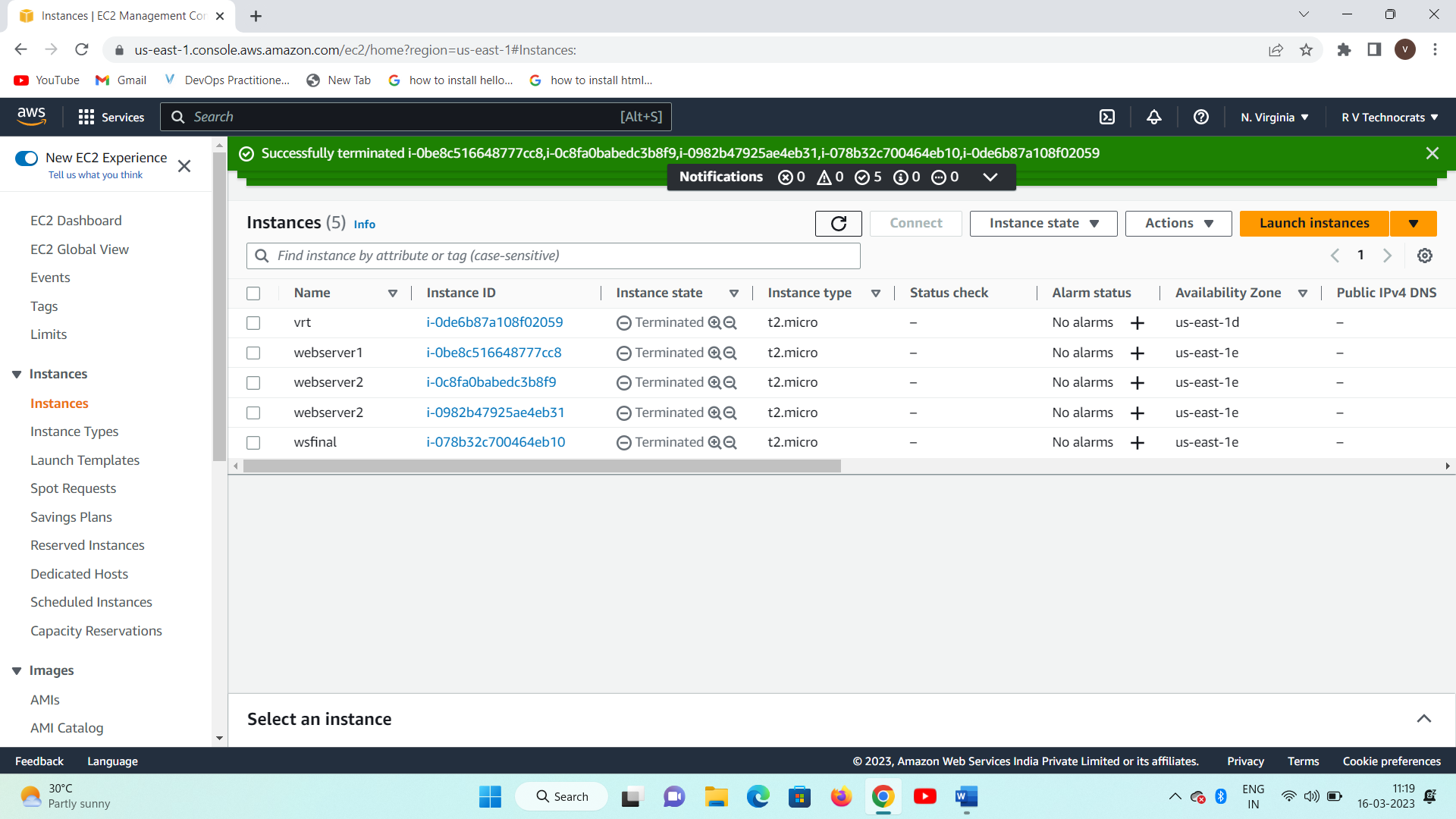
LAUNCHING A WEBSITE ON EC2

Steps:

1. Choose your region
2. search vpc-->select default vpc
3. Then search for security group🡪in security groups open http(80) and ssh(22) ports🡪save.
4. Launch ec2 server [select existing security group🡪provide our security group]
5. Goto subnet🡪choose any subnet (e).
6. Launch the instance.
7. Copy public ip
8. Open putty and connect with instance public ip.
9. Install the s/w that your website needs
10. Put the files in the HTML directory.
11. Start the apache server

Step 1:

Choose your region. (top right side corner of the console choose your region eg. N Virginio)





Step 2:

Creating security groups:

1. Search vpc
2. Scroll down to security group
3. Create your own security group name (delete selected vpc and select default)
4. Edit inbound rules
5. Add 2 rules 1. http (80) 2. Ssh (22)

Step 3:

Launch instance:

1. Instance name
2. Key pair
3. Network settings 🡪security group 🡪 choose your own security group
4. Launch

Step 4:

Select ec2🡪details🡪 copy ip address

Open putty

Provide public ip and choose auth 🡪credentials 🡪 select keypair

Step 5:

Install the s/w that we need on ec2

Following commands need to run to complete the hosting

1. Sudo su
2. Yum install -y httpd
3. Cd /var/www/html
4. Vi index.html
5. Press I(i🡪to enter into insert mode)
6. Type your programd
7. Esc
8. Shift+colon+w+q
9. Systemctl update
10. Systemctl enable httpd ( activates the middle tier to launch our website )
11. Systemctl start httpd

\*index.html is the startupfile that required to launch website

**Share the files and folders from vm to vm**

1. Goto desktop create a folder ( shared folder)
2. Goto folder and right click 🡪properties🡪sharing🡪share🡪everyone🡪r/w permissions🡪share.
3. Goto vm
4. Devices🡪insert guest addons🡪you are getting a notification🡪click on the notification and run the addons to be installed. 🡪restart
5. After reboot 🡪 goto devices🡪 shared folders 🡪 add share🡪select shared folders 🡪browse for folder
6. Copy and paste or drag and drop from one vm to another. Or host.

**CLI based installations:**

Google provides sdk to launch your applications, but it works for lower versions. To launch the app use python2.7 and follow the steps

1. Create a folder on the desktop or any where
2. Create a python file with the following content .

print(“Hello World”)

(save this file as index.py)

1. Create a yaml file with the following content.

Runtime: python27

Api\_version:1

Threadsafe : false

Handlers:

-url:/

Script:index.py

(save this file as app.yaml)

**. Develop a Serverless Web App using AWS**

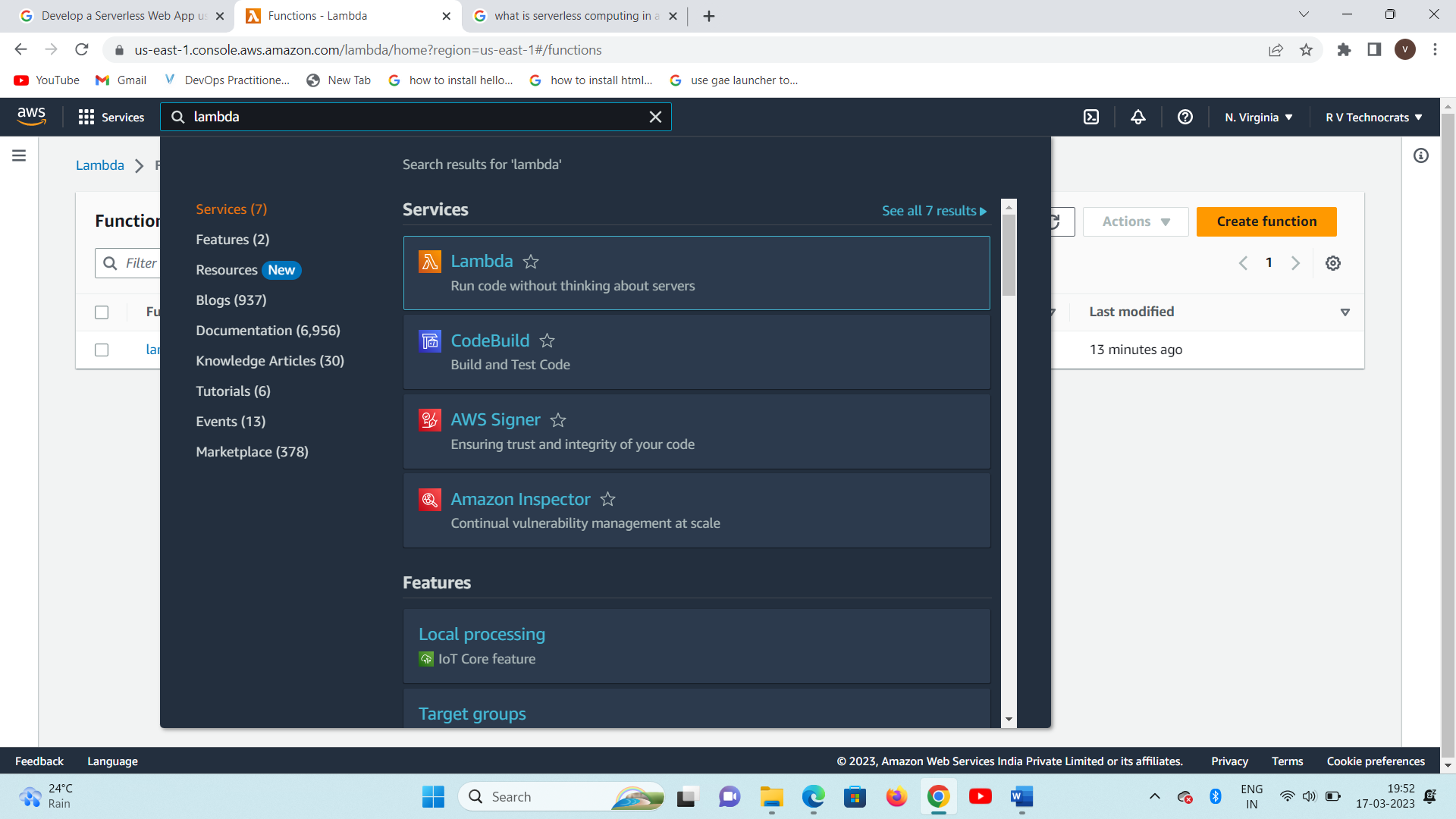
A serverless architecture is **a way to build and run applications and services without having to manage infrastructure**. Your application still runs on servers, but all the server management is done by AWS.

Overview. Serverless is **a cloud-native development model that allows developers to build and run applications without having to manage servers**. There are still servers in serverless, but they are abstracted away from app development.

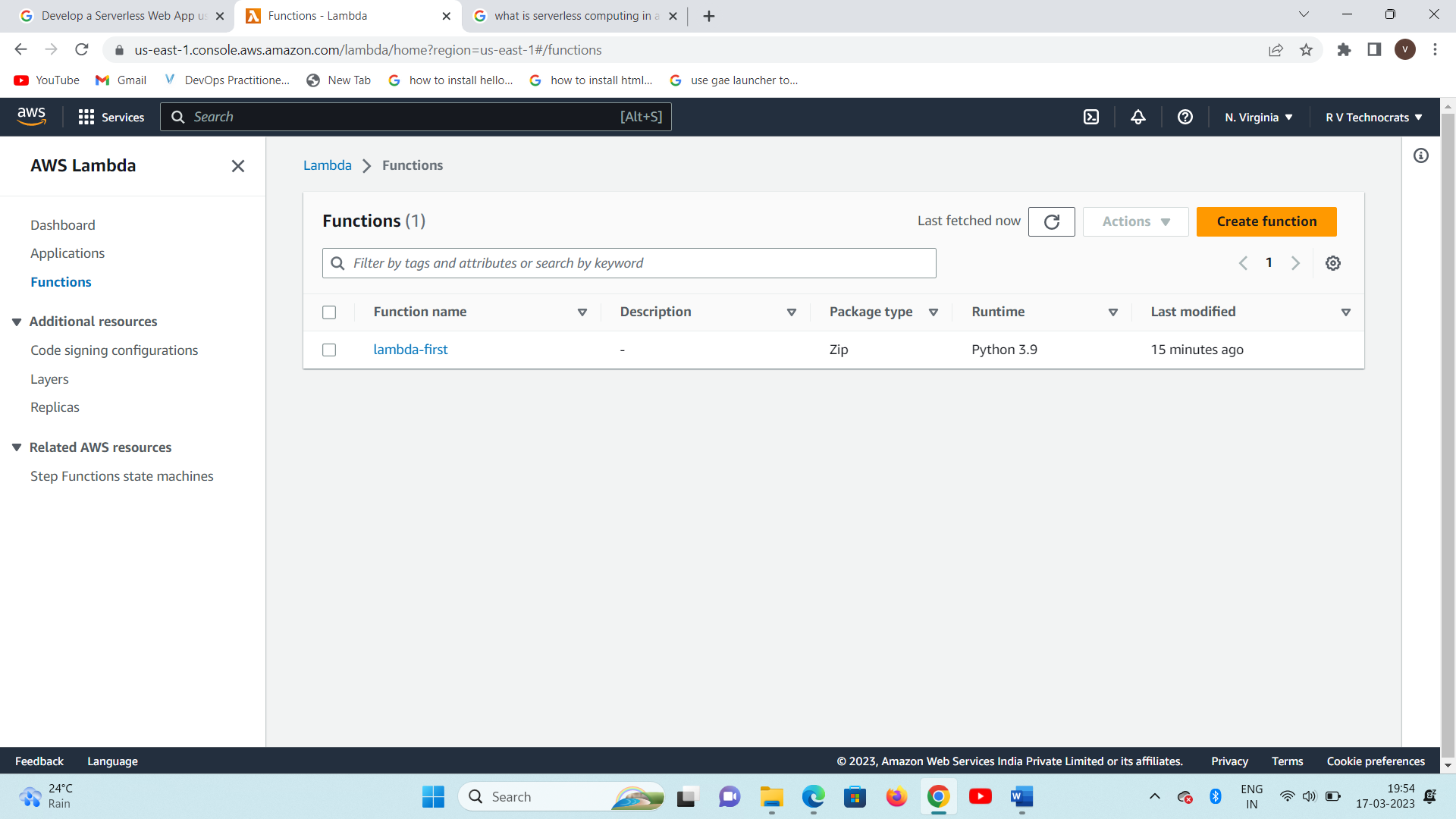
Aws lambda is the service that provides serverless architecture

Development process

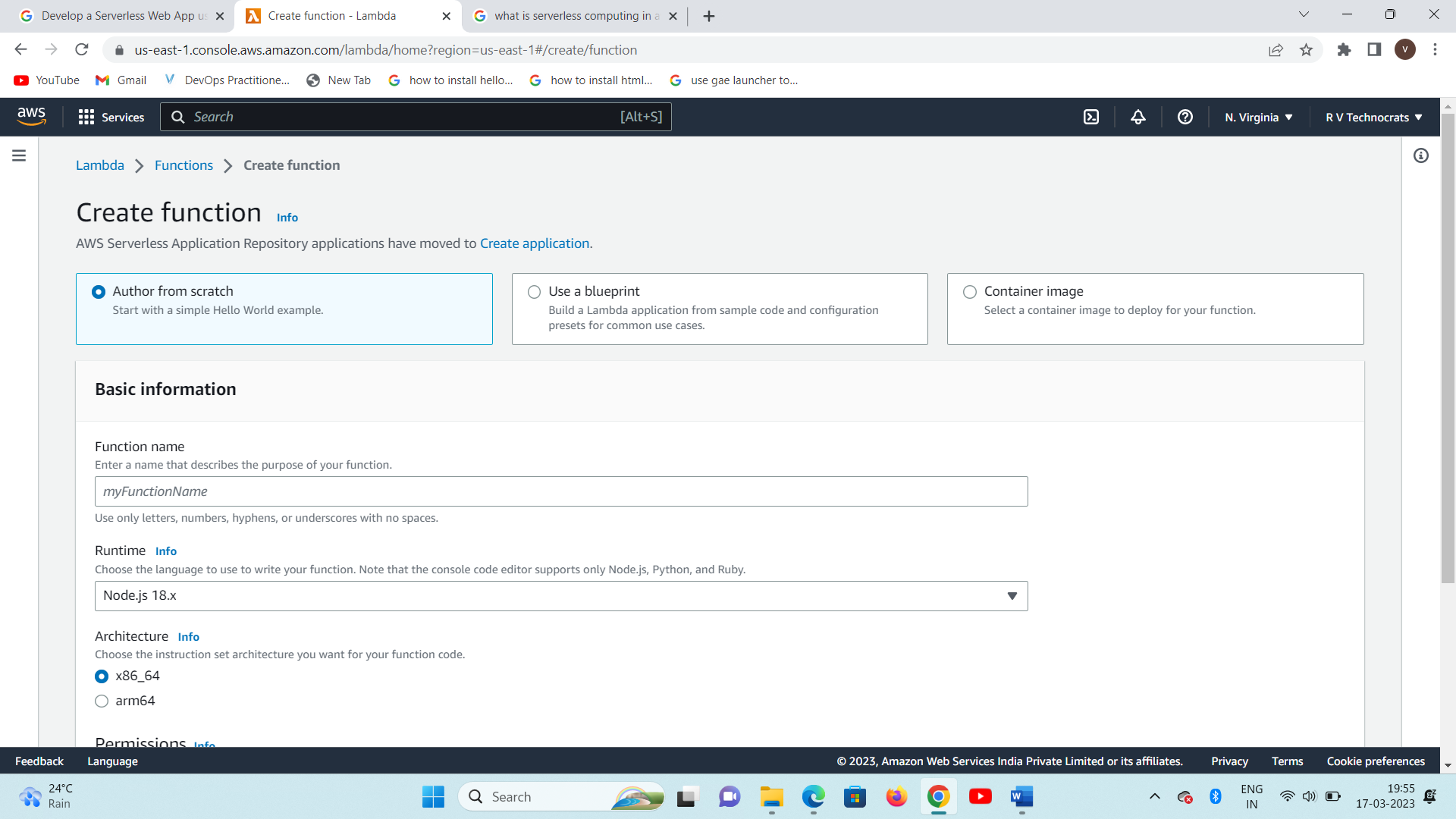
Step 1: search lambda in aws console



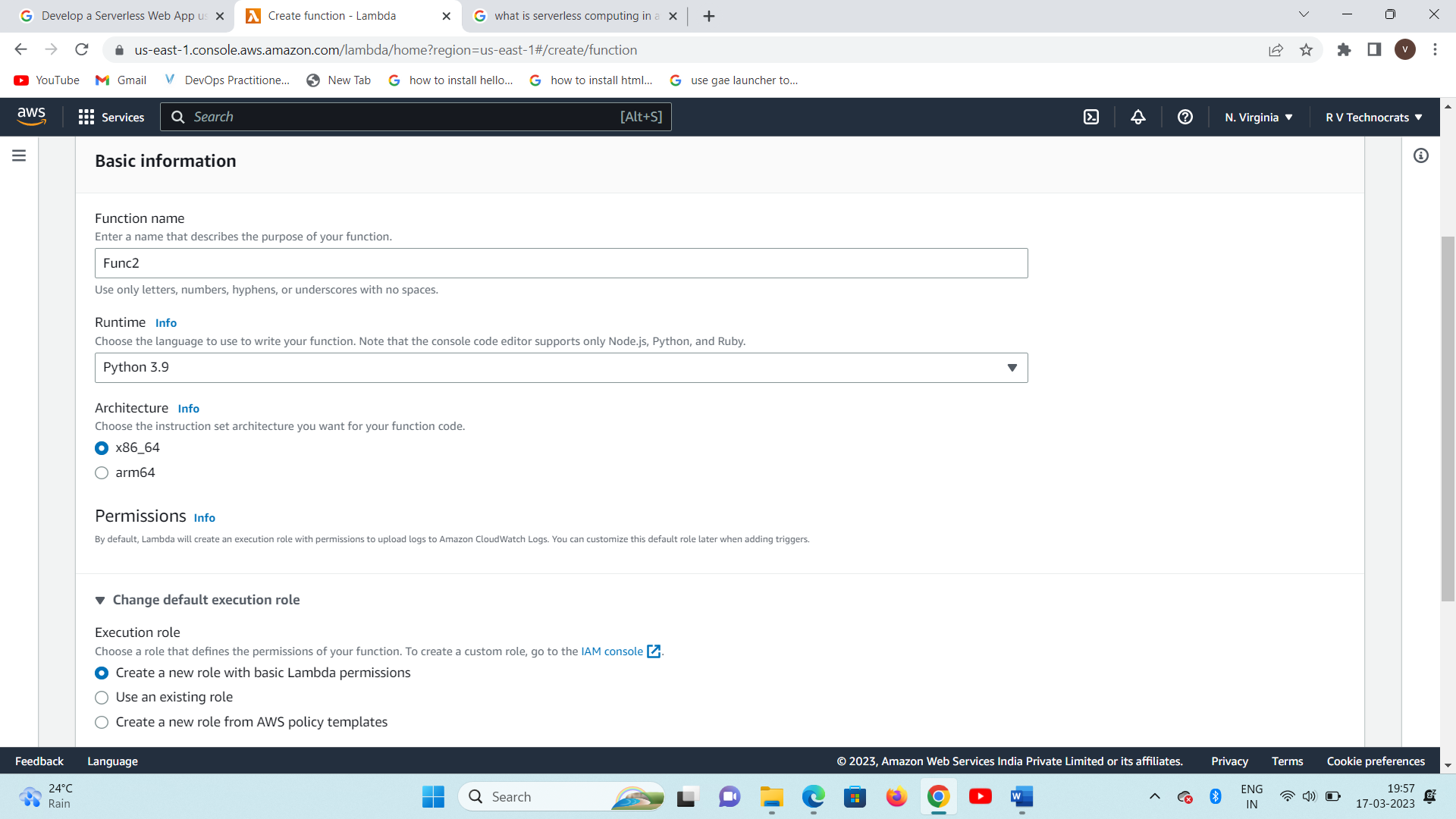
**Step2: Create Function**



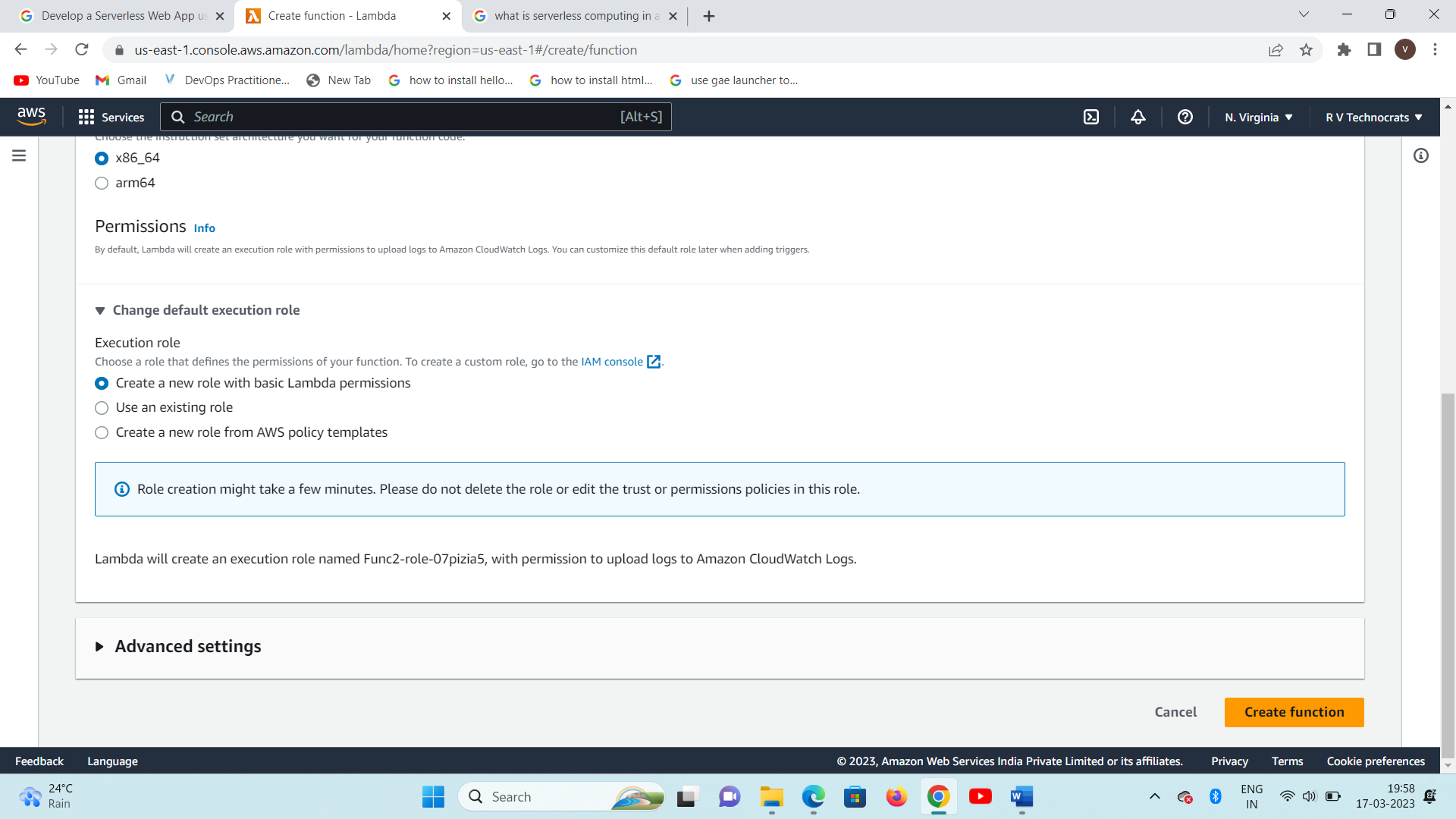
**Step 3: choose author from scratch**



**Step 4: provide functions name and choose run time as python or equivalent**

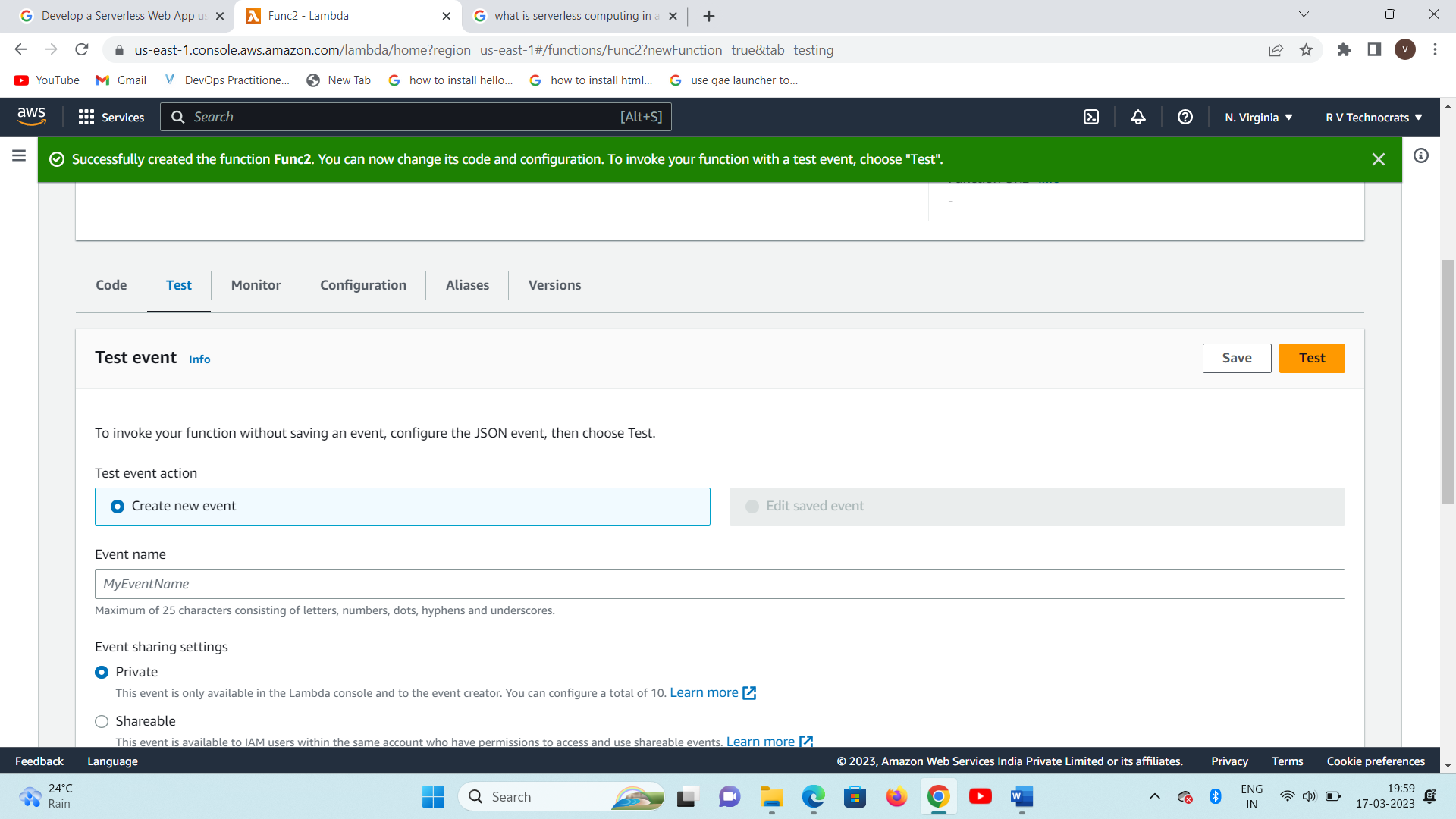


**Choose the default role**



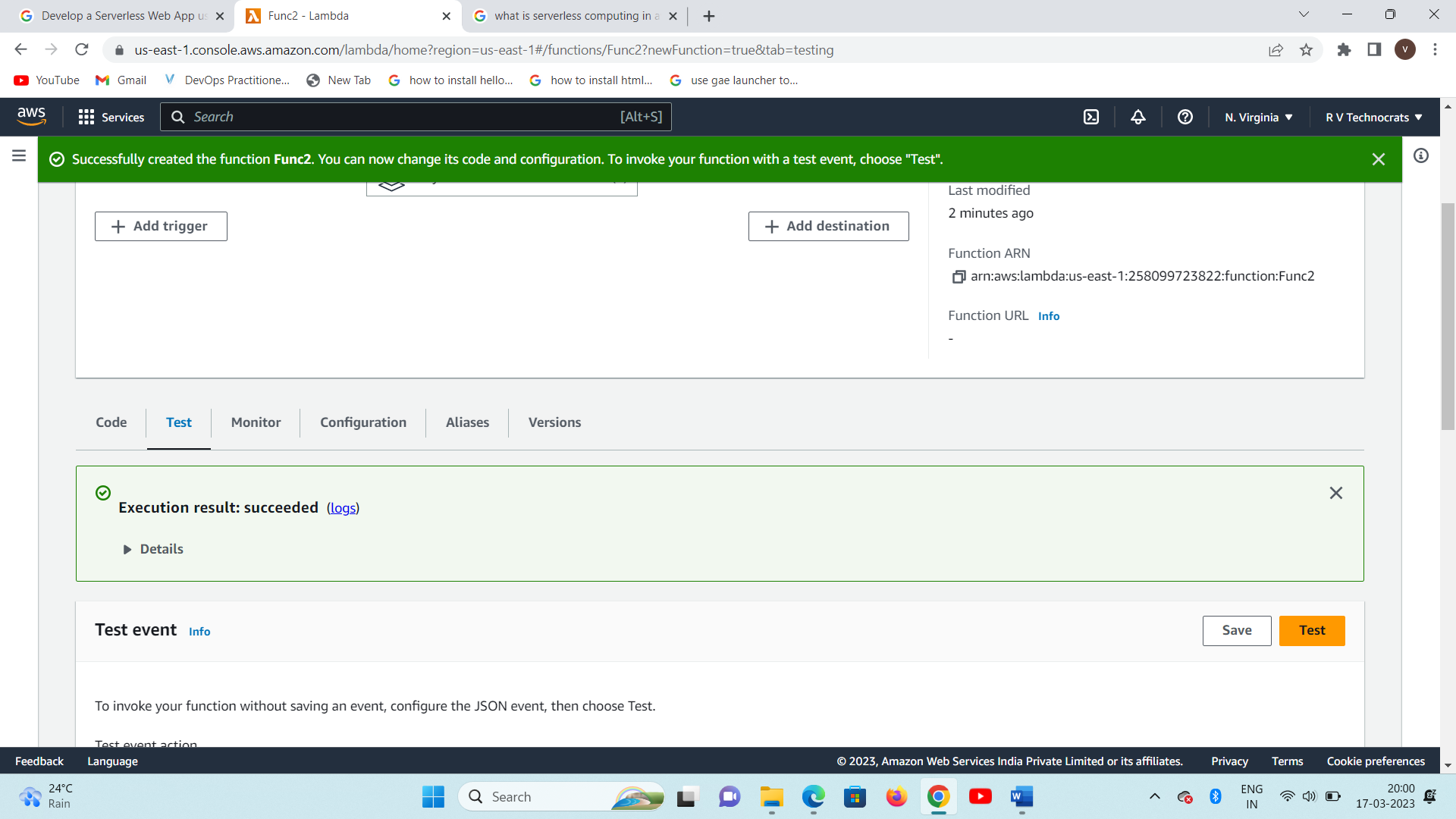
**Create function**

**Next : select test**

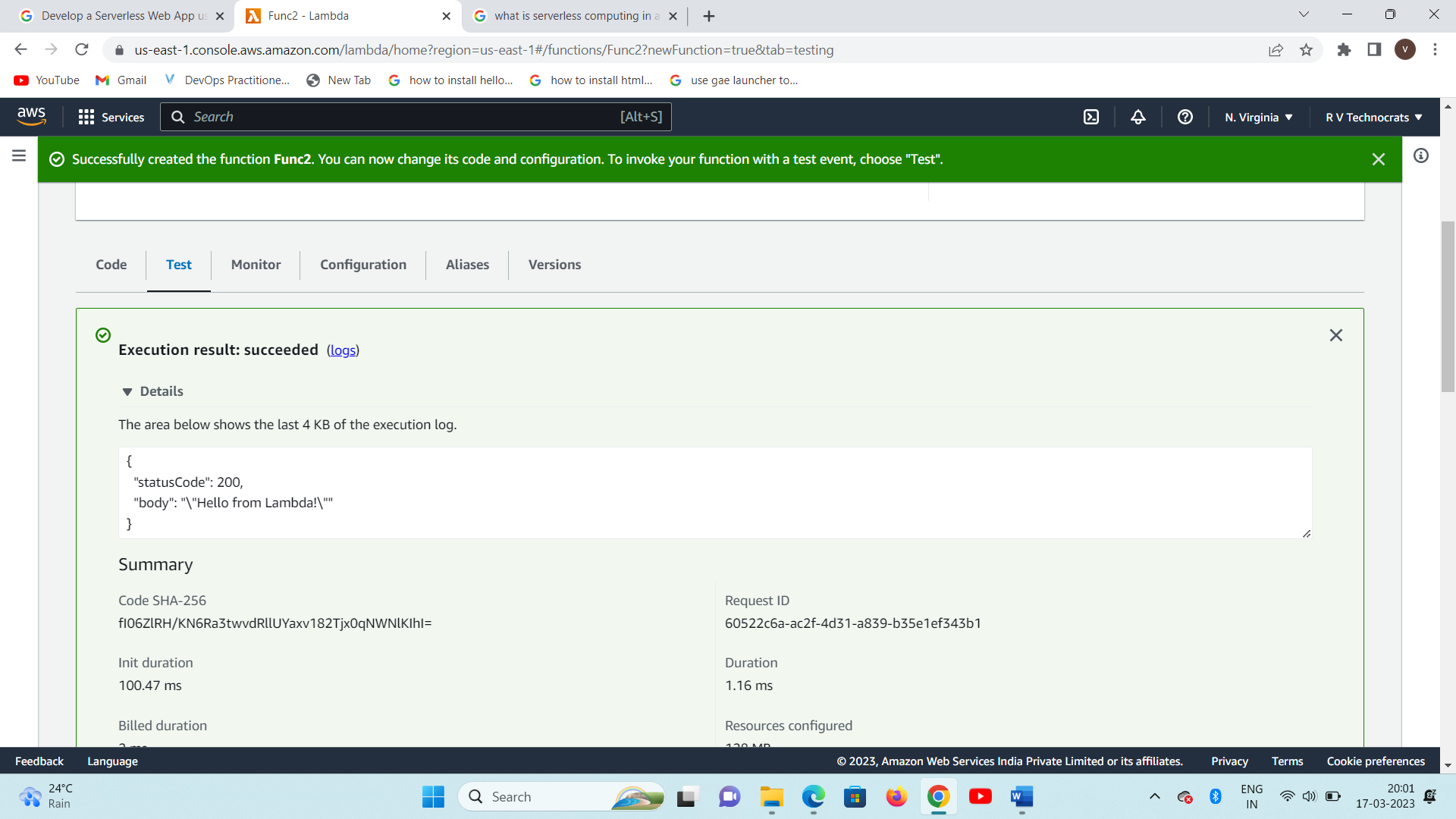


**Click on test**

**The test results will be displayed**



**The test result is 200 defines the success of lambda function**



🡪App Engine java guest book:

Original Google App Engine Java sample, created in 2009, supporting Google App Engine Standard with Java8, Java11, and Java17 using Bundled Services like App Engine Datastore API,and App Engine Users API. By changine the field in the appengine-web.xml, you can select with Java version to use to execute this Java EE Web App, using Servlet and JSP. It demonstrates that code written back in 2009 with Google App Engine can still serve today with different JVMs.

## Prerequisites

### Download Maven

These samples use the [Apache Maven](https://maven.apache.org/) build system. Before getting started, be sure to [download](https://maven.apache.org/download.cgi) and [install](https://maven.apache.org/install.html) it. When you use Maven as described here, it will automatically download the needed client libraries.

### Create a Project in the Google Cloud Platform Console

If you haven't already created a project, create one now. Projects enable you to manage all Google Cloud Platform resources for your app, including deployment, access control, billing, and services.

1. Open the [Cloud Platform Console](https://console.cloud.google.com/).
2. In the drop-down menu at the top, select **Create a project**.
3. Give your project a name.
4. Make a note of the project ID, which might be different from the project name. The project ID is used in commands and in configurations.

### Setup

Use either:

* gcloud init
* gcloud auth application-default login

### Google Cloud Shell, Open JDK 11 setup:

🡪To switch to an Open JDK 11 in a Cloud shell session, you can use:

sudo update-alternatives --config java

# And select the usr/lib/jvm/java-11-openjdk-amd64/bin/java version.

# Also, set the JAVA\_HOME variable for Maven to pick the correct JDK:

export JAVA\_HOME=/usr/lib/jvm/java-11-openjdk-amd64

🡪Create a appengine.web.xml file

<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">

<runtime>java11</runtime> <!-- or java17-->

<app-engine-apis>true</app-engine-apis>

</appengine-web-app>

🡪While the Java17 runtime is in Beta, in order to deploy the application, you can use the beta value for the gcloudMode Cloud SDK parameter like:

mvn appengine:deploy -Dapp.deploy.gcloudMode=beta

🡪Deploy the application

## Maven

### Running locally

mvn appengine:run

### Deploying

mvn clean package appengine:deploy -Dapp.deploy.gcloudMode=beta -Ddeploy.projectId=XXXX

code available in zip file.

12. Design a Content Recommendation system using AWS

Follow the steps as first program , and down load the code from github as per the link defines

While running the commands , follow the commands

Following commands need to run to complete the hosting

1. Sudo su
2. Yum install -y httpd
3. Cd /var/www/html
4. Wget <http://github.com/azeezsalu/techmax/archive/refs/heads/main.zip>
5. Unzip main.zip
6. Cp -r techmax-main/\* var/www/html
7. Rm -r techmax-main main.zip
8. Systemctl enable httpd
9. Systemctl start httpd

**15. Design E-learning cloud-based system:**

Methods for Providing Cloud-Based E-Learning System E-learning is cognitive science principles and excellent solution for learners having effective multimedia learning using electronic educational technology. E-learning is a fast, efficient procedure for learning and cost-effective as per the user’s ease as well as organization’s profitability.

Figure 1 shows the framework of e-learning, whereas Fig. 2 shows how cloud-based e-learning works. E-learning has plentiful direct as well as positive effects on the learners :

• Learner’s ratio is increased at national as well as global level.

• It simply assists the learners in learning specific program or subject.

• It improved the long-term retention of data and information.

• Traveling cost is not required. E-learning demand is increasing day by day, and it is not going to vanish in coming years. Some e-learning trends of 2017 are discussed below .

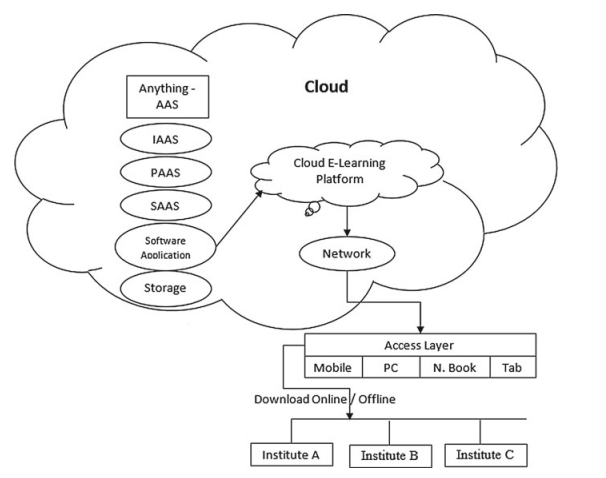


Figure : cloud architecture

**E-Learning Trends Mobile Learning Enhancement:**

In current years, due to the massive increase in the smartphone users, the quality of mobile learning is improving gradually. Every day, the users of smartphones are increasing and they want to facilitate themselves. The users of smartphone do not want to use laptops or desktop PC, to access various applications. As a result, mobile e-learning trend is increasing more rapidly.

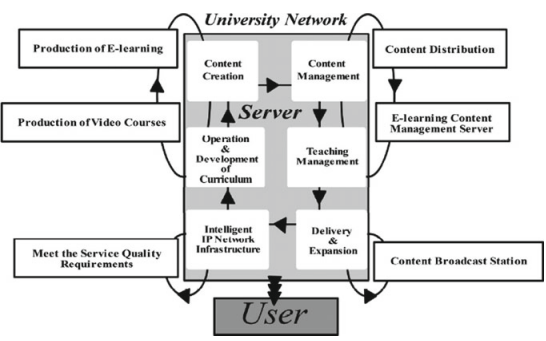


Figure : E learning architecture

**E Learning Trends:**

e-learning is attracting more and more users and can improve the knowledge of the learners anywhere anytime. Due to this, the organizations are paying a more attention toward this learning tool.

**YouTube/Video-Based Learning**: Nowadays, learning based on videos is one of the compelling and effective methods of learning. In YouTube or video-based learning, the academicians, researchers, and practitioners are sharing their recorded video lectures. The videos help students and employees to improve their understandings, training, and knowledge of specific course.

**Forum-Based Learning:** Another proficient technique is forum-based learning, in which the learners exchange their questions, ideas, and problems on a common platform. In response to questions, different answers or ideas are shared by an expert that helps to improve the skills of the learner and boost his confidence.

**Social E-Learning:** Social e-learning is a modern tool for effective e-learning and the latest trend of 2017–2018, for the learners. The learner is having a facility to participate and share his views and experiences, to discover and explore suitable learning. Learners find this tool quite rich and dependable for better understanding and to improve efficiency.

**Benefits of Cloud Computing for E-Learning Solutions**

As we know that many educational institutions do not have resources and infrastructure that are needed to run or install e-learning solution. Some of the versions of base applications of BlackBoard and MOODLE are cloud-oriented for e-Learning software .

Different educational organizations are widely using e-learning at several levels for continuous education, academic courses, company training, etc. Implementing e-learning systems with cloud computing has numerous advantages over other e-learning systems

• Low cost.

• Instant software updates.

• Cloud-based e-learning platforms are safe.

• Increases employee productivity and performance.

• Allows for seamless collaboration in the distributed workforce.

• Improved document format compatibility.

• Internal IT support is not required.

• Improves employee retention.

• Cloud-based e-learning platforms are dependable.

• Benefits for students’ point of view.

(a) Take online course.

(b) Send homework or assignments, projects,

(c) Send feedback, and

(d) Take exams.

• Benefits for teachers or trainers:

(a) Deal with content management,

(b) Communicate with students (forums),

(c) Prepare tests,

(d) Evaluate tests,

(e) Assess students homework or assignments, projects, and

(f) Send feedback

16. To create an Amazon Lex bot (console)

1. Sign in to the AWS Management Console and open the Amazon Lex console at <https://console.aws.amazon.com/lex/>.
2. If this is your first bot, choose **Get Started**; otherwise, on the **Bots** page, choose **Create**.
3. On the **Create your Lex bot** page, provide the following information, and then choose **Create**.
   * Choose the **OrderFlowers** blueprint.
   * Leave the default bot name (OrderFlowers).
   * For **COPPA**, choose **No**.
   * For **User utterance storage**, choose the appropriate response.
4. Choose **Create**. The console makes the necessary requests to Amazon Lex to save the configuration. The console then displays the bot editor window.
5. Wait for confirmation that your bot was built.
6. Test the bot.

