

Name : Varush H

Branch : Electrical and Electronics Engineering

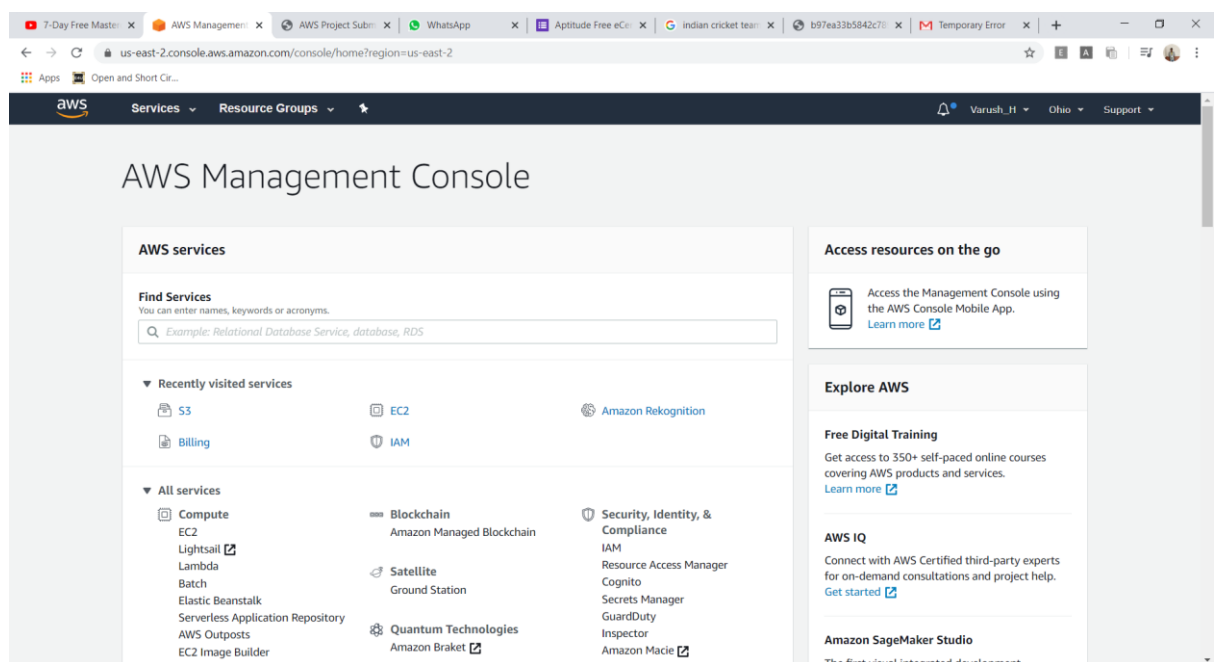
Reg-No : 17BEE0154

Contact : 8344955229

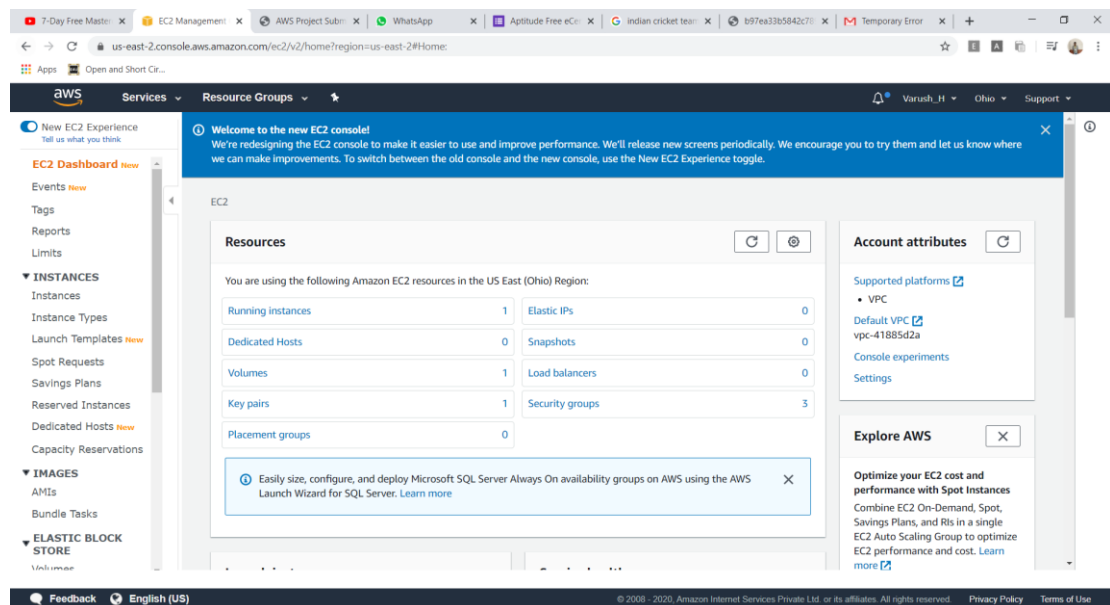
College : Vellore Institute Of Technology - Vellore

ScreenShots for Dashboards:

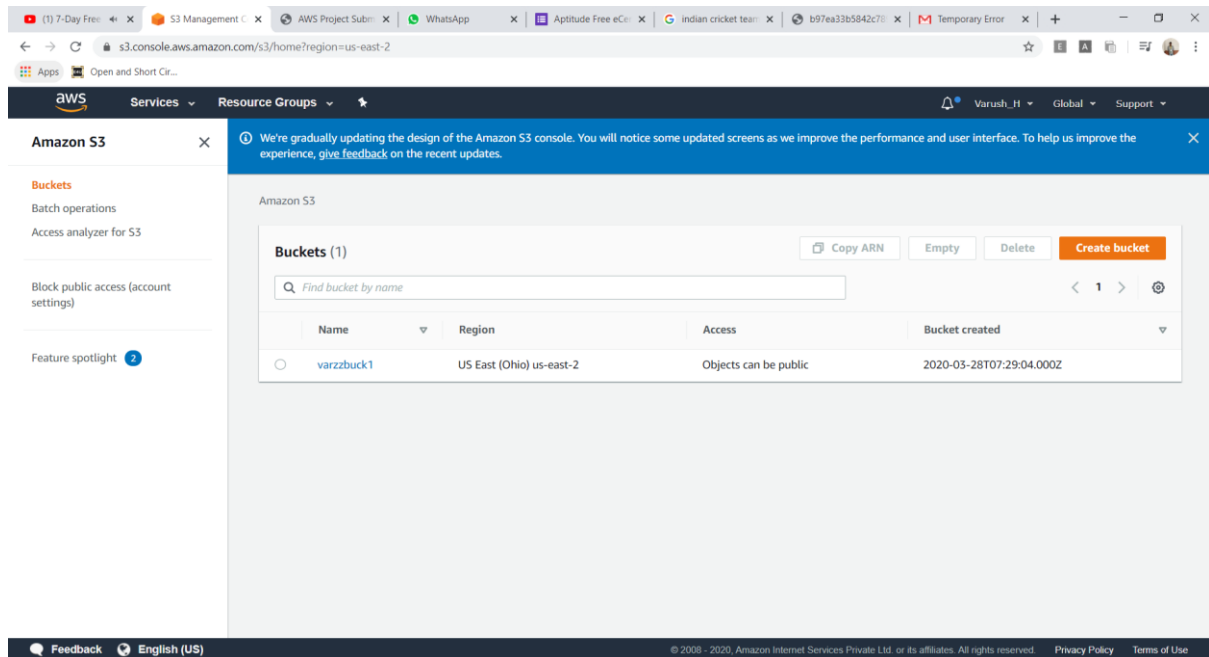
1. Login Screen with username :



2. EC2 Dashboard :



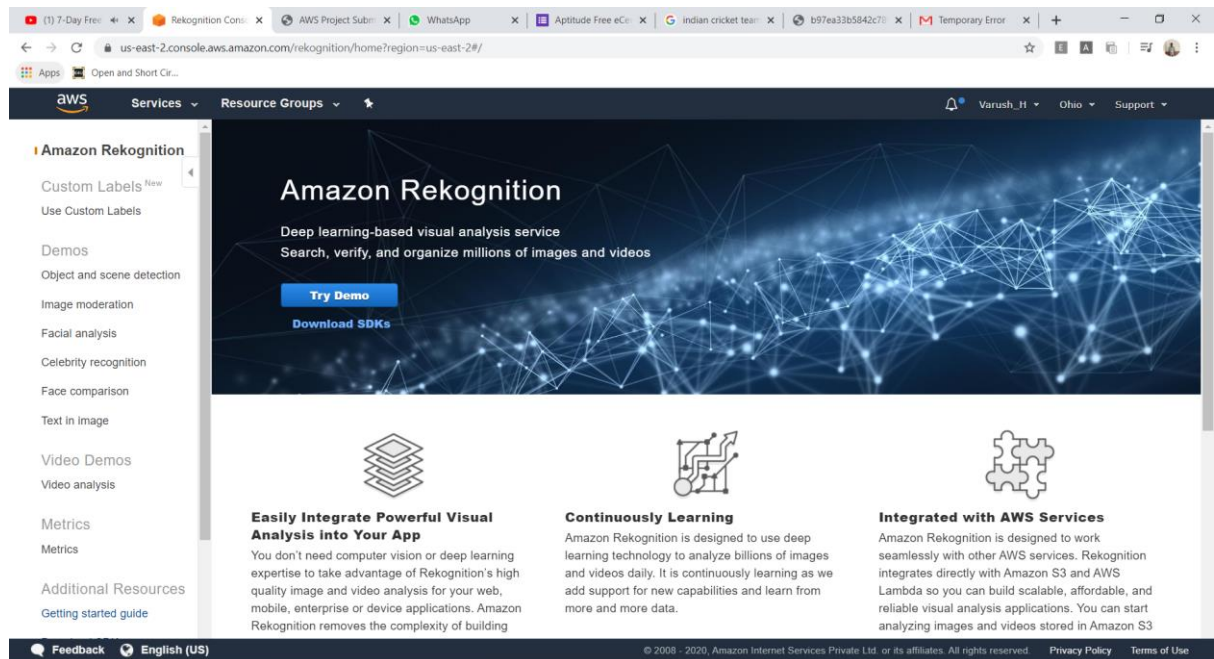
3. S3 Dashboard :



The screenshot shows the Amazon S3 console interface. The left sidebar contains navigation links: Buckets, Batch operations, Access analyzer for S3, Block public access (account settings), and Feature spotlight. The main content area displays a message about the console update and a 'Buckets (1)' section. This section includes a search bar, a table with one bucket, and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'.

Name	Region	Access	Bucket created
varzsbuck1	US East (Ohio) us-east-2	Objects can be public	2020-03-28T07:29:04.000Z

4. Rekognition Dashboard :



The screenshot shows the Amazon Rekognition dashboard. The left sidebar lists various features: Custom Labels, Demos, Object and scene detection, Image moderation, Facial analysis, Celebrity recognition, Face comparison, Text in image, Video Demos, Metrics, and Additional Resources. The main content area features a large header with the text 'Amazon Rekognition' and 'Deep learning-based visual analysis service'. Below this, there are three columns of content: 'Easily Integrate Powerful Visual Analysis into Your App', 'Continuously Learning', and 'Integrated with AWS Services'.

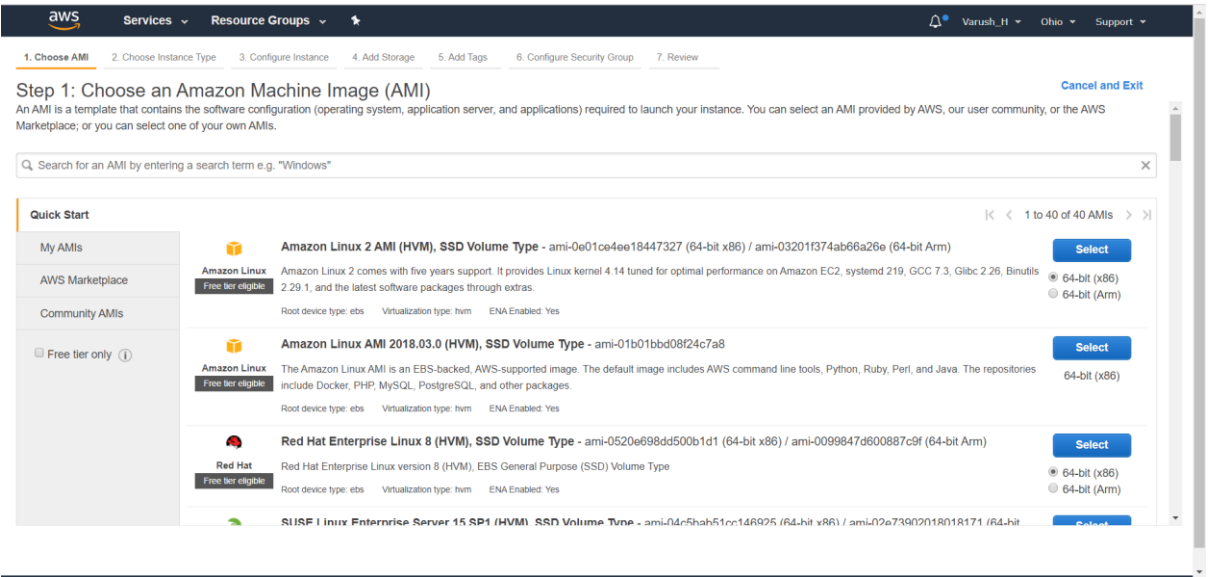
Easily Integrate Powerful Visual Analysis into Your App
You don't need computer vision or deep learning expertise to take advantage of Rekognition's high quality image and video analysis for your web, mobile, enterprise or device applications. Amazon Rekognition removes the complexity of building

Continuously Learning
Amazon Rekognition is designed to use deep learning technology to analyze billions of images and videos daily. It is continuously learning as we add support for new capabilities and learn from more and more data.

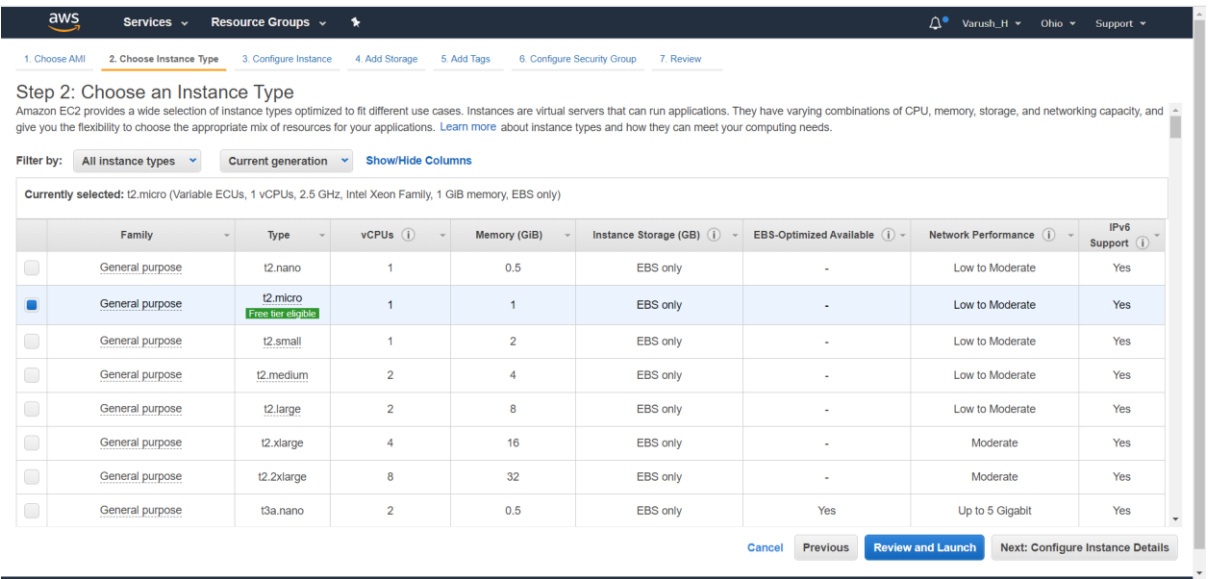
Integrated with AWS Services
Amazon Rekognition is designed to work seamlessly with other AWS services. Rekognition integrates directly with Amazon S3 and AWS Lambda so you can build scalable, affordable, and reliable visual analysis applications. You can start analyzing images and videos stored in Amazon S3

Screen Shots for EC2 :

1. Choosing an AMI :



2. Choosing an instance type :



Step 3: Configure Instance Details
Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option: ☐ Request Spot Instances

Network: vpc-41885d2a (default) [Create new VPC](#)

Subnet: No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP: Use subnet setting (Enable)

Placement group: ☐ Add instance to placement group

Capacity Reservation: Open [Create new Capacity Reservation](#)

IAM role: None [Create new IAM role](#)

Shutdown behavior: Stop

Stop - Hibernate behavior: ☐ Enable hibernation as an additional stop behavior

Enable termination protection: ☐ Protect against accidental termination

Monitoring: ☐ Enable CloudWatch detailed monitoring
Additional charges apply.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

3. Adding Storage :

Step 4: Add Storage
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-0f54692056aaa4c20	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

4. Configuring security group :

Step 6: Configure Security Group
A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group

☐ Select an existing security group

Security group name: launch-wizard-3

Description: launch-wizard-3 created 2020-04-05T11:41:46.056+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[Add Rule](#)

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

5. Key Pair Download :

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair

Select a key pair

aws-webinar-key

☒ I acknowledge that I have access to the selected private key file (aws-webinar-key.pem), and that without this file, I won't be able to log into my instance.

Cancel

Launch Instances

aws

Services

Resource Groups

Varush_HOhioSupport

Launch Status

✓

Your instances are now launching

The following instance launches have been initiated: I-0b92774169777b3f6 [View launch log](#)

ℹ

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click **View instances** to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

▼ Here are some helpful resources to get you started

• [How to connect to your Linux instance](#)

• [Learn about AWS Free Usage Tier](#)

• [Amazon EC2: User Guide](#)

• [Amazon EC2: Discussion Forum](#)

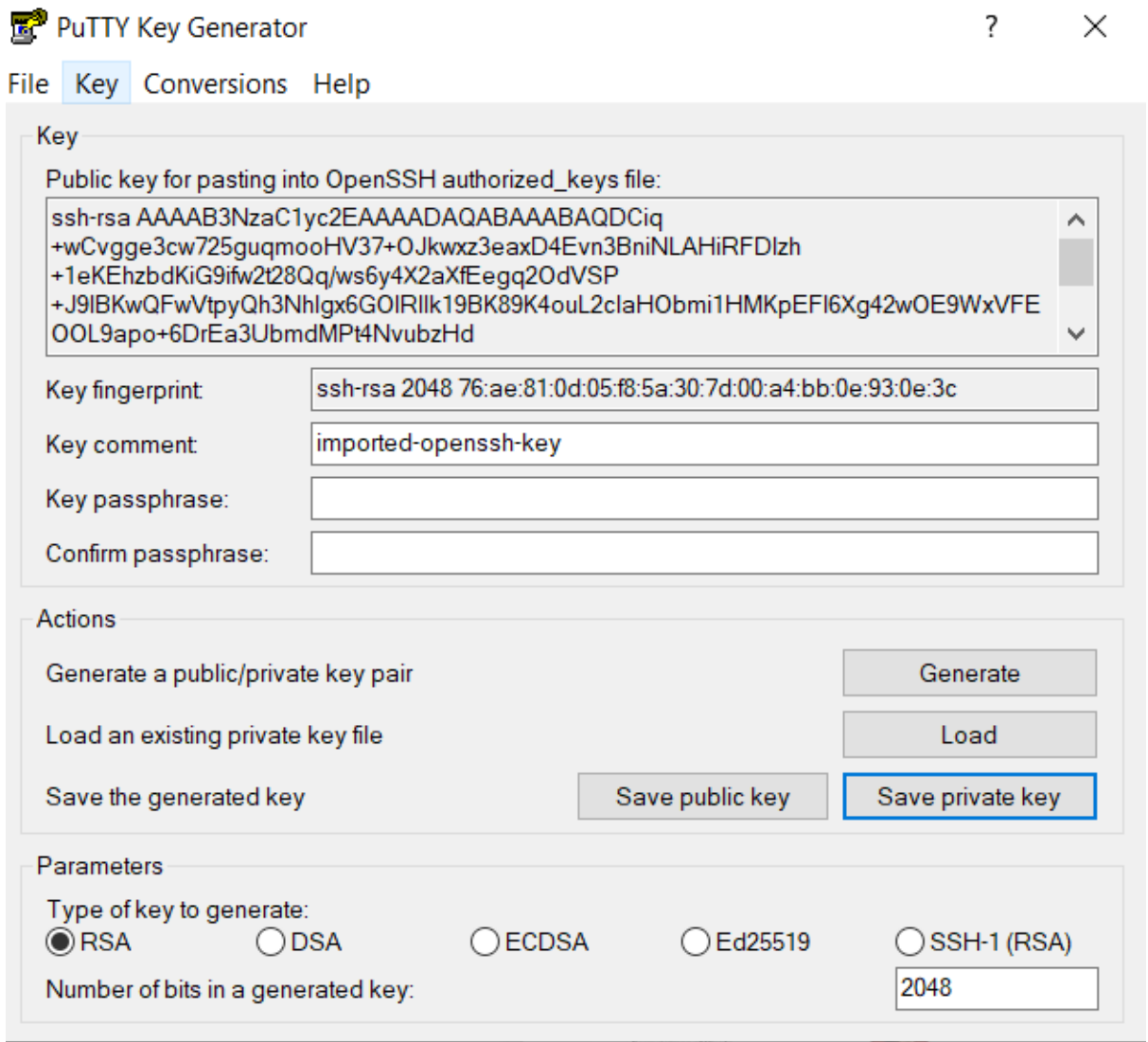
While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

[Create and attach additional EBS volumes](#) (Additional charges may apply)

[Manage security groups](#)

6. PuTTYgen conversion from pem to ppk:

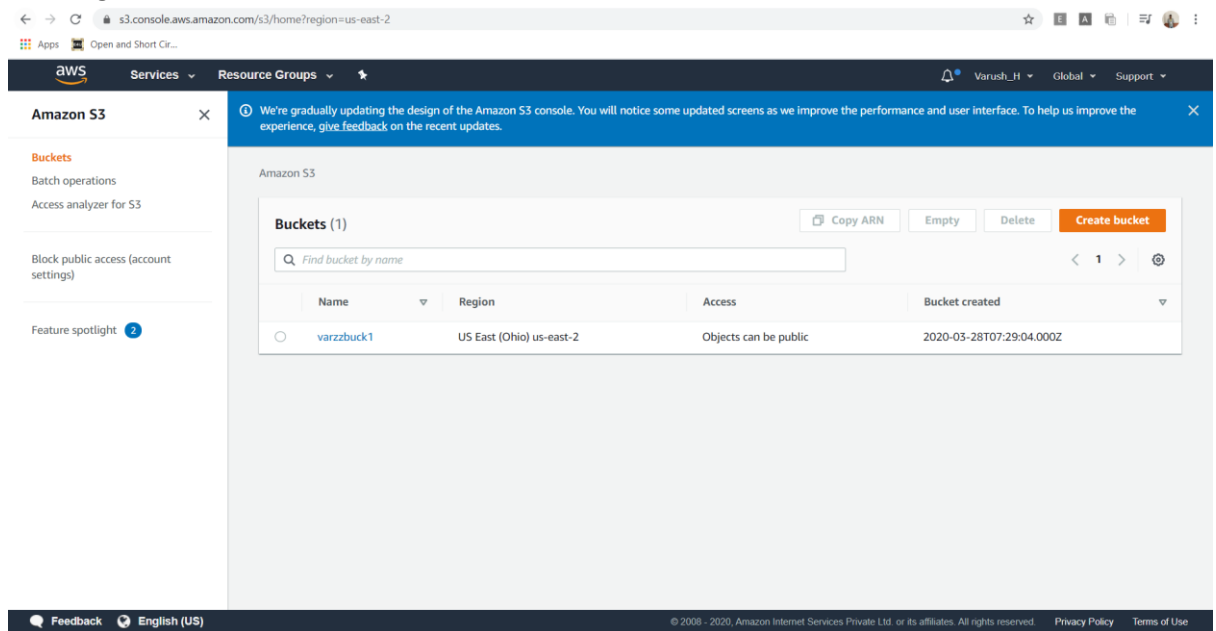


7. Logged in EC2 black screen

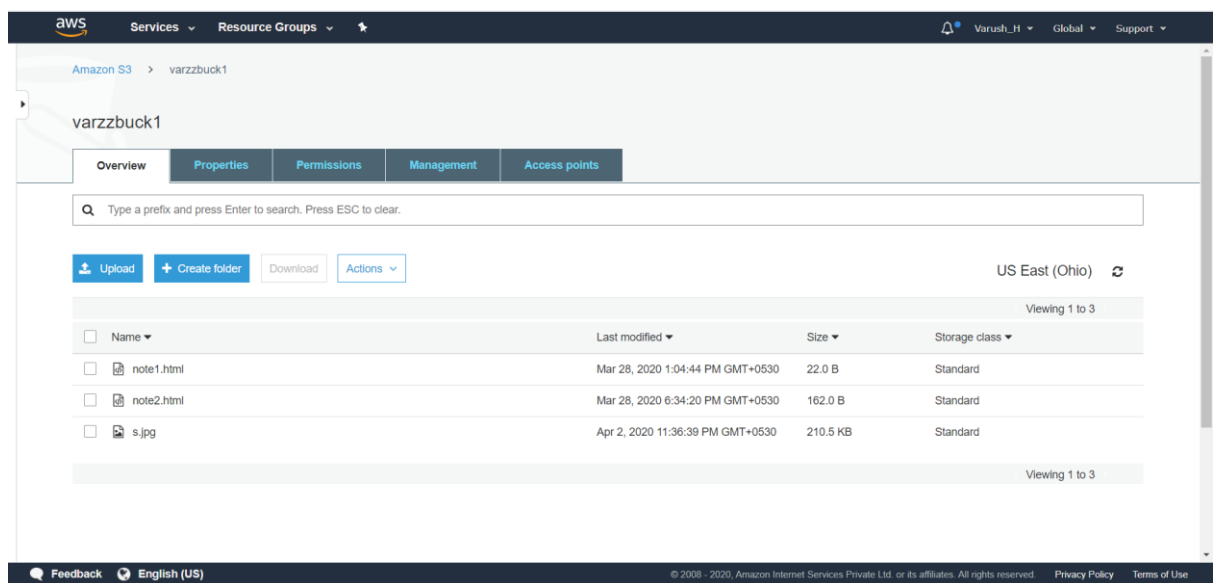
```
ec2-user@ip-172-31-42-137:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
 _ | _ | )  
 _ | ( _ /  Amazon Linux 2 AMI  
 _ | \ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
1 package(s) needed for security, out of 7 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-42-137 ~]$ sudo yum install httpd  
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd  
amzn2-core | 2.4 kB 00:00  
Resolving Dependencies  
--> Running transaction check  
--> Package httpd.x86_64 0:2.4.41-1.amzn2.0.1 will be installed  
--> Processing Dependency: httpd-tools = 2.4.41-1.amzn2.0.1 for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: httpd-filesystem = 2.4.41-1.amzn2.0.1 for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: mod_http2 for package: httpd-2.4.41-1.amzn2.0.1.x86_64
```

Screen Shots for S3:

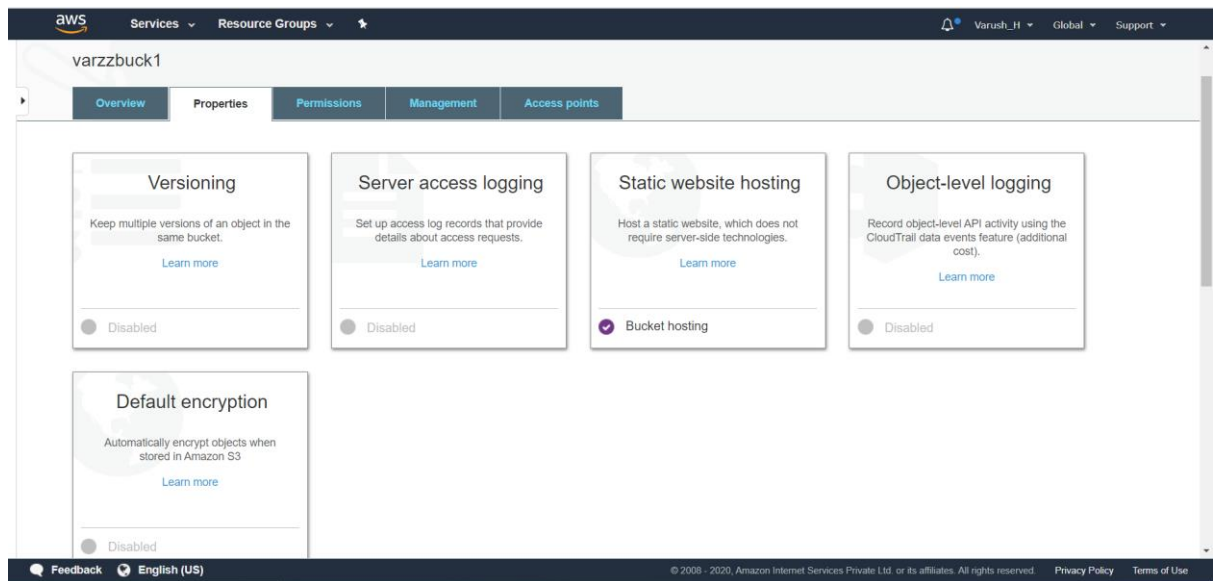
1. Creating a bucket :



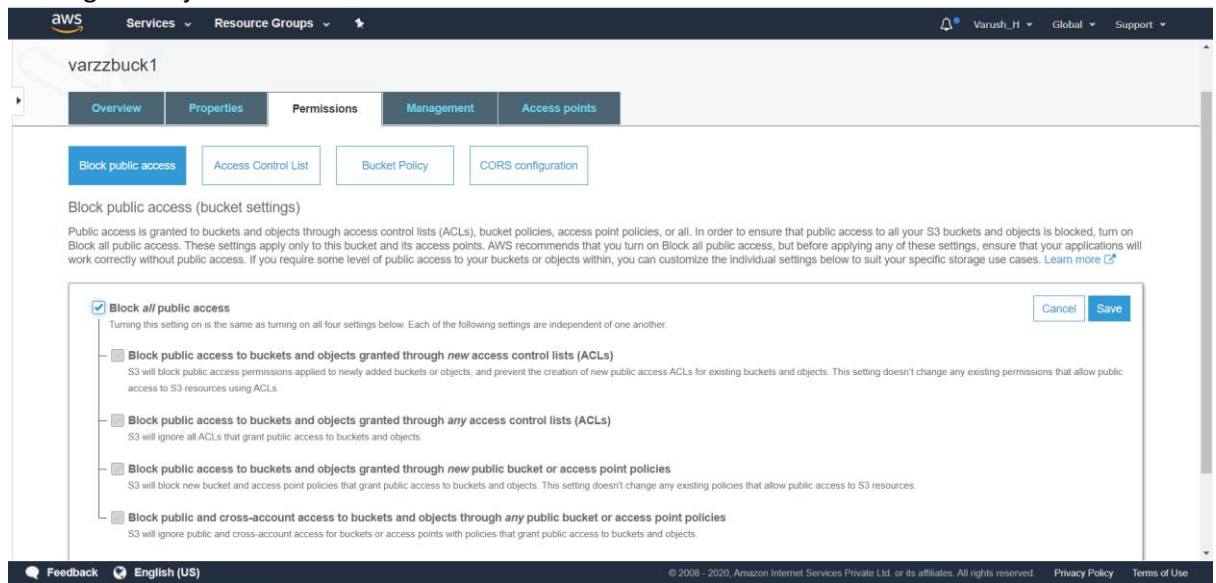
2. Uploading an object :



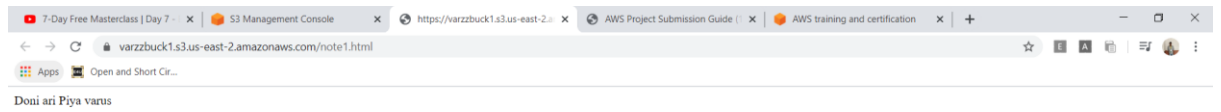
3. Enabling Static Website :



4. Making the Object Public:

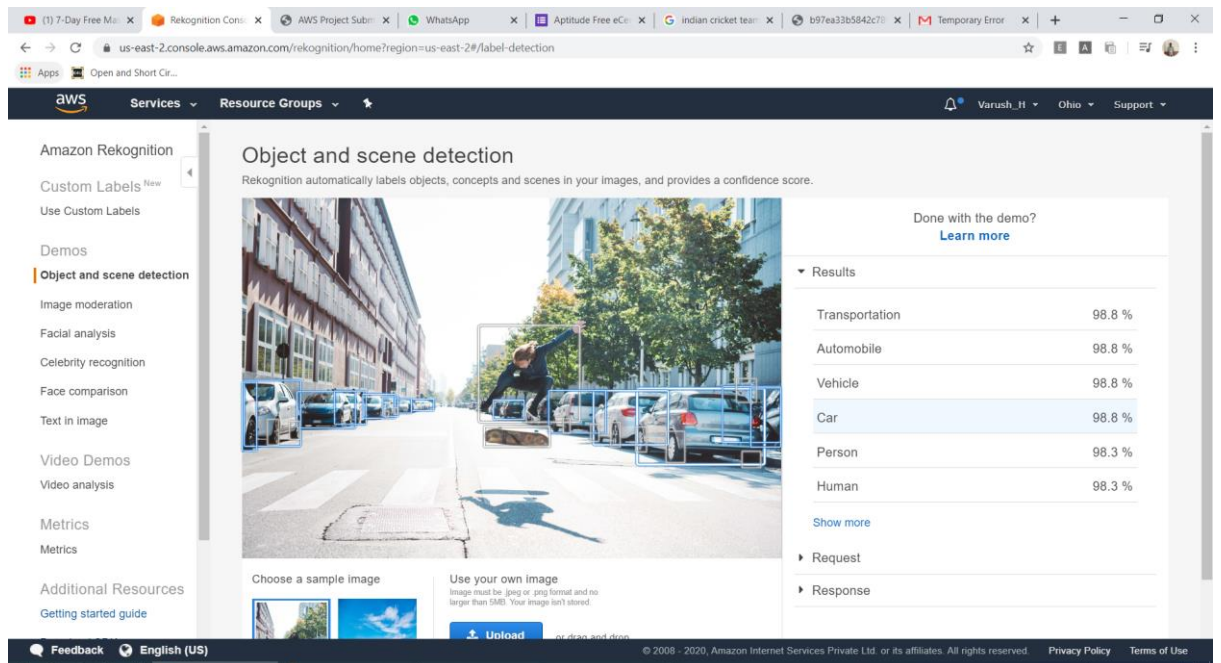


5. Checking the S3 link on the browser :

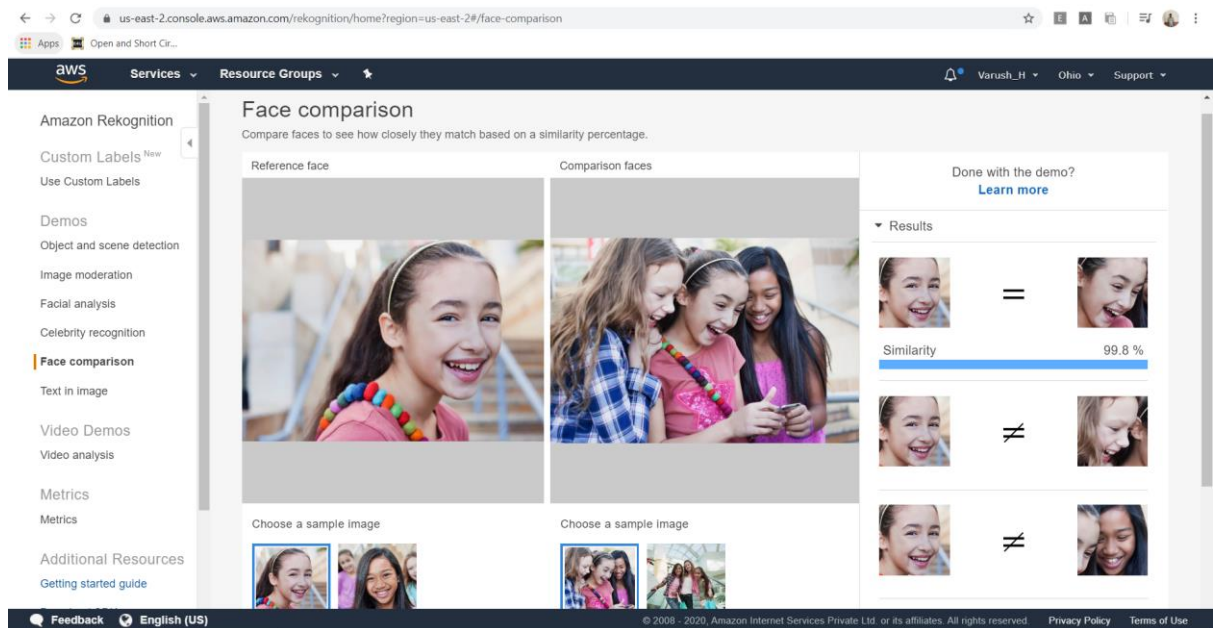


ScreenShots for Rekognition :

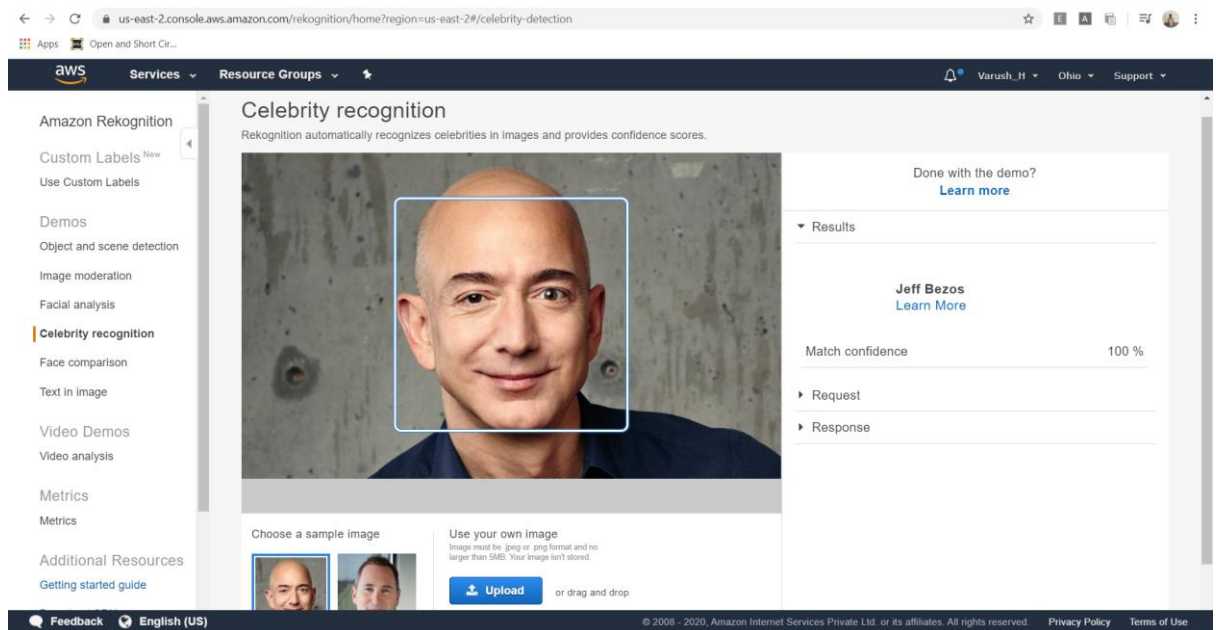
1. Face Detect :



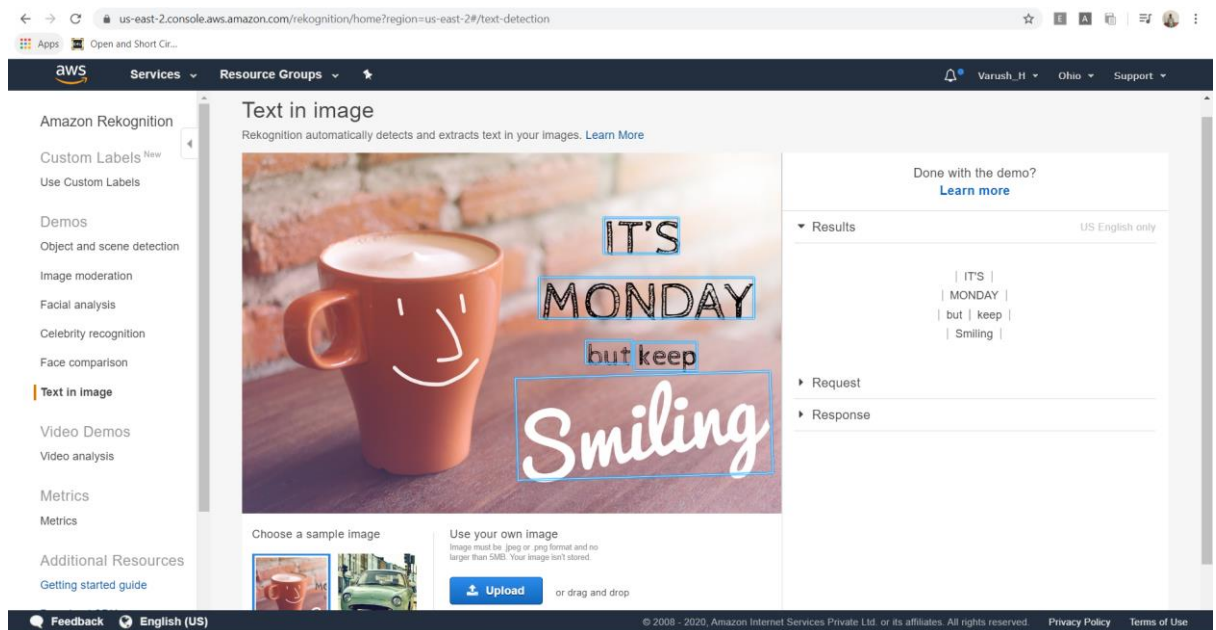
2. Face Compare:



3. Celebrity Recognition



4. Text In Image:

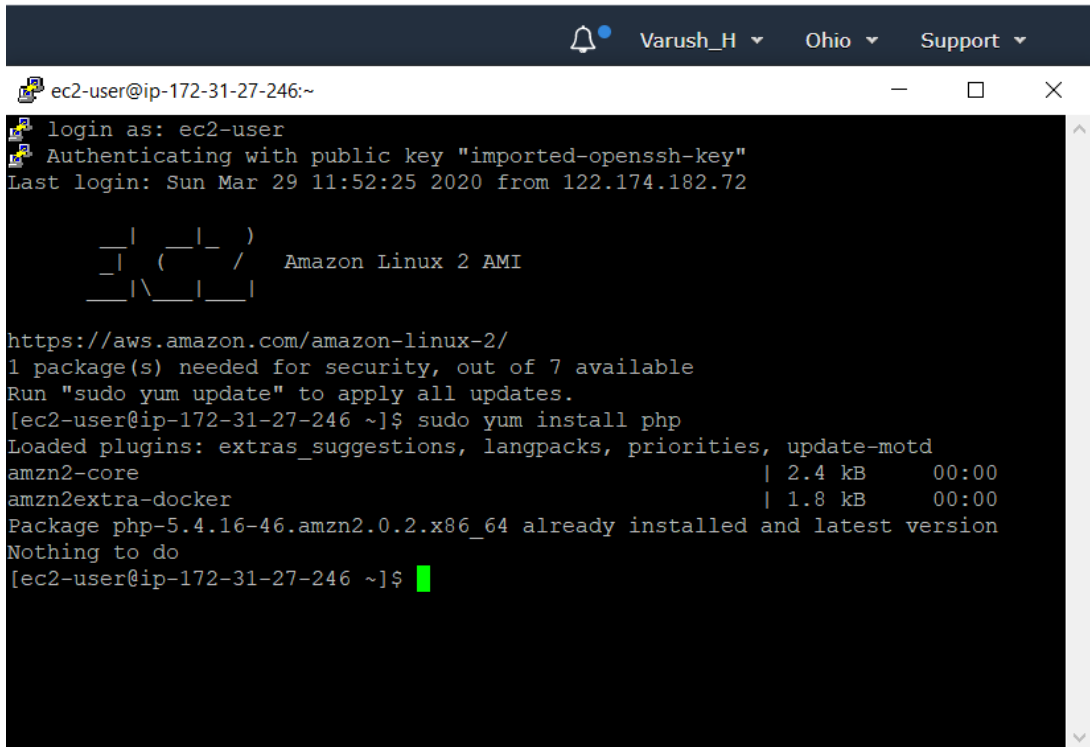


ScreenShots for EC2 and S3 :

1. Installing aws-sdk

```
ec2-user@ip-172-31-27-246:~/var/www/html/face
1073741824 bytes (1.1 GB) copied, 13.411 s, 80.1 MB/s
[ec2-user@ip-172-31-27-246 face]$ sudo /sbin/mkswap /var/swap.1
mkswap: /var/swap.1: insecure permissions 0644, 0600 suggested.
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=77514935-1134-4e03-bela-c529ce35be5c
[ec2-user@ip-172-31-27-246 face]$ sudo /sbin/swapon /var/swap.1
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-27-246 face]$ sudo php -d memory_limit=-1 ~/composer.phar require aw
s/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 3 installs, 0 updates, 0 removals
  - Installing symfony/event-dispatcher (v2.8.52): Loading from cache
  - Installing guzzle/guzzle (v3.9.3): Downloading (100%)
  - Installing aws/aws-sdk-php (2.8.31): Downloading (100%)
symfony/event-dispatcher suggests installing symfony/dependency-injection
symfony/event-dispatcher suggests installing symfony/http-kernel
guzzle/guzzle suggests installing guzzlehttp/guzzle (Guzzle 5 has moved to a new package
name. The package you have installed, Guzzle 3, is deprecated.)
aws/aws-sdk-php suggests installing doctrine/cache (Adds support for caching of credenti
als and responses)
aws/aws-sdk-php suggests installing ext-apc (Allows service description opcode caching,
request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HTTP requ
ests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write manifests f
or creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle ins
tead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-27-246 face]$
```

2. Installing PHP :



```
ec2-user@ip-172-31-27-246:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
Last login: Sun Mar 29 11:52:25 2020 from 122.174.182.72  
  
  _ | _ | _ )  
  _ | ( _ | /  Amazon Linux 2 AMI  
  _ | \ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
1 package(s) needed for security, out of 7 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-27-246 ~]$ sudo yum install php  
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd  
amzn2-core | 2.4 kB 00:00  
amzn2extra-docker | 1.8 kB 00:00  
Package php-5.4.16-46.amzn2.0.2.x86_64 already installed and latest version  
Nothing to do  
[ec2-user@ip-172-31-27-246 ~]$
```

3 Index.php Code:

```
*index.php - Notepad  
File Edit Format View Help  
<?php  
  
/*  
Install php - sudo yum install php  
curl -sS https://getcomposer.org/installer | php  
cd /var/www/html  
sudo mkdir face  
cd face  
sudo php -d memory_limit=-1 ~/composer.phar require aws/aws-sdk-php  
  
In case if you get memory error -  
sudo /bin/dd if=/dev/zero of=/var/swap.1 bs=1M count=1024  
sudo /sbin/mkswap /var/swap.1  
sudo /sbin/swapon /var/swap.1  
  
sudo wget https://i.pinimg.com/originals/b9/7e/a3/b97ea33b5842c7894b804923c6c05580.jpg  
sudo mv b97ea33b5842c7894b804923c6c05580.jpg sample.jpg  
  
*/  
error_reporting(0);  
  
require_once(__DIR__ . '/vendor/autoload.php');  
  
use Aws\S3\S3Client;  
use Aws\Rekognition\RekognitionClient;  
  
$bucket = 'varzzbuck1';  
$keyname = 's.jpg';  
  
$s3 = S3Client::factory([  
    'profile' => 'default',  
    'region' => 'us-east-2',  
    'version' => '2006-03-01',  
    'signature' => 'v4'  
]);  
  
try {  
    // Upload data.  
    $result = $s3->putObject([  
        'Bucket' => $bucket,  
        'Key' => $keyname,  
        'SourceFile' => __DIR__ . "/" . $keyname,  
        'ACL' => 'public-read'  
    ]);  
  
    // Print the URL to the object.  
    $imageUrl = $result['ObjectURL'];  
    if($imageUrl) {  
        echo "Image upload done... Here is the URL: " . $imageUrl;  
    }  
} catch (Exception $e) {  
    echo $e->getMessage() . PHP_EOL;  
}  
  
<
```

4. Upload Success Screenshot of image :

```
ec2-user@ip-172-31-27-246:/var/www/html/face
swapon: /var/swap.1: insecure permissions 0644, 0600 suggested.
[ec2-user@ip-172-31-27-246 face]$ sudo php -d memory_limit=-1 ~/composer.phar require aw
s/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 3 installs, 0 updates, 0 removals
  - Installing symfony/event-dispatcher (v2.8.52): Loading from cache
  - Installing guzzle/guzzle (v3.9.3): Downloading (100%)
  - Installing aws/aws-sdk-php (2.8.31): Downloading (100%)
symfony/event-dispatcher suggests installing symfony/dependency-injection
symfony/event-dispatcher suggests installing symfony/http-kernel
guzzle/guzzle suggests installing guzzlehttp/guzzle (Guzzle 5 has moved to a new package
name. The package you have installed, Guzzle 3, is deprecated.)
aws/aws-sdk-php suggests installing doctrine/cache (Adds support for caching of credenti
als and responses)
aws/aws-sdk-php suggests installing ext-apc (Allows service description opcode caching,
request and response caching, and credentials caching)
aws/aws-sdk-php suggests installing monolog/monolog (Adds support for logging HTTP reque
sts and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write manifests f
or creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle ins
tead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-27-246 face]$ ls
composer.json composer.lock index.php sample.jpg s.jpg vendor
[ec2-user@ip-172-31-27-246 face]$ sudo rm index.php
[ec2-user@ip-172-31-27-246 face]$ sudo vim index.php
[ec2-user@ip-172-31-27-246 face]$ sudo php index.php
Image upload done... Here is the URL: https://varzzbuck1.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip
```

