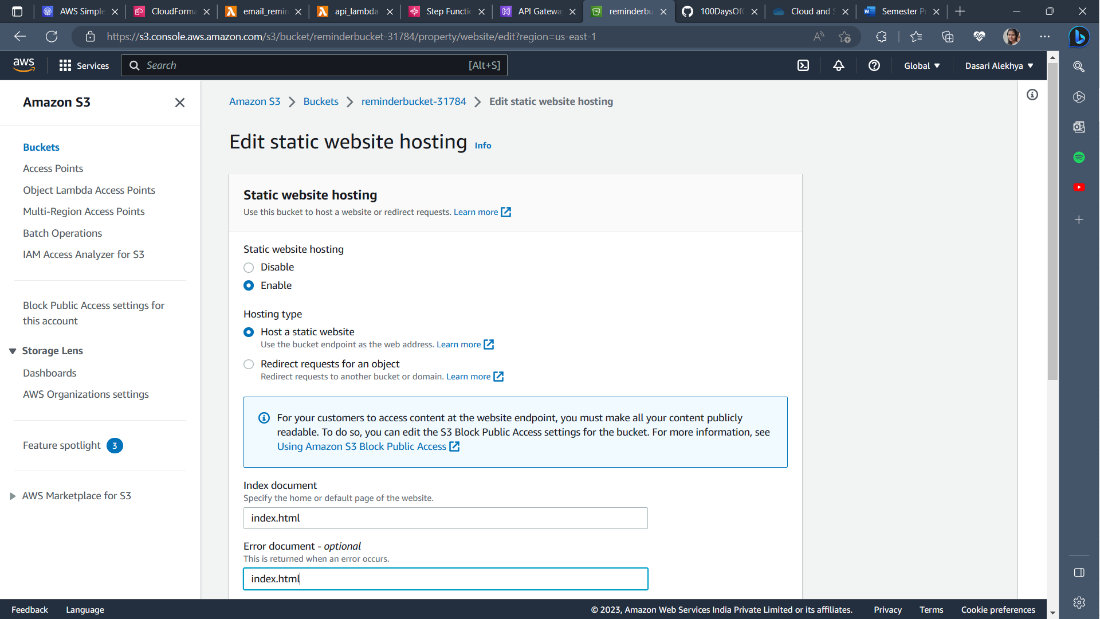
}

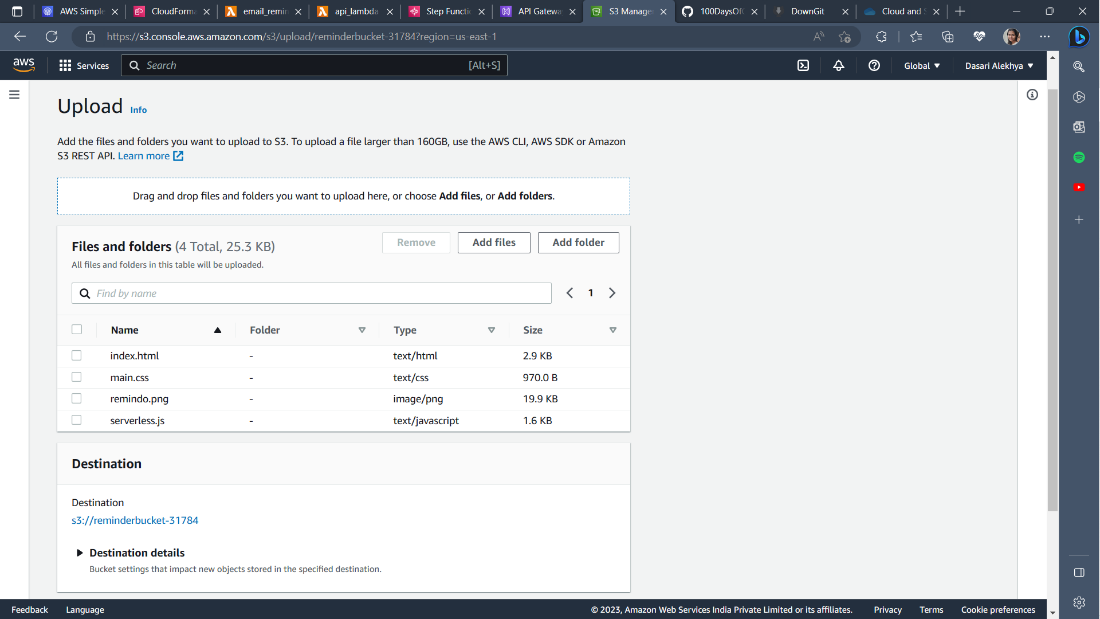
* + Replace REPLACEME\_BUCKET\_ARN with your bucket ARN and save changes.
* **Enable Static Hosting.**
  + Click on the Properties Tab.
  + Scroll down and locate Static website hosting and click Edit.
  + Select Enable Select Host a static website.
  + For both Index Document and Error Document enter index.html.
  + Click Save Changes.

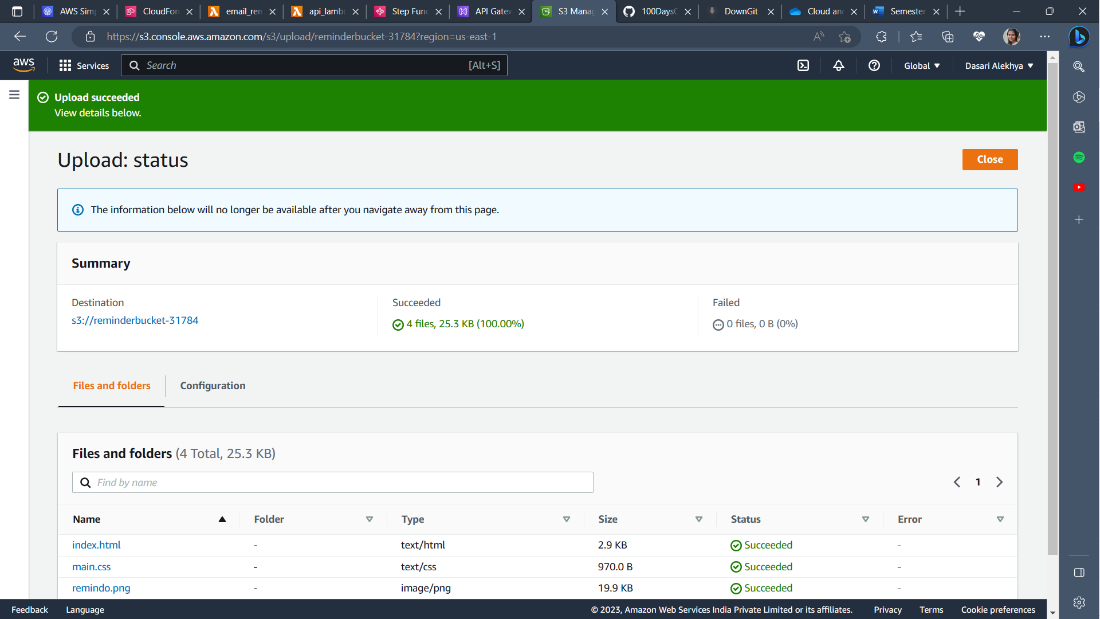












* **Testing.**
  + Go to the S3 bucket, properties tab and scroll all the way down and click on the link (bucket website endpoint).
  + Enter the details.
  + The email you should give is the email you created and verified in SES to receive the reminder.
  + Click Email.
  + Wait until the time period (seconds you mentioned) to receive the reminder as an email.
  + Check the email.

