**Cloud Computing**

Develop a web page with student information and make web pages accessible to public through browser. Illustrate the use of AWS EC2 and S3 for developing the website, Use EC2 for installing Apache server and S3 for storing webpage details

**GRP-BANGLORE-NOIDA**

TEAM MEMBERS-

1. **ASHISH RANJAN - (2021CT93023)**
2. **RASHI CHAUDHARY- (2021CT93024)**
3. **SANDYA SHANMUGHAM-(2021CT93031)**
4. **RITU BALA RAI- (2021CT93003)**

S1-21\_SECTSZG527

Table of Contents

1. Overview1
2. EC2 Management Console4
3. Steps for Creating EC2 instance…………………………………………...............................
4. Installing Apache server using ec2 instance………………………………………………………
5. Screenshot of EC2 instance running………………………............................................
6. **S3 Management Console……………………………………………………………………………………………………….**
   * + - 1. Steps for Creating S3 bucket…………………………………………………………………………....
         2. Uploading the folder in S3 bucket…………………………………………………………………….
         3. Screenshot of Amazon S3 instance running……………………………………………………..
7. **Contact Form Email using AWS SES ……………………………………………………………………………………….**
   * + - 1. Introduction of AWS SES…………………………………………………………………………………..
         2. Steps for creating Amazon SES identity..………………………………………………………….
         3. Steps for creating Lambda function…………………………………………………………………
         4. Steps for creating API gateway………………………………………………………………………..
8. **The public DNS of your instance……………………………………………………………………………………………**

**6) Screenshot of browser using the public DNS of your instance………………………………………………**

## **Overview**

## **Introduction**

Amazon Web Services (AWS) is BITS’s preferred and recommended cloud service for faculty-led computational needs. The service provides access to Amazon’s cloud computing services, including computing, storage, database, etc. The service is a multi-mission platform that can facilitate the advancement of science, education, and service across the University.

In this course we learn beyond lift-and-shift to elevate our workloads in the cloud.

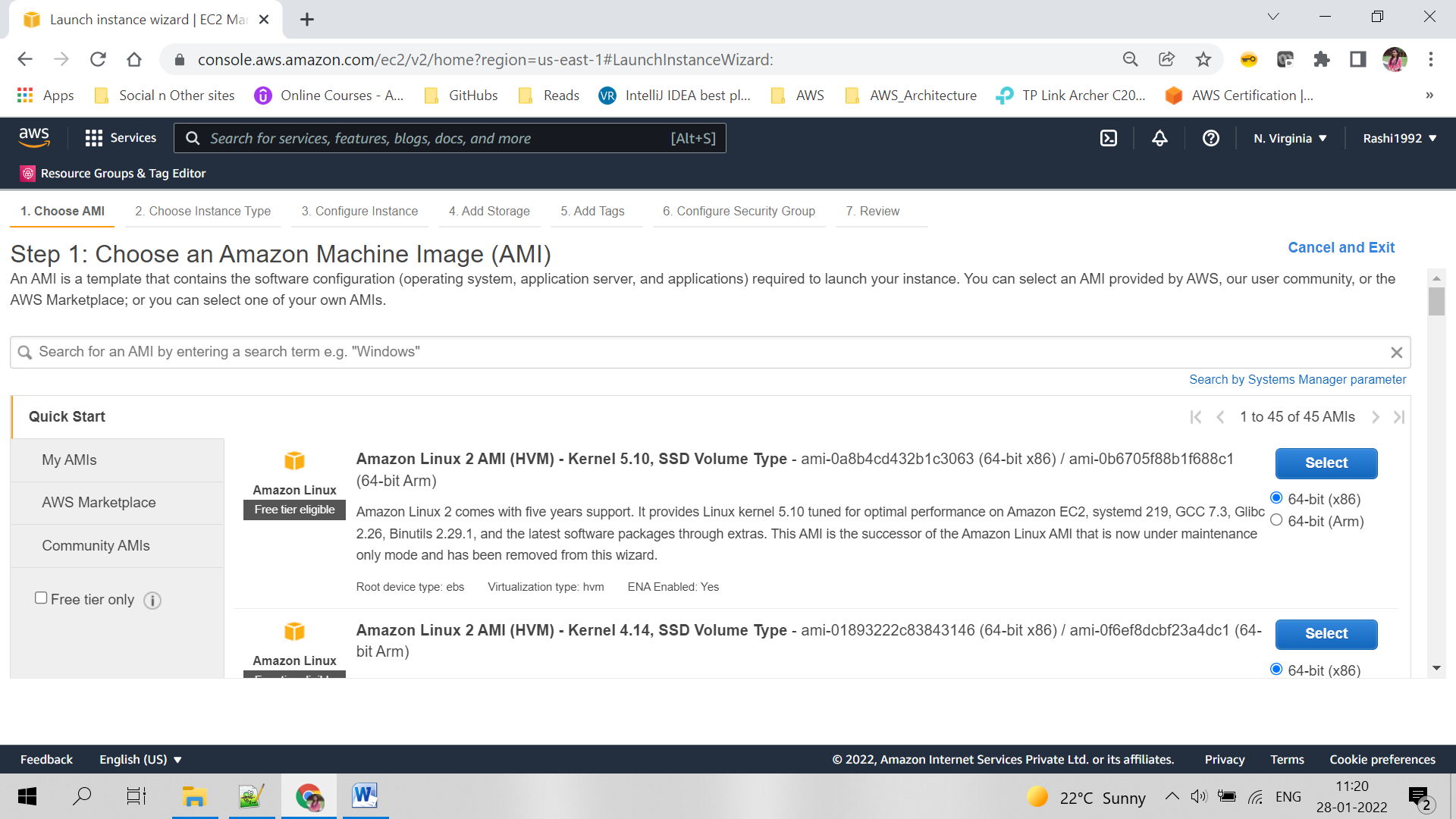
* Access continued support along cloud journey.
* Demystify our AWS spend and maximize ROI.
* Extend cloud native to on-premises with AWS Outposts.

The list of AWS services we have used while implementing this project are:

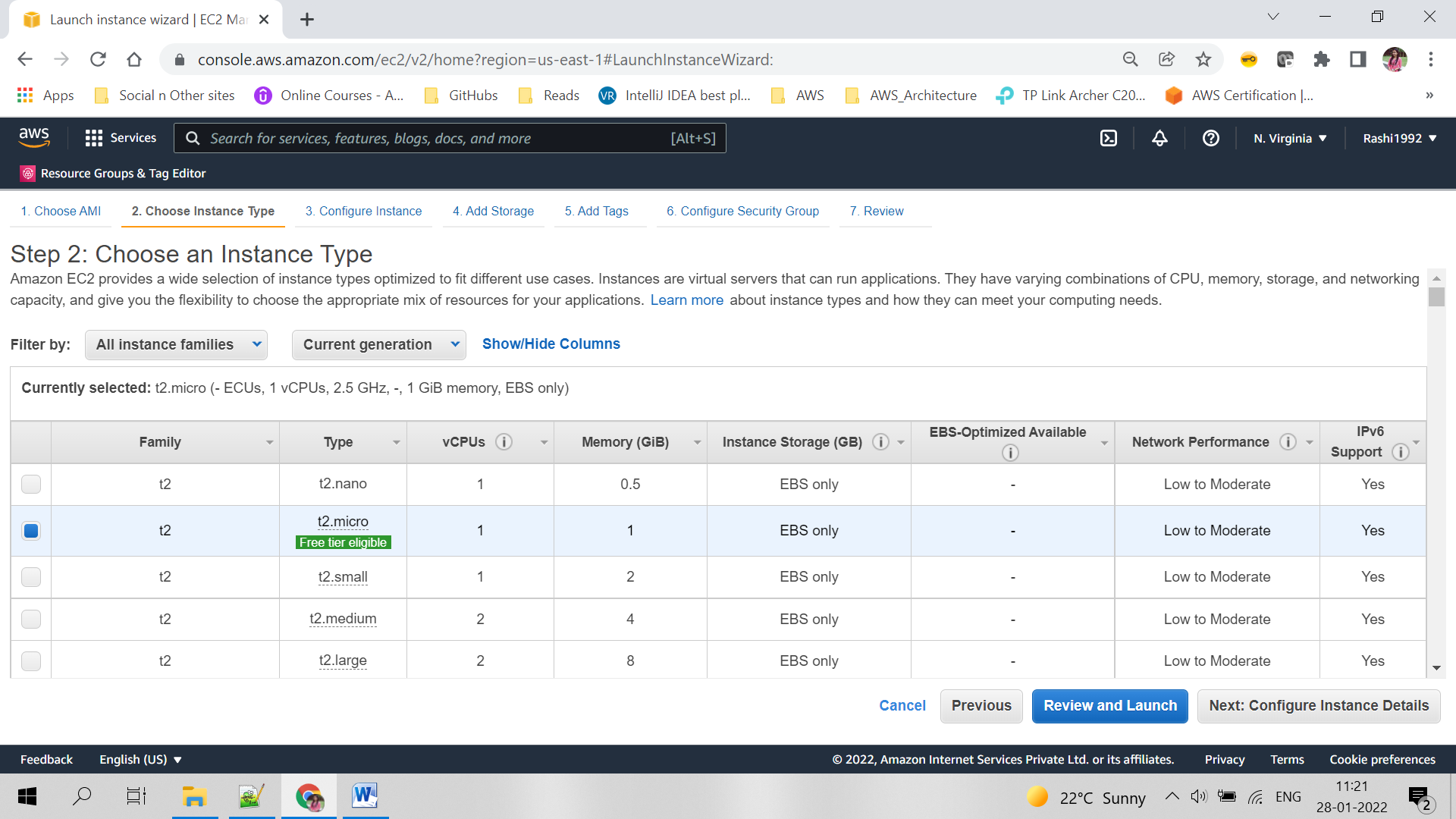
* Amazon EC2(Elastic Cloud Compute) Service
* Amazon S3 or Amazon Simple Storage Service
* Amazon Simple Email Service
* Amazon Lambda
* Amazon API Gateway

1-Steps for creating EC2 instance

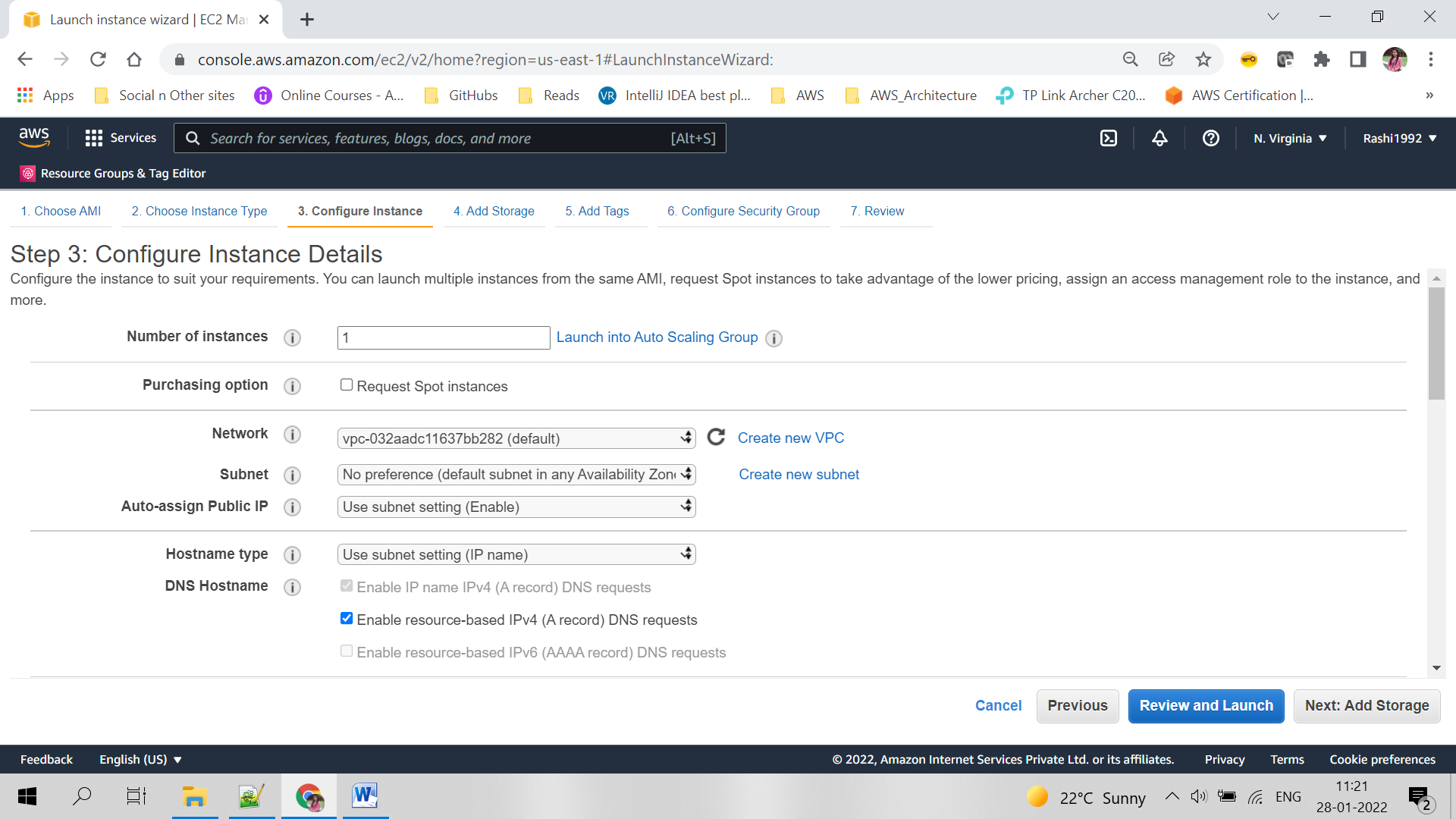
Search EC2 instances on search bar and click on launch instance.



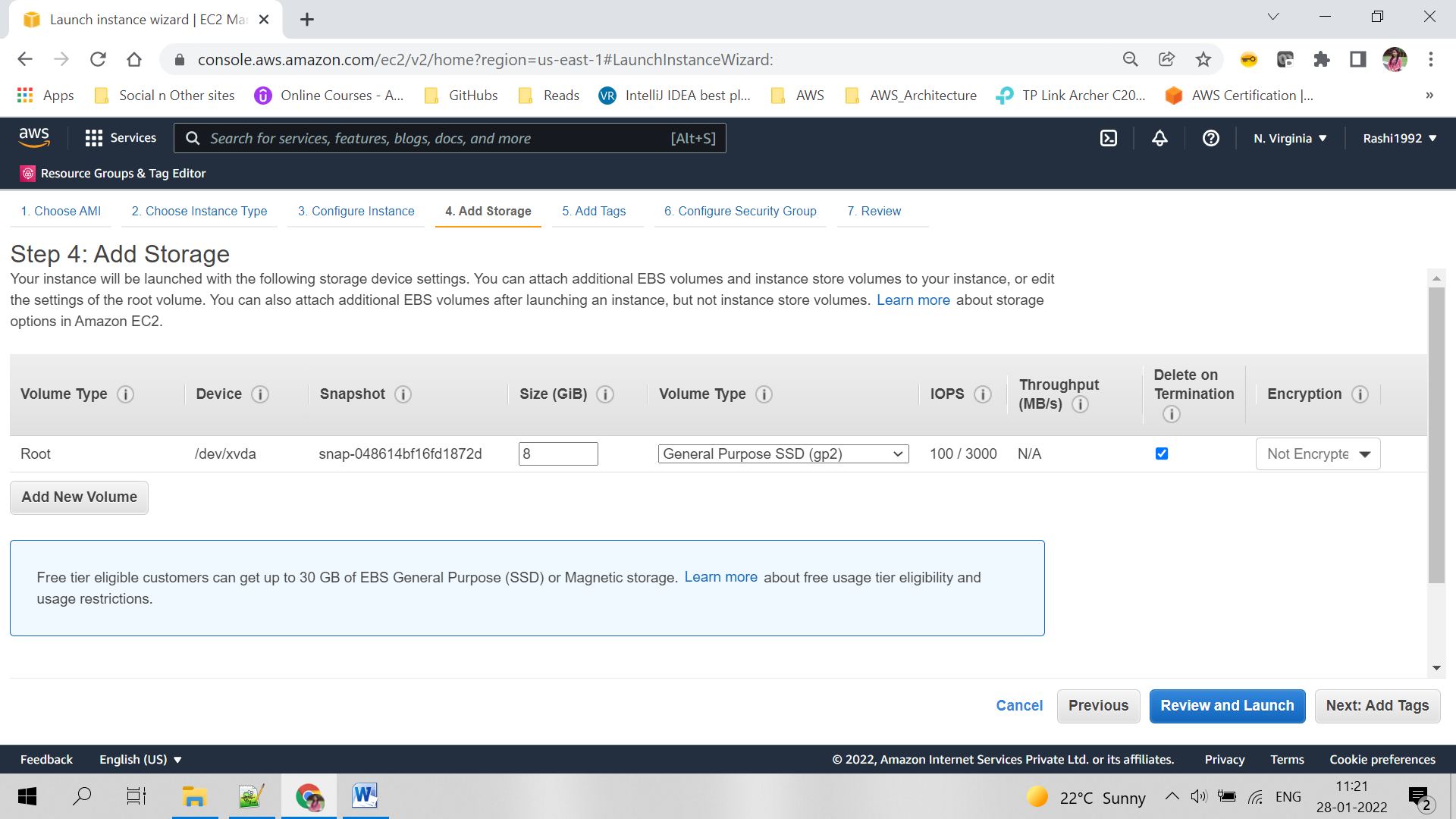
Select the required instance setup, click on Next: Configure Instance Details



Select all configuration details for instance, click Next: Add Storage



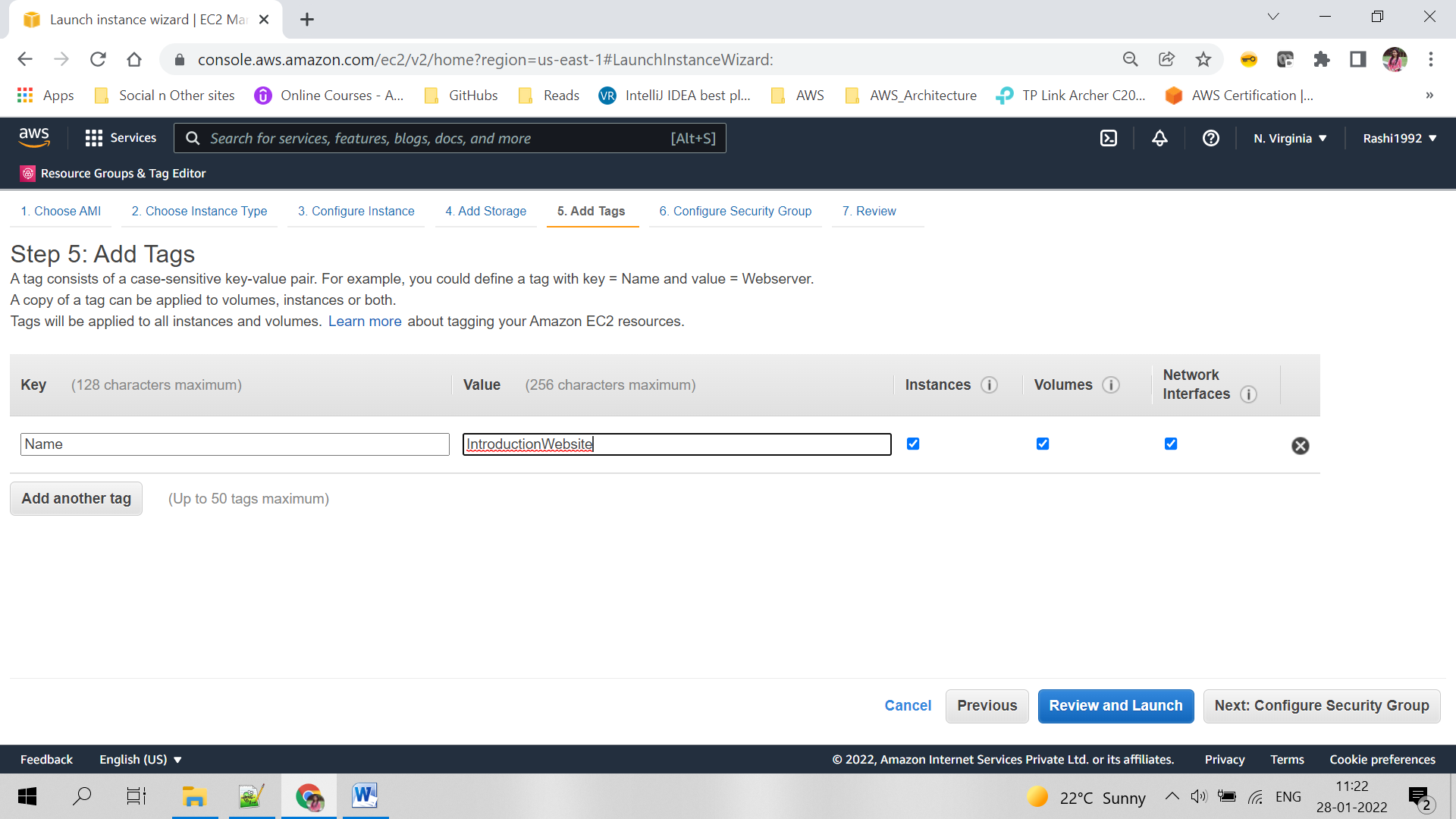
Add extra storage if required otherwise leave default one



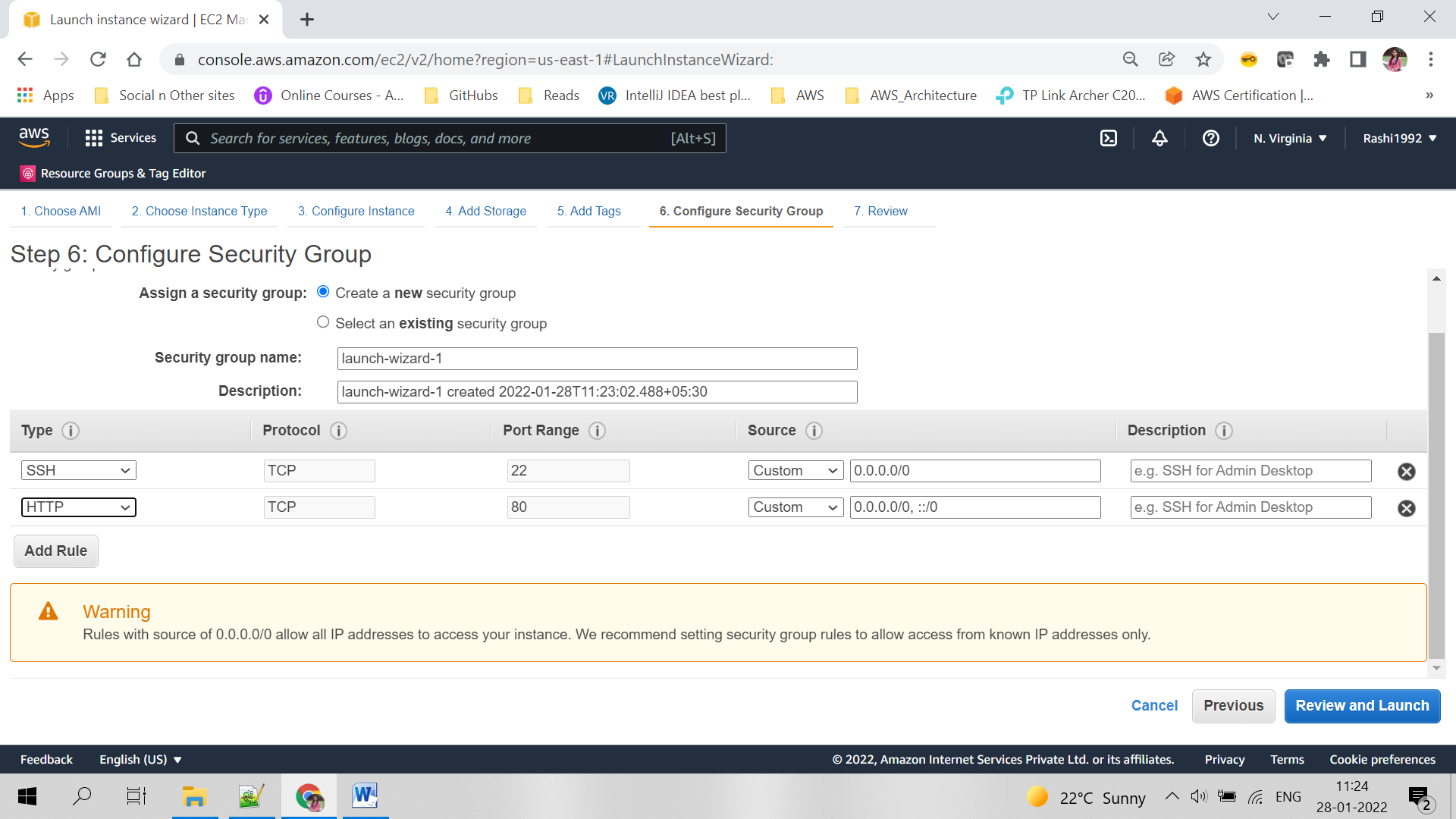
Add tags in key value like

Key-Name

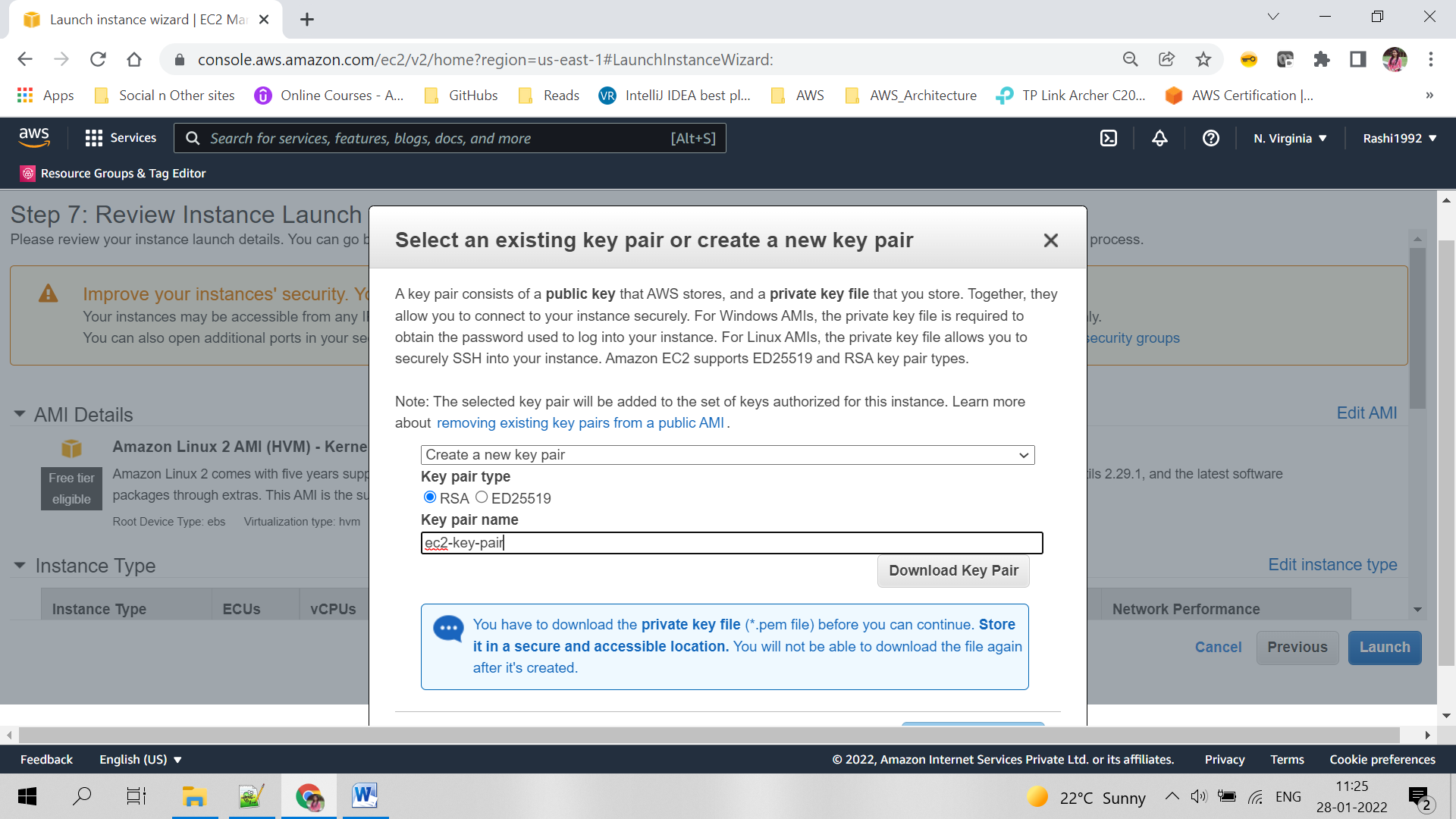
Value- EC2\_DeploymentMachine

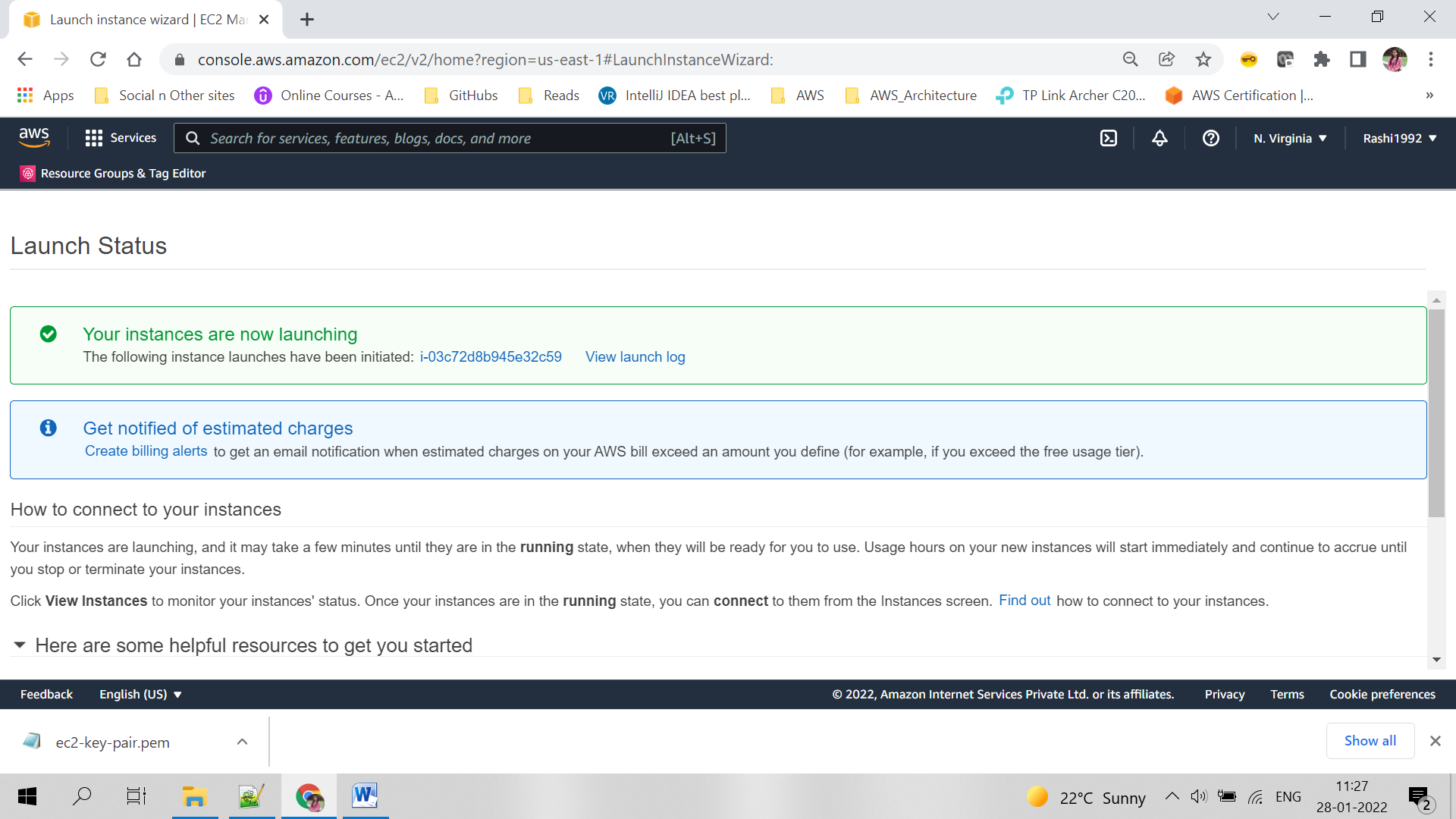


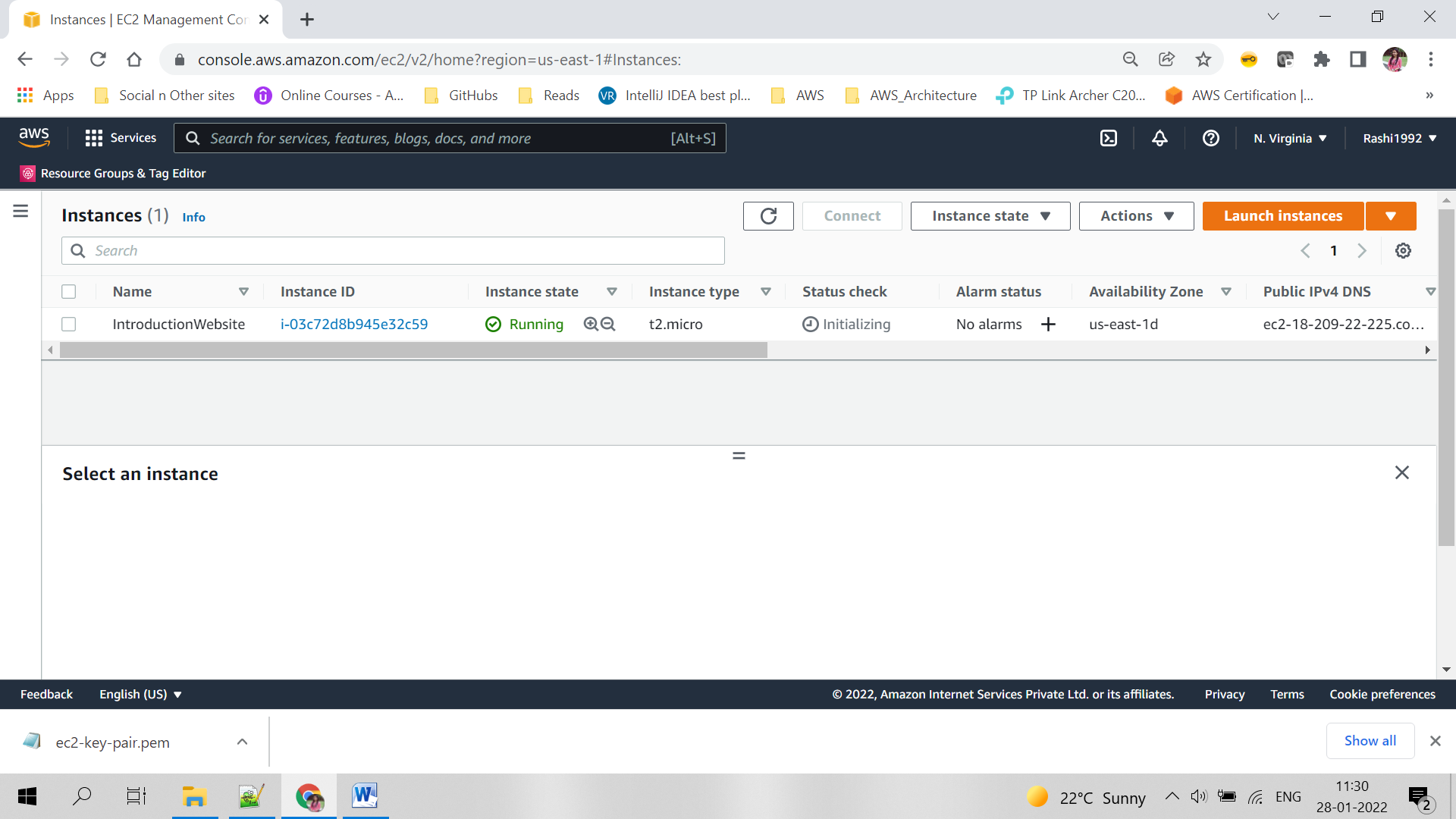
Use existing security group or create a new one. Add http for TCP protocol for opening the port 80



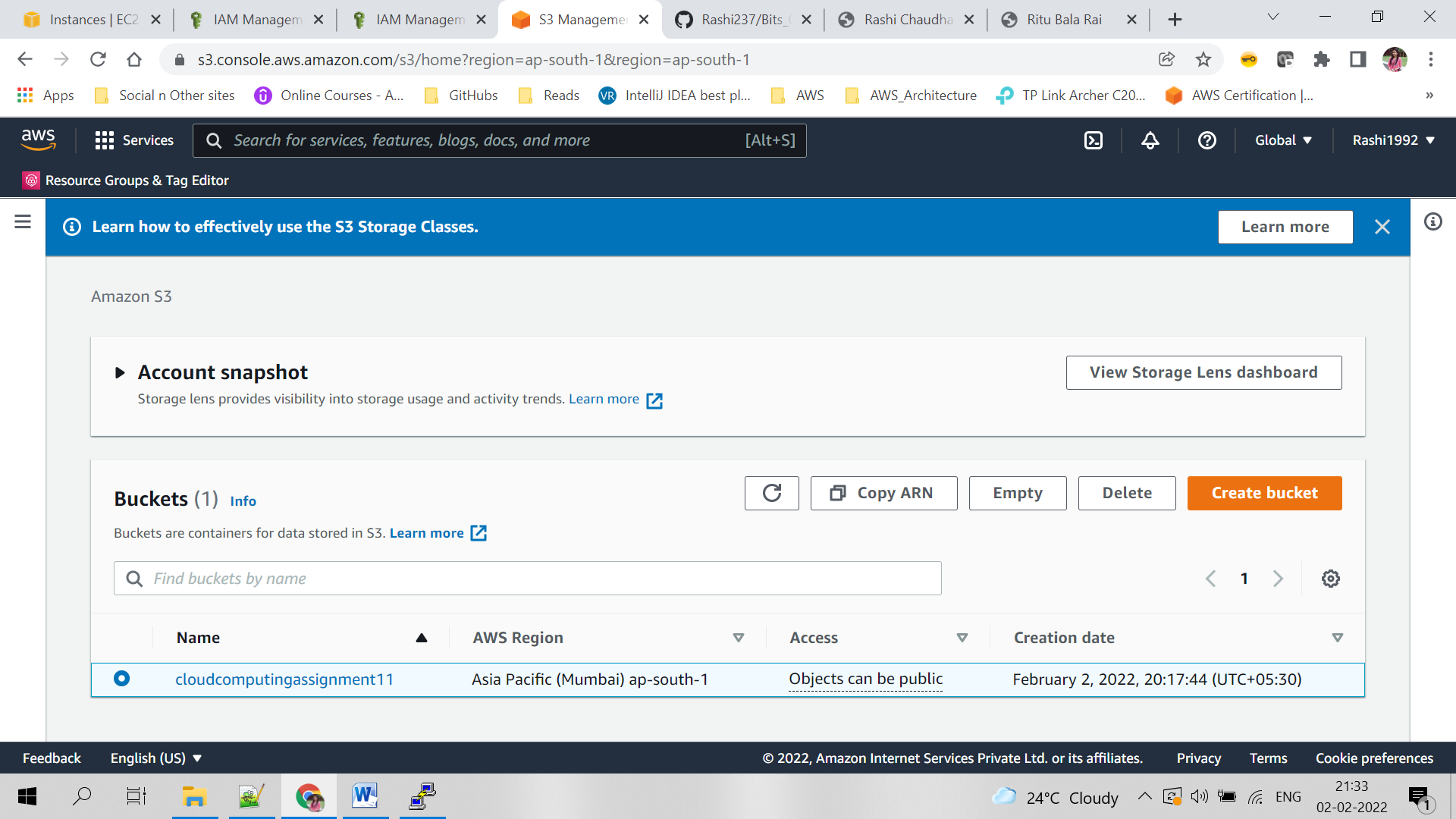
Select an existing key-value pair or add a new one. Download it and keep it safe, will be useful for SSH into the instance.



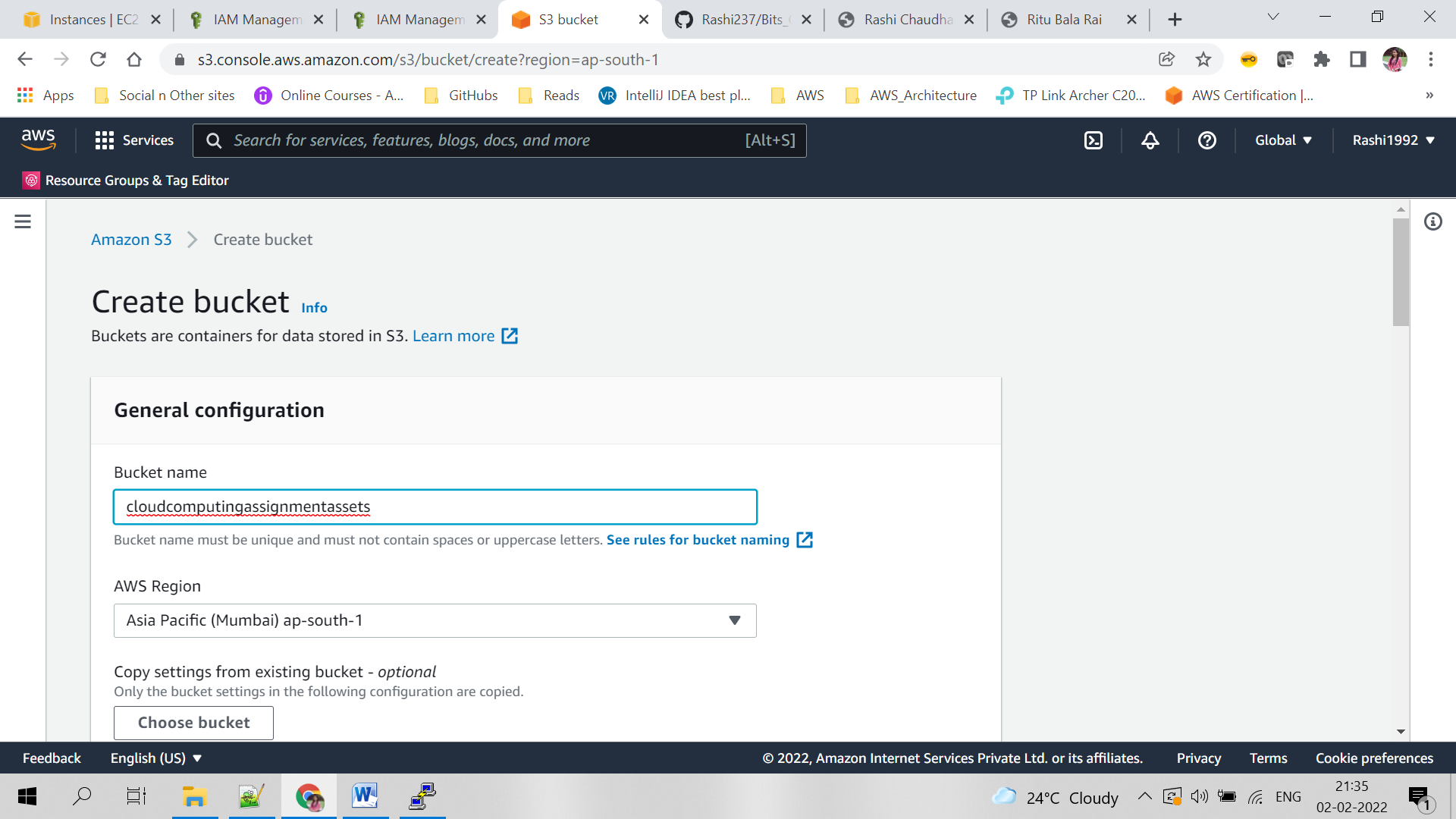




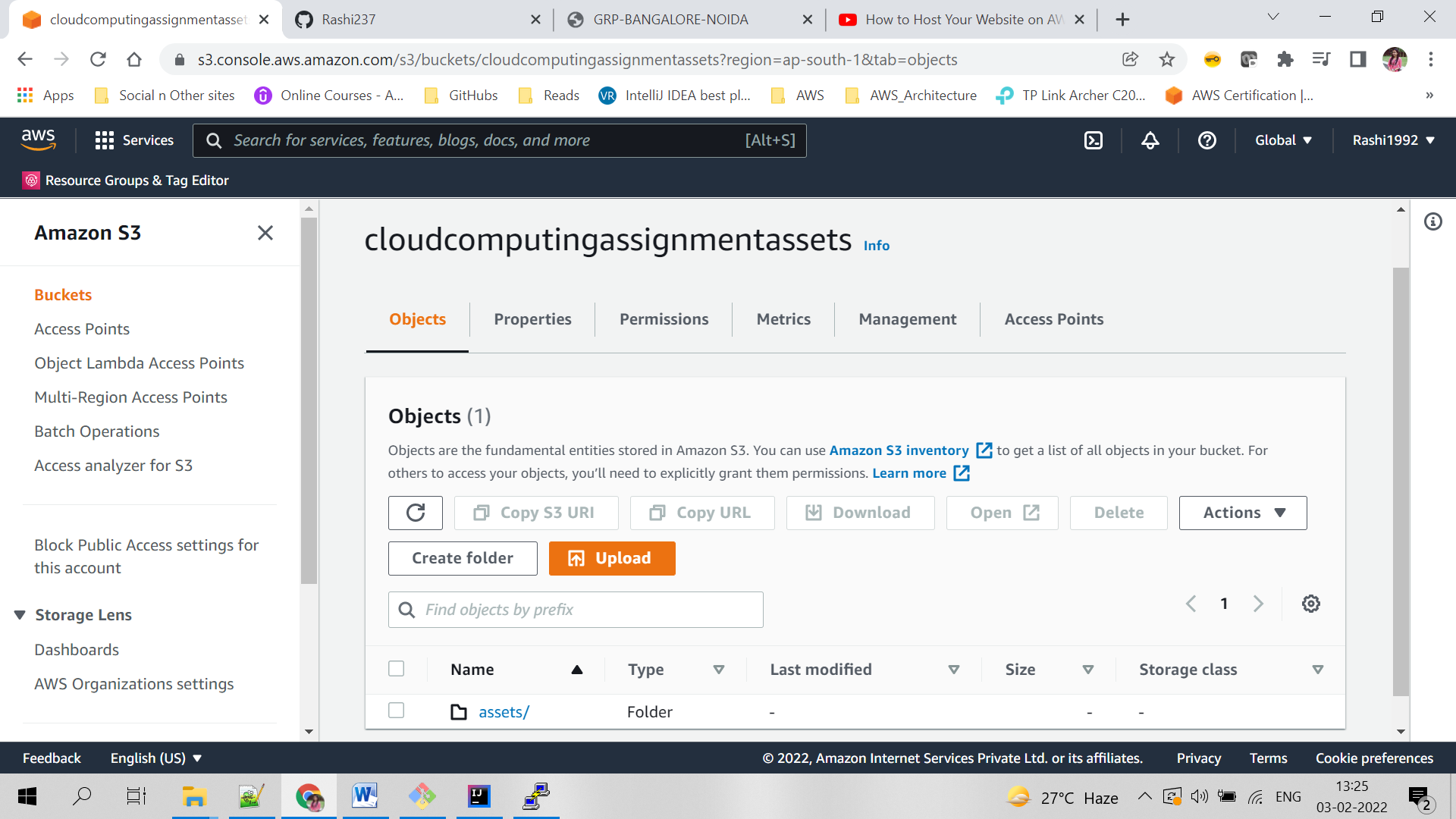
2. Steps for creating S3 bucket



Click on create bucket and give the unique bucket name as shown below leave rest as default

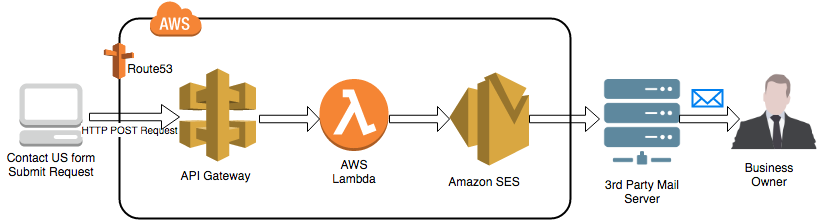


Once bucket is created, upload all the objects in S3.



Contact Form Email using AWS SES

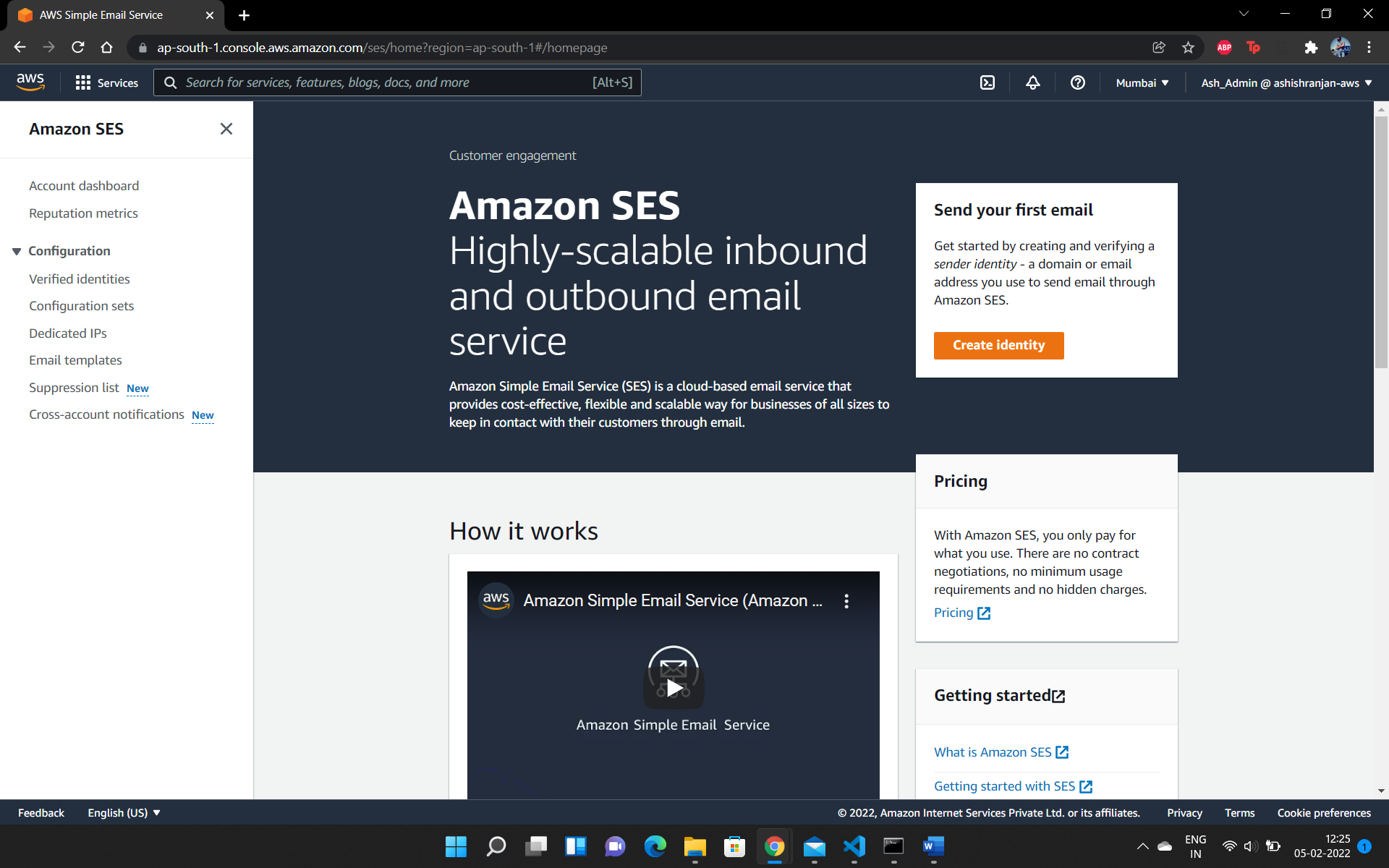
The ‘Contact Me’ section in each of the profile pages contains a form using which the user can contact the respective person through email. This functionality has been implemented using API Gateway, AWS Lambda & Amazon SES.



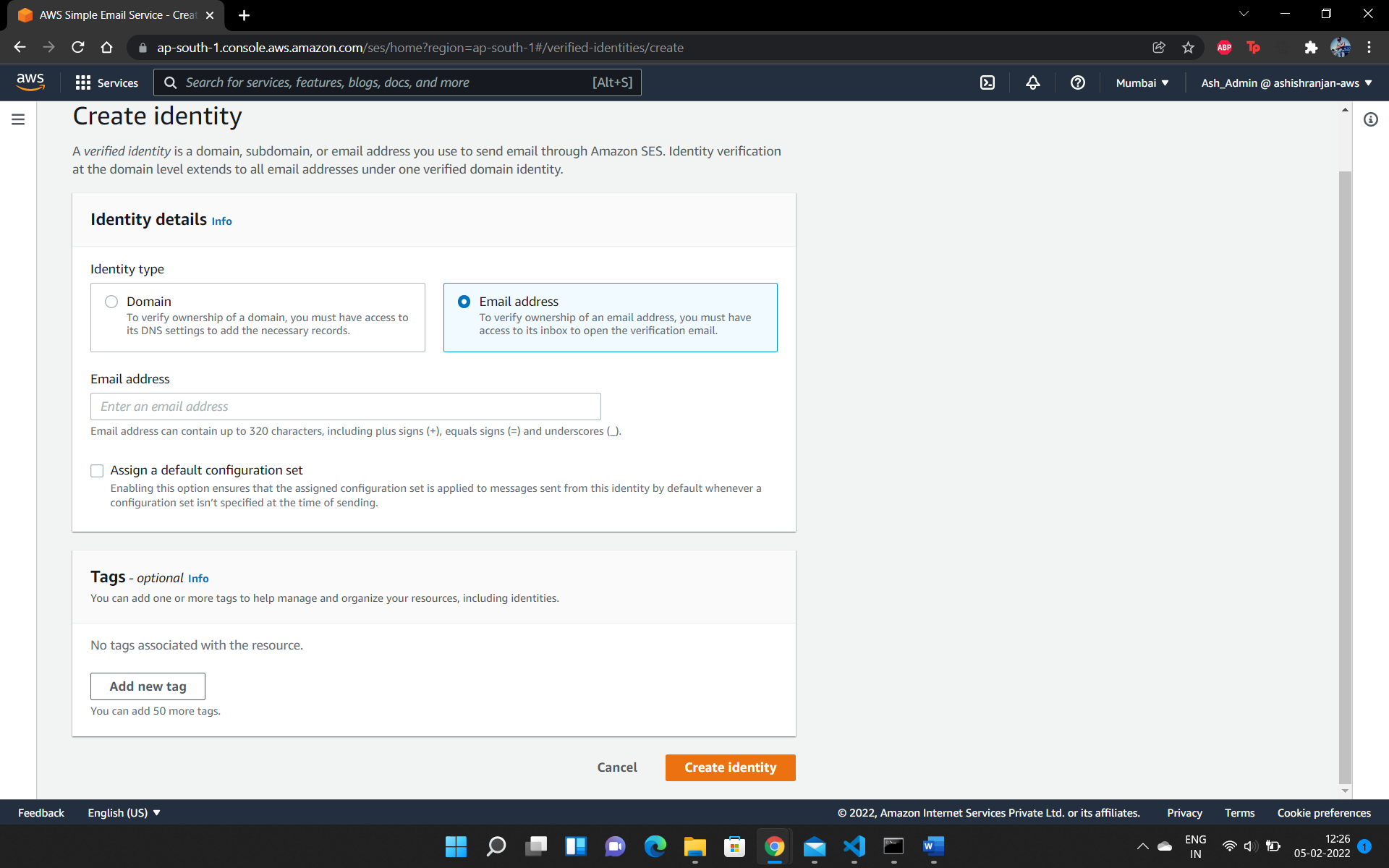
This demonstrates the power of cloud and AWS. We have built a static website and as we know, email functionality is usually implemented in the backend. However, because of AWS, we can add an AJAX call to API gateway which executes a Lambda function to send email using SES. Also, having a serverless API means we are not charged for keeping a server running when we are not using it but instead only being charged when we hit the API endpoint. Below are the steps for implementing the required functionality on AWS.

1. Steps for creating Amazon SES identity

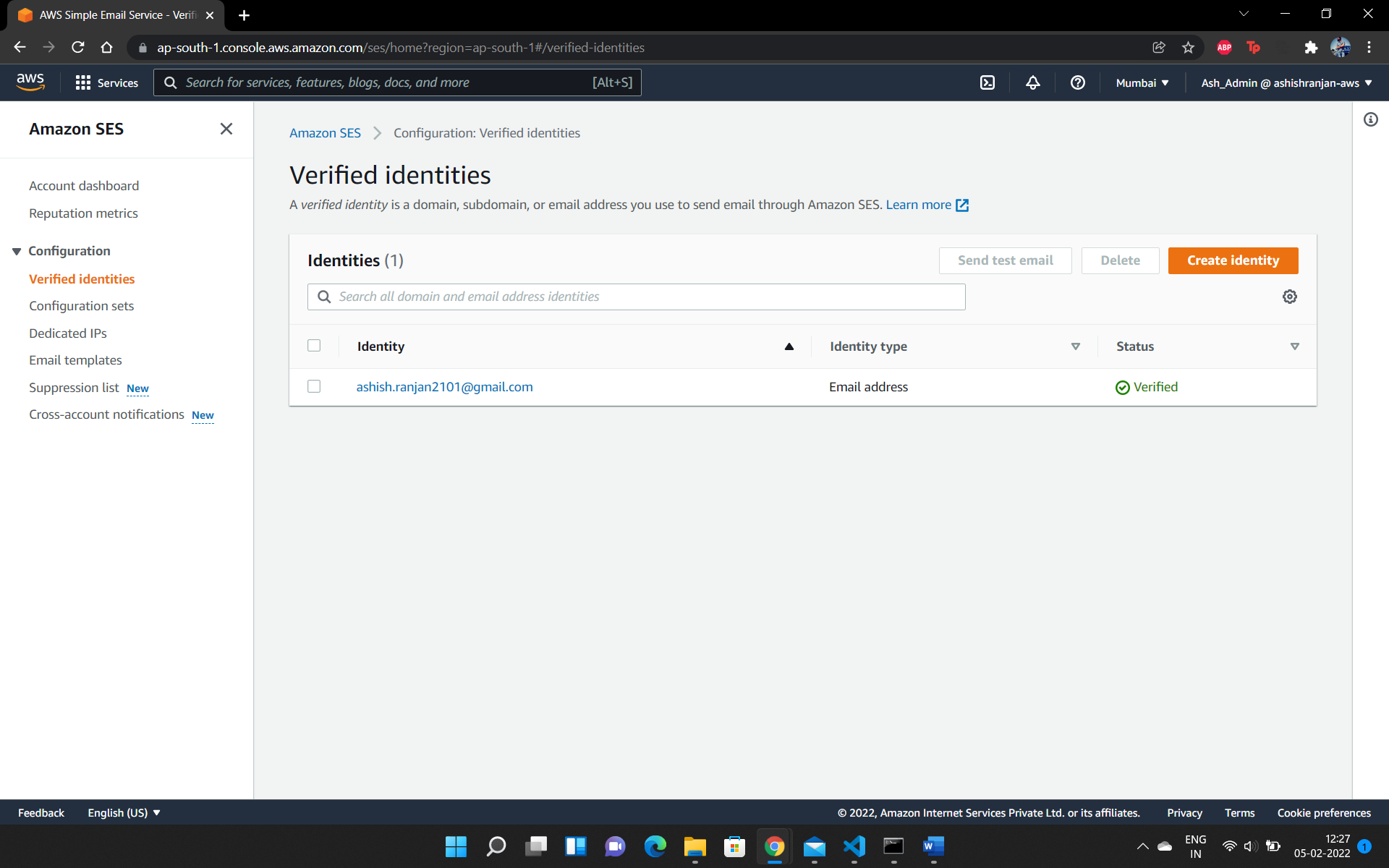
Open ‘Amazon Simple Email Service’ in AWS console and click on ‘Create Identity.’



Select ‘Email address’, enter the email address which will be user to send the emails and click on ‘Create Identity’.

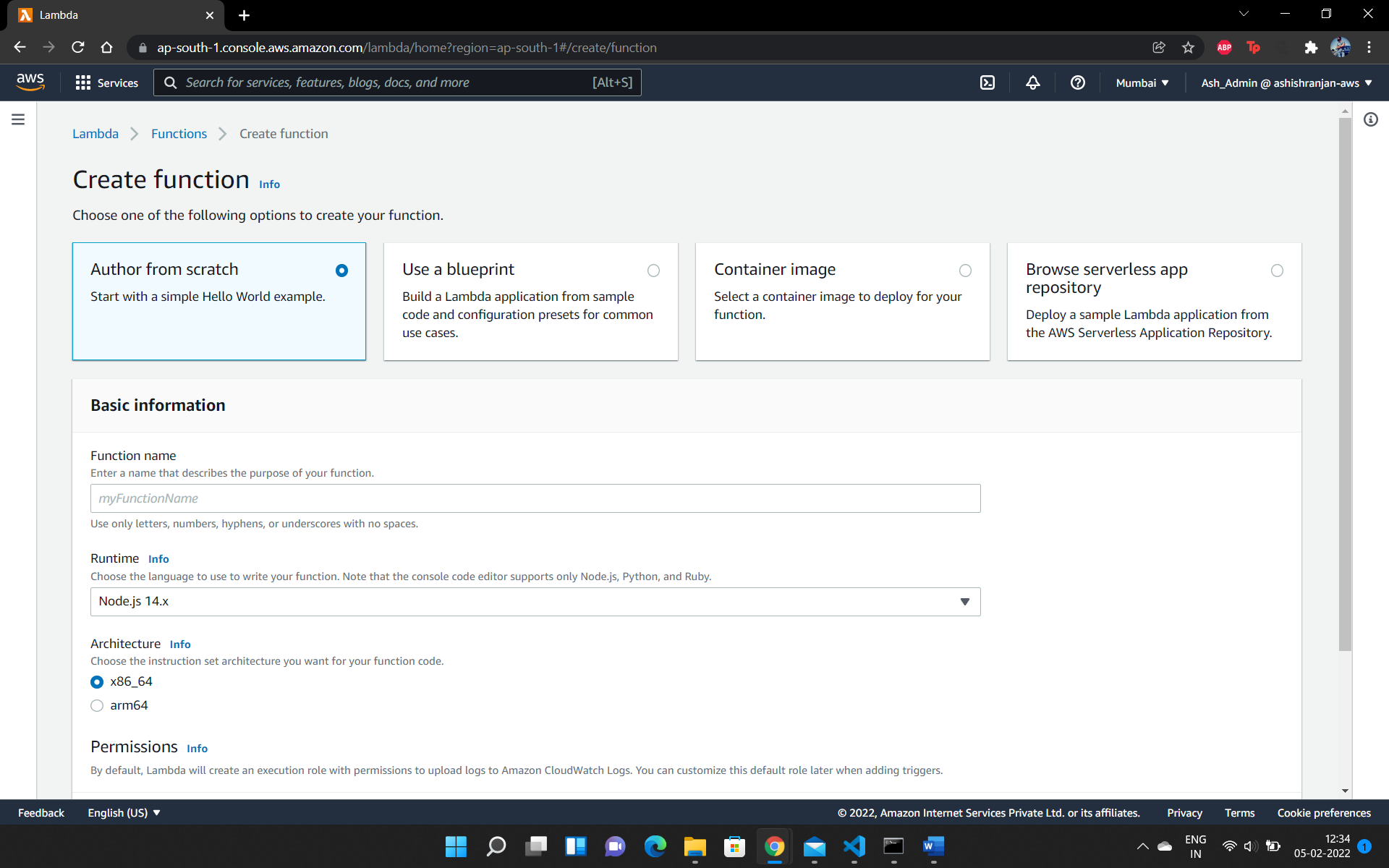


You will receive an email on the email address to verify the identity. Once it is done, you can see the email address listed in ‘Verified identities.’

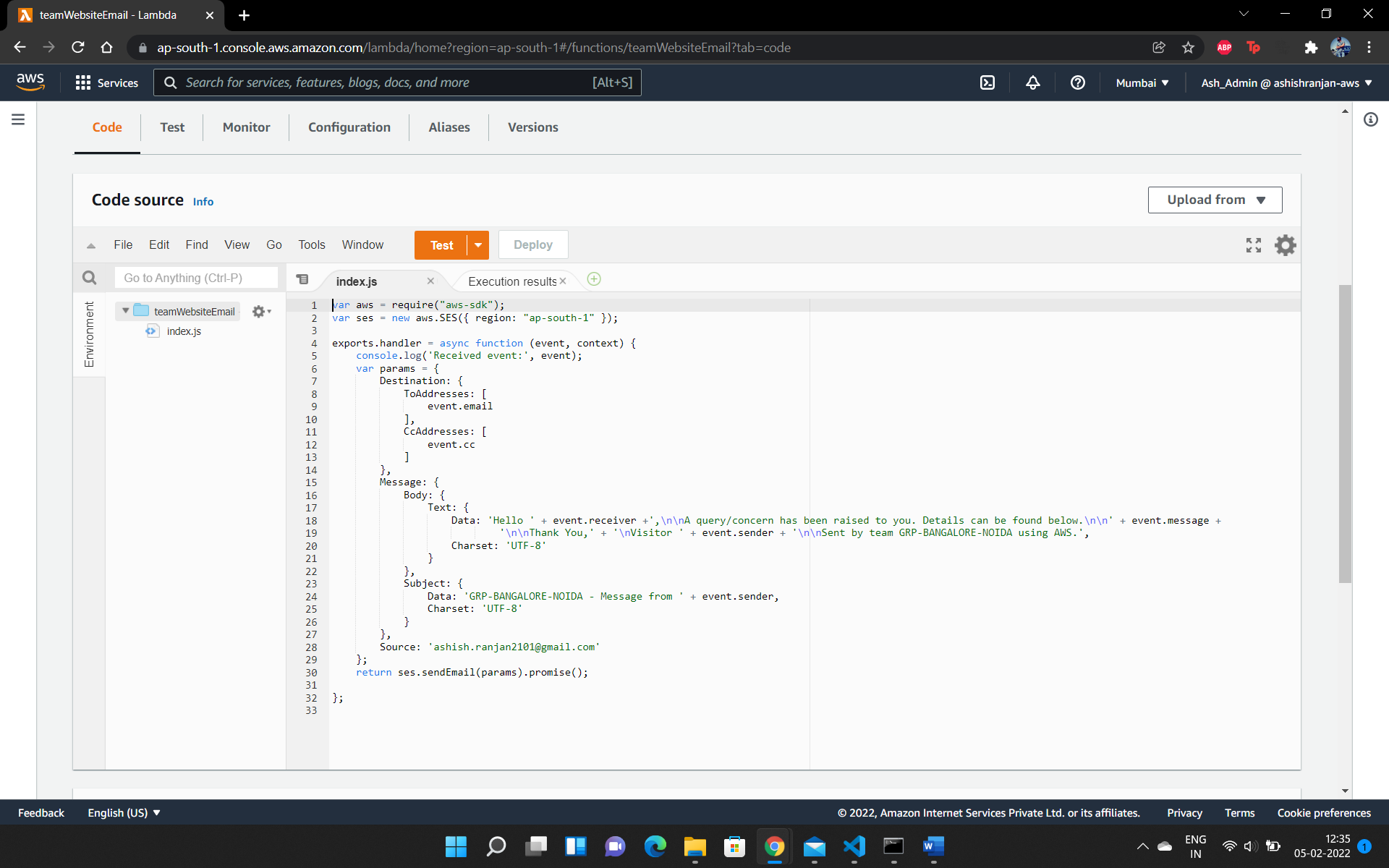


2. Steps for creating Lambda function

Open AWS Lambda in AWS console and click on ‘Create Function.’

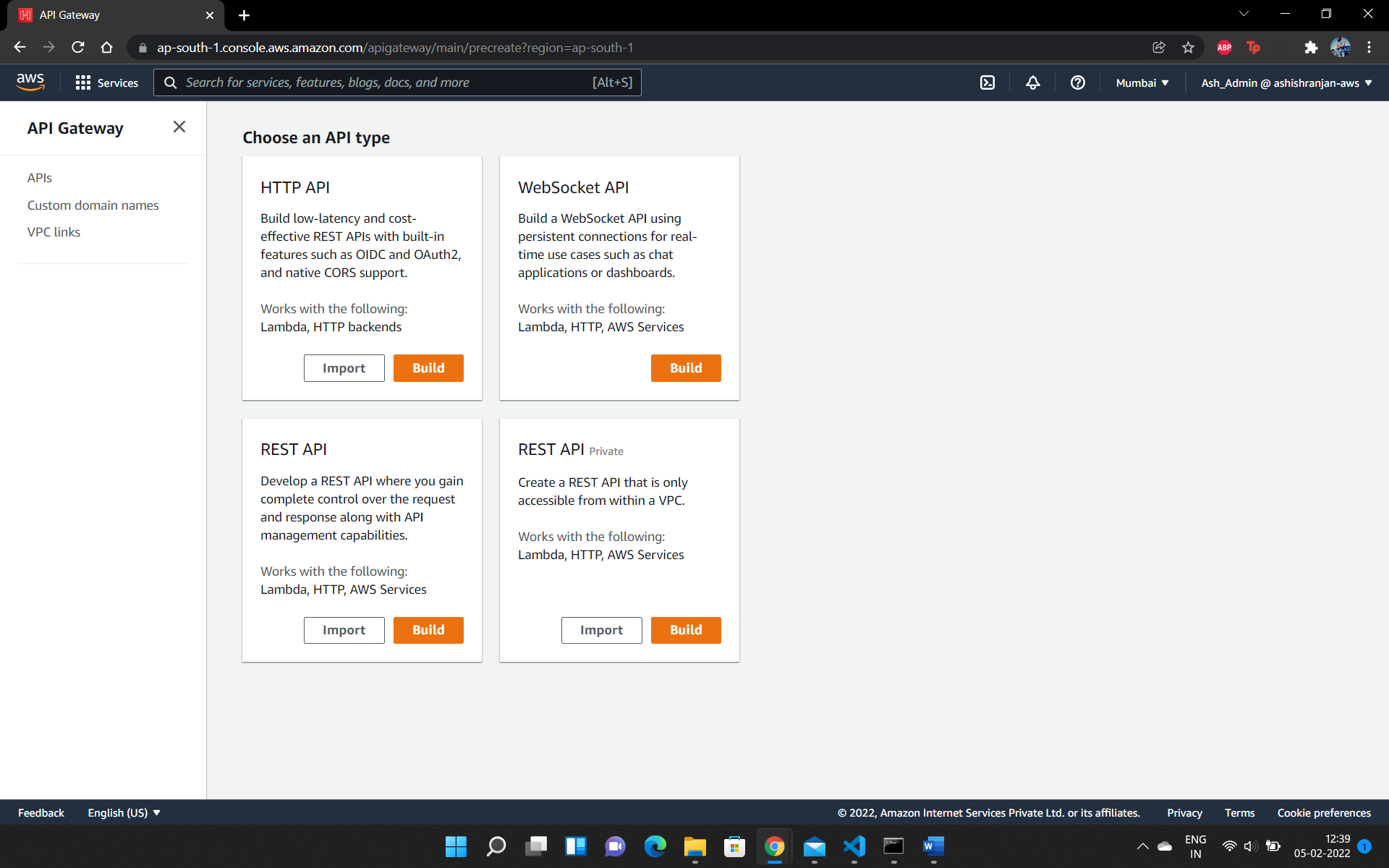


In the Lambda function, enter the below code to call SES and send the mail

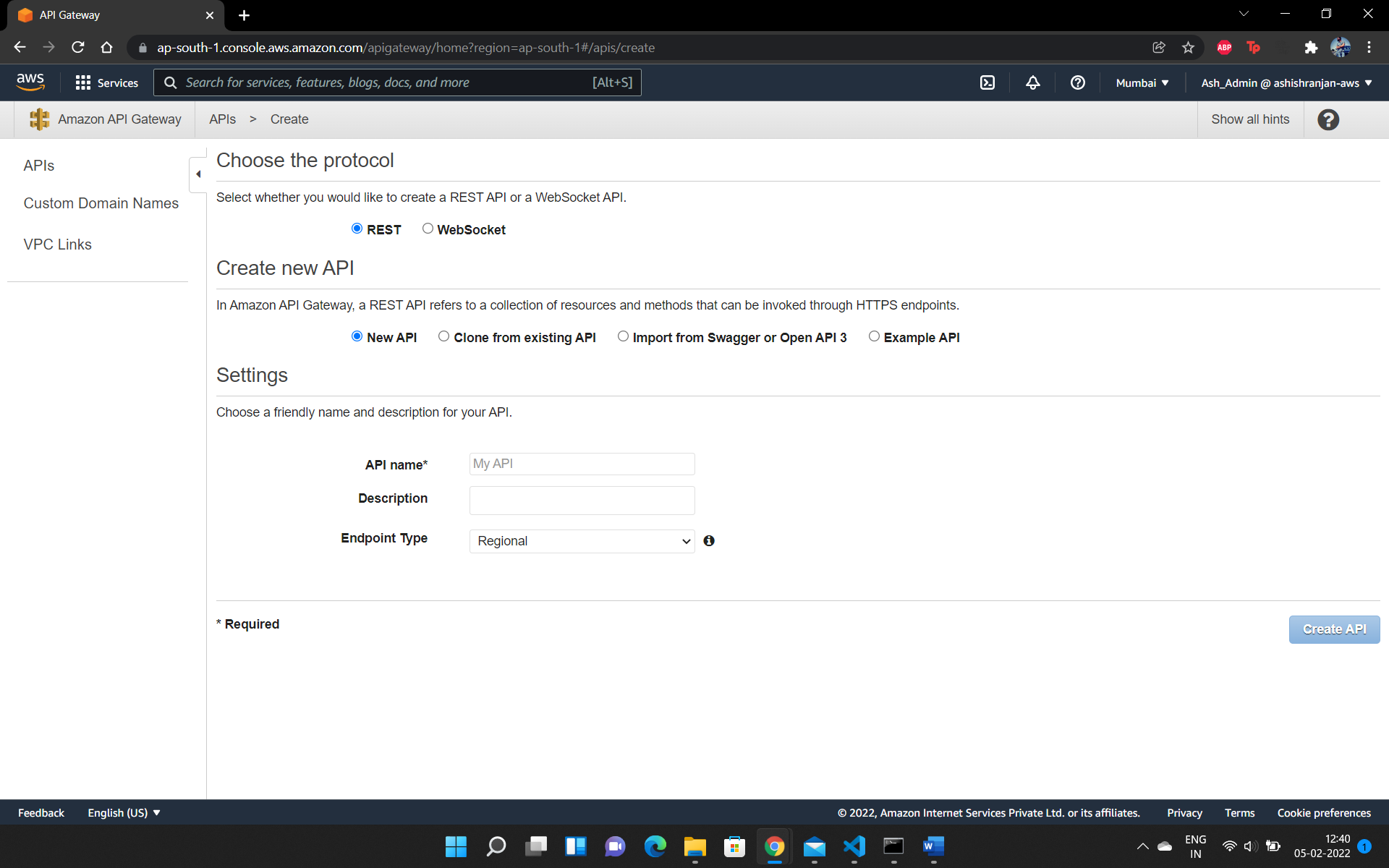


3. Steps for creating API Gateway

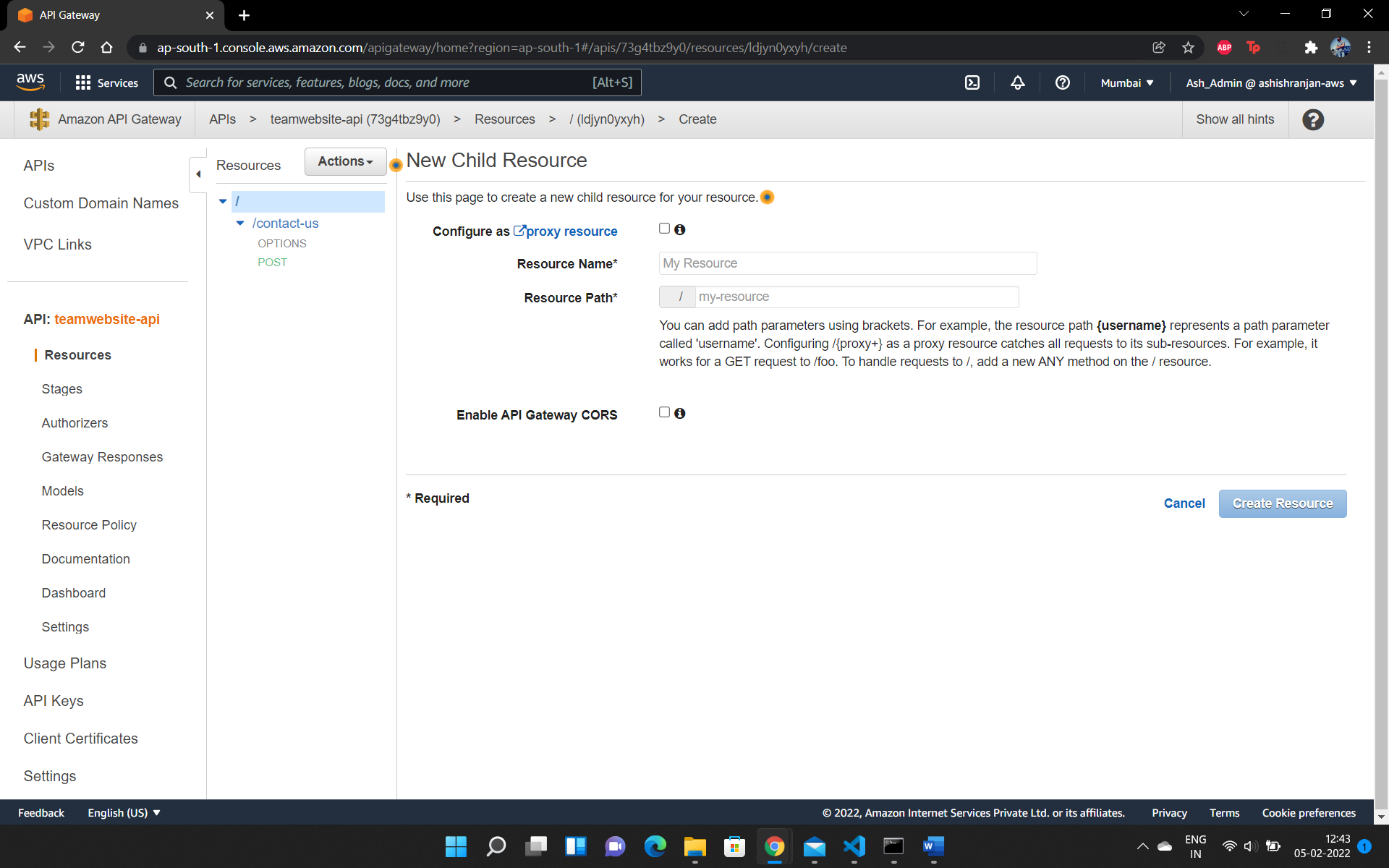
Open API Gateway in AWS console and click on ‘Create API.’



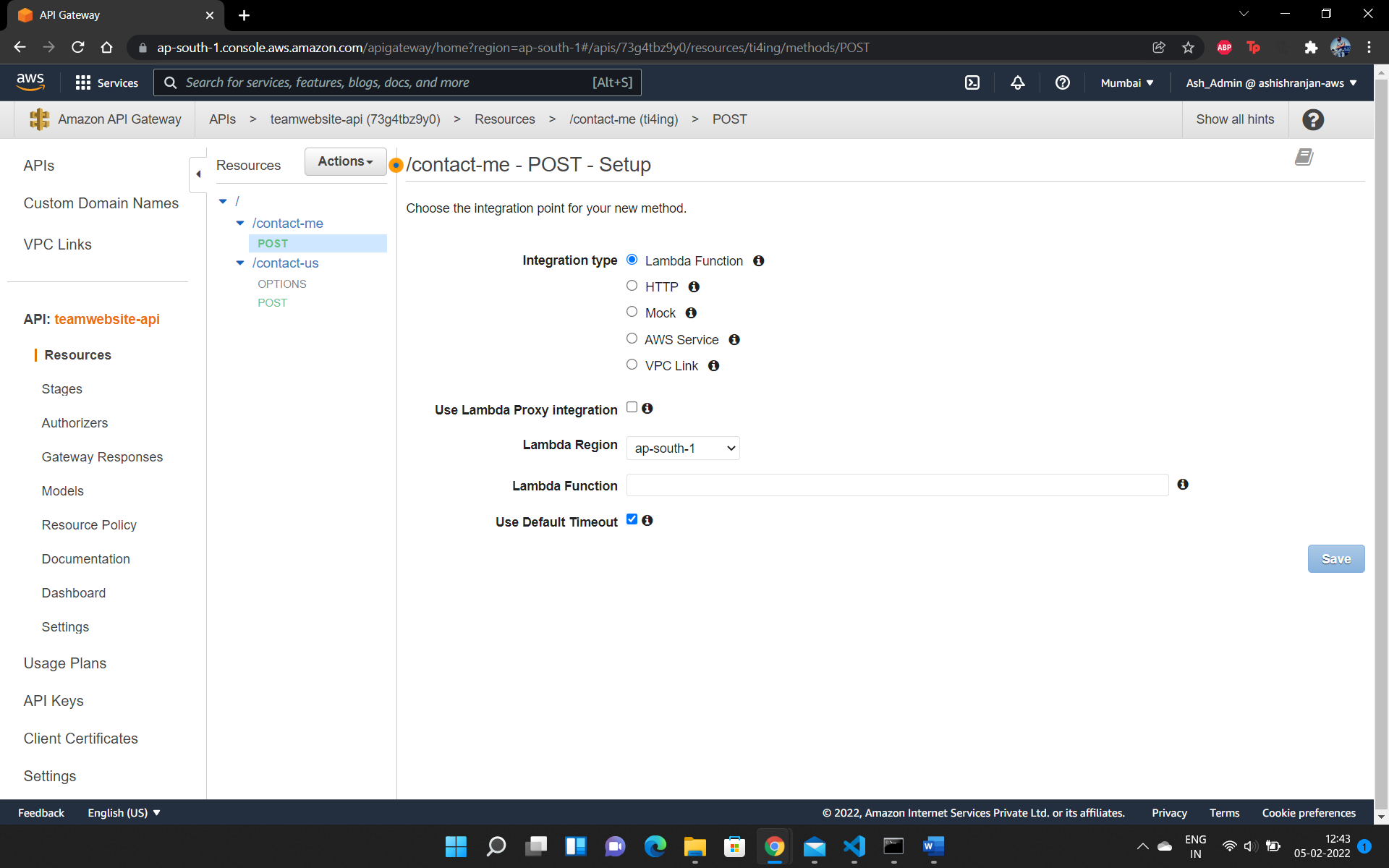
Enter the API name and click on ‘Create API.’



Open the newly created API and create a new resource with the required endpoint.



Create a new ‘POST’ method in the new resource and choose the Lambda function created in the previous step.



Go to Actions -> Deploy API to deploy the API. Then, go to ‘Stages’ to get the invoke URL which can be added in the AJAX call on the website to send the mail.

