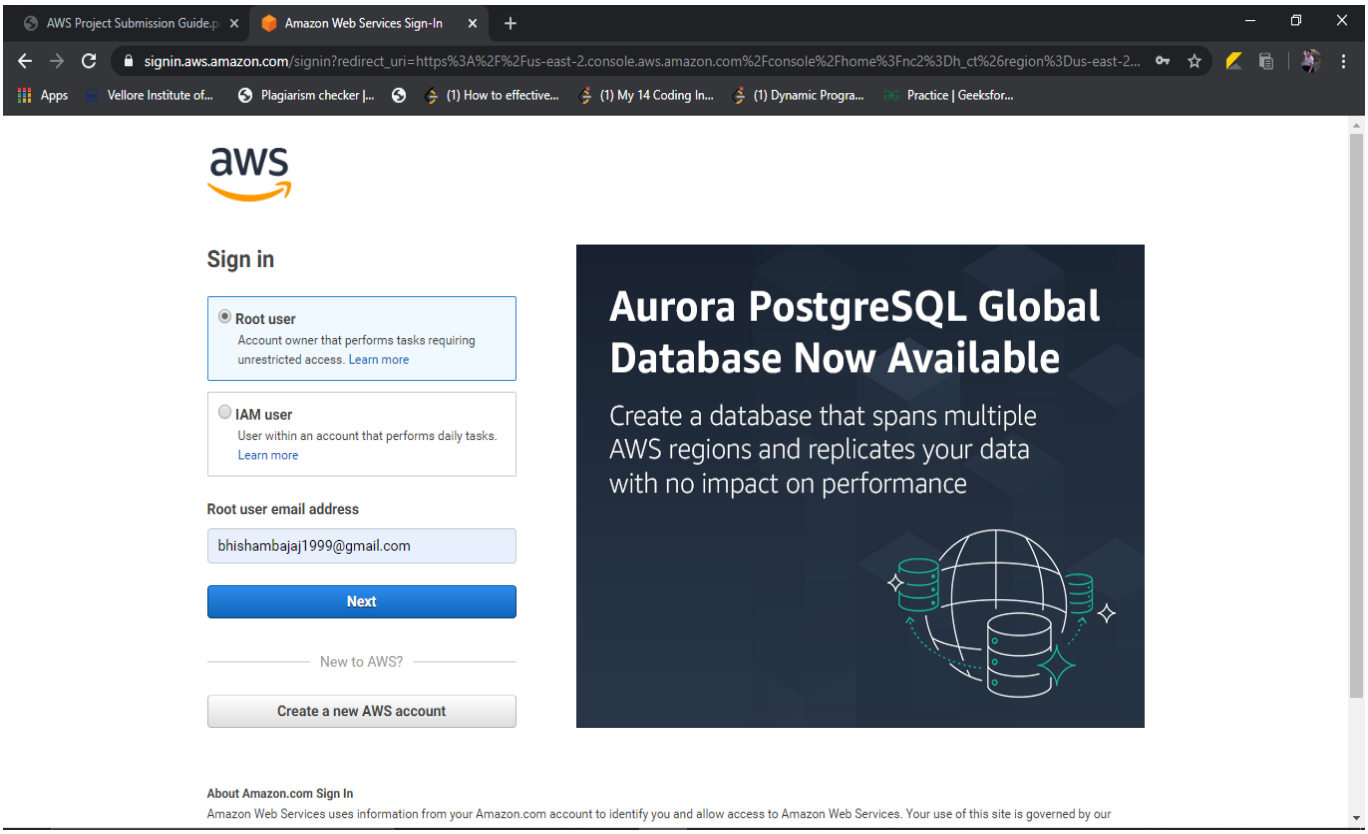
Masterclass on AWS Cloud-Building a Face Detection App

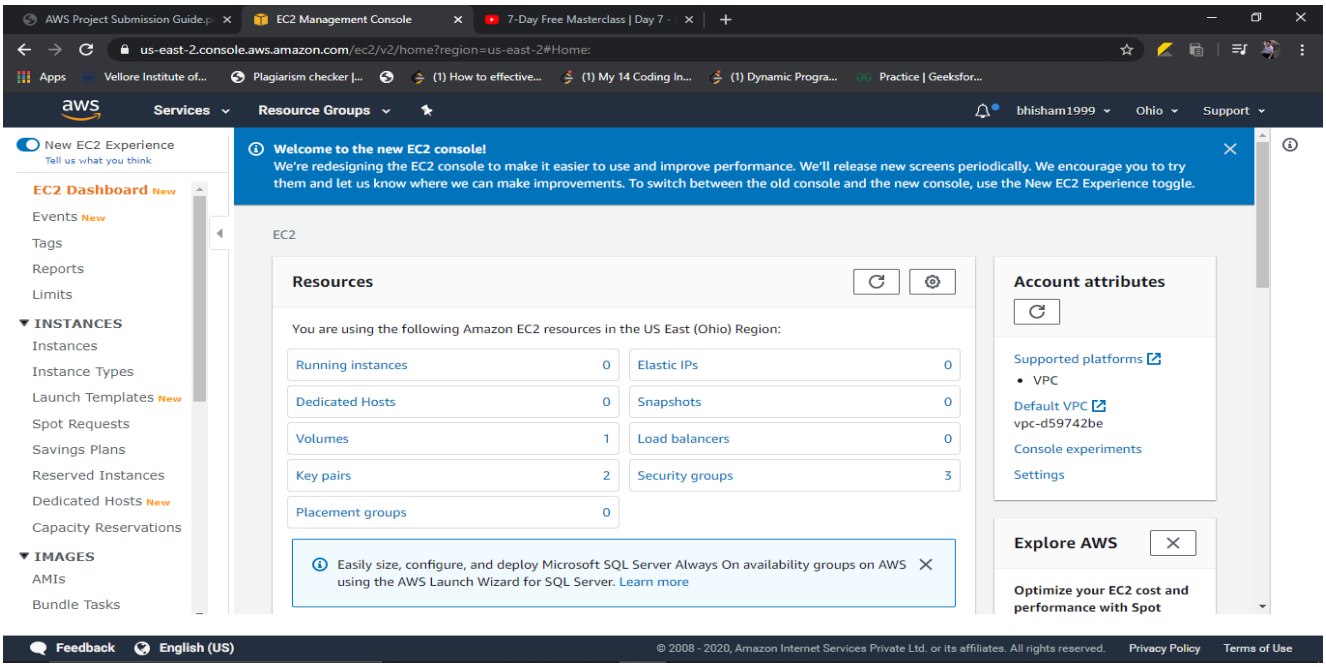
Proof of Working (Screenshots):

Screenshots needed for Dashboards:

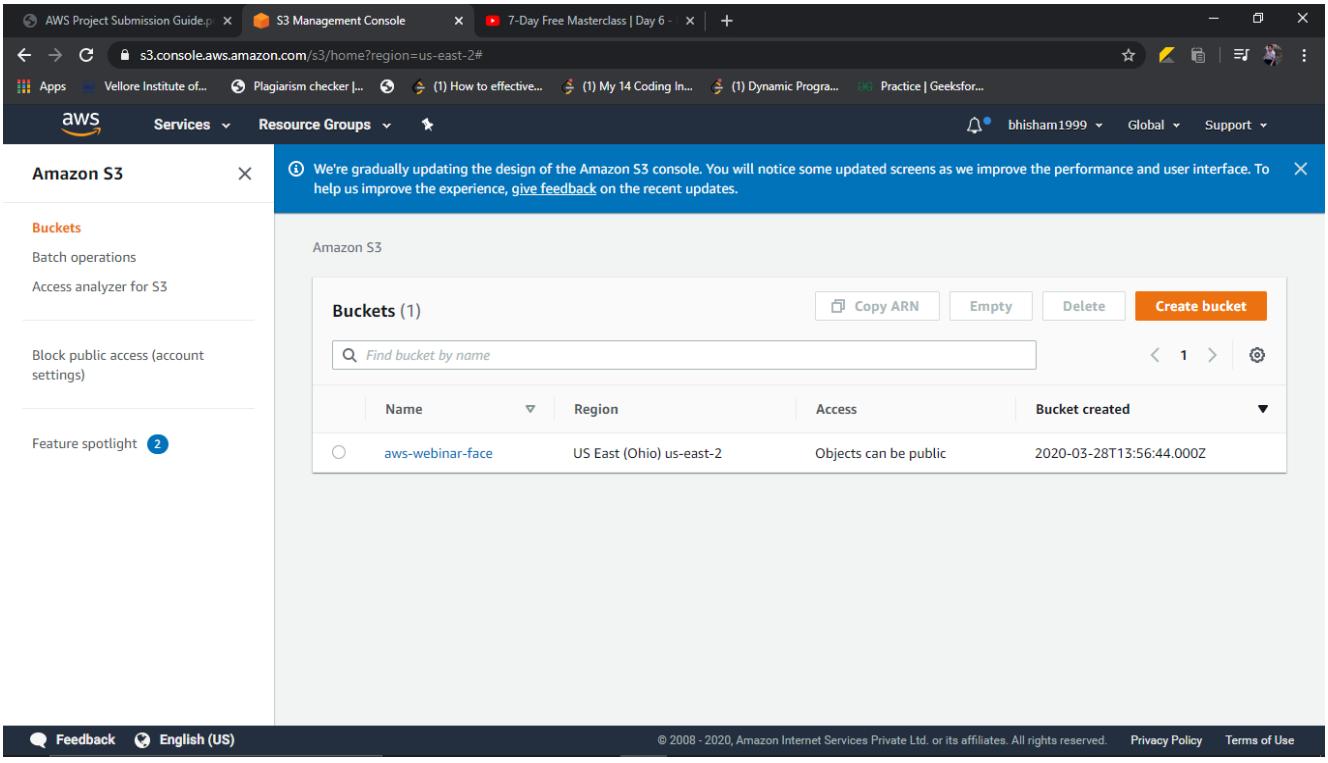
1.) AWS Login screen with username



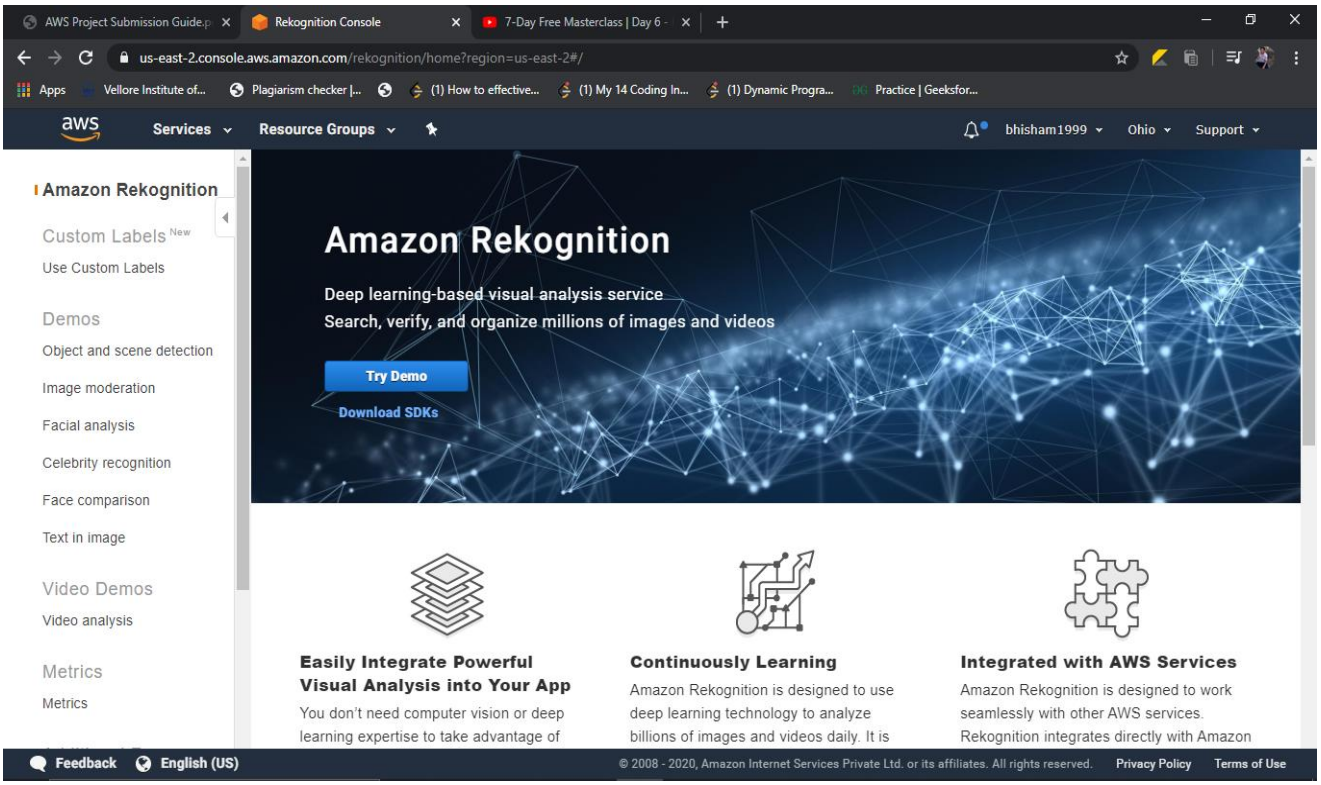
2.) EC2 Dashboard



3.) S3 dashboard

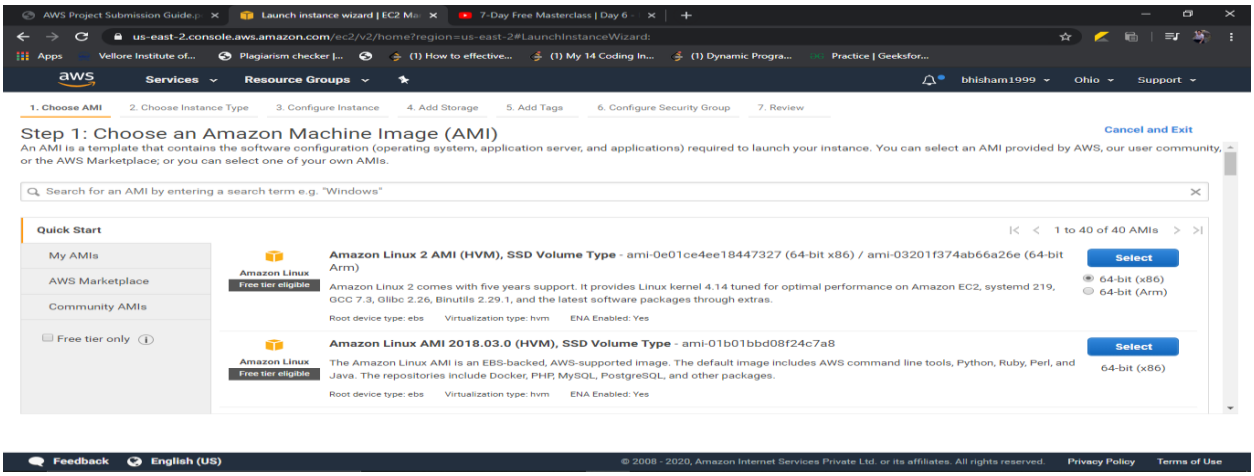


4.) Recognition Dashboard

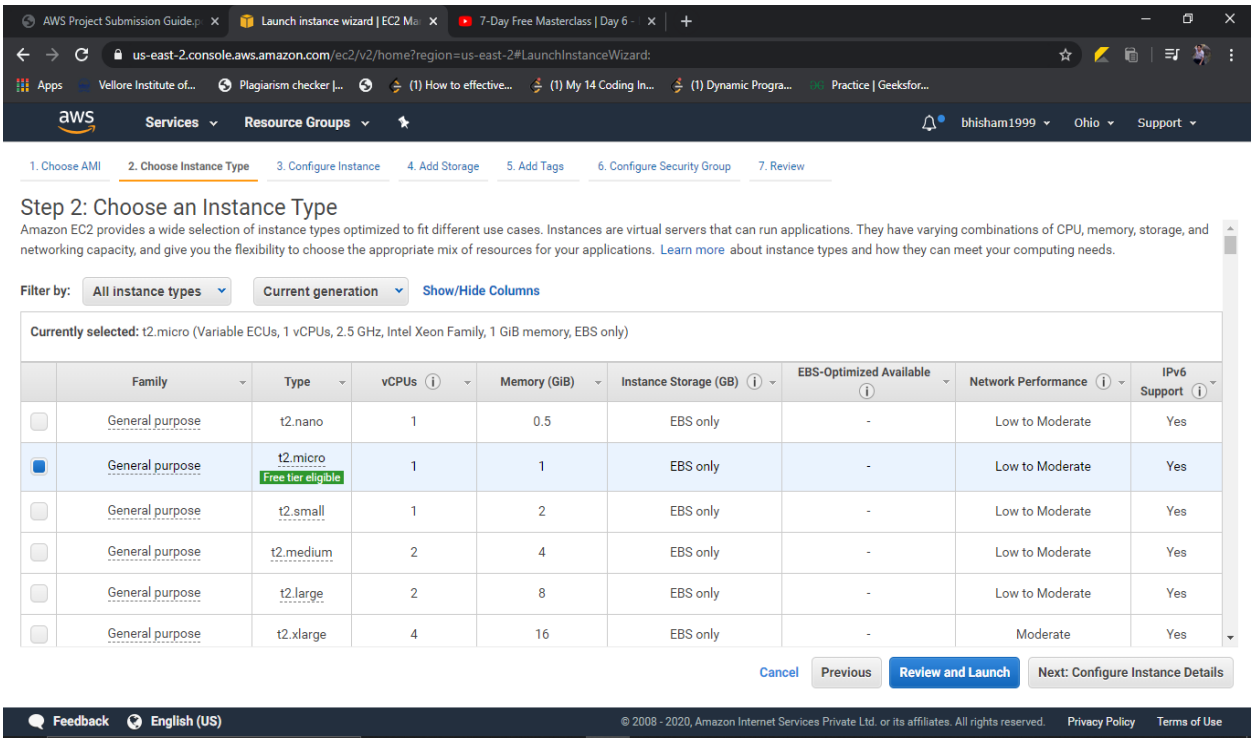


Screenshots needed for EC2:

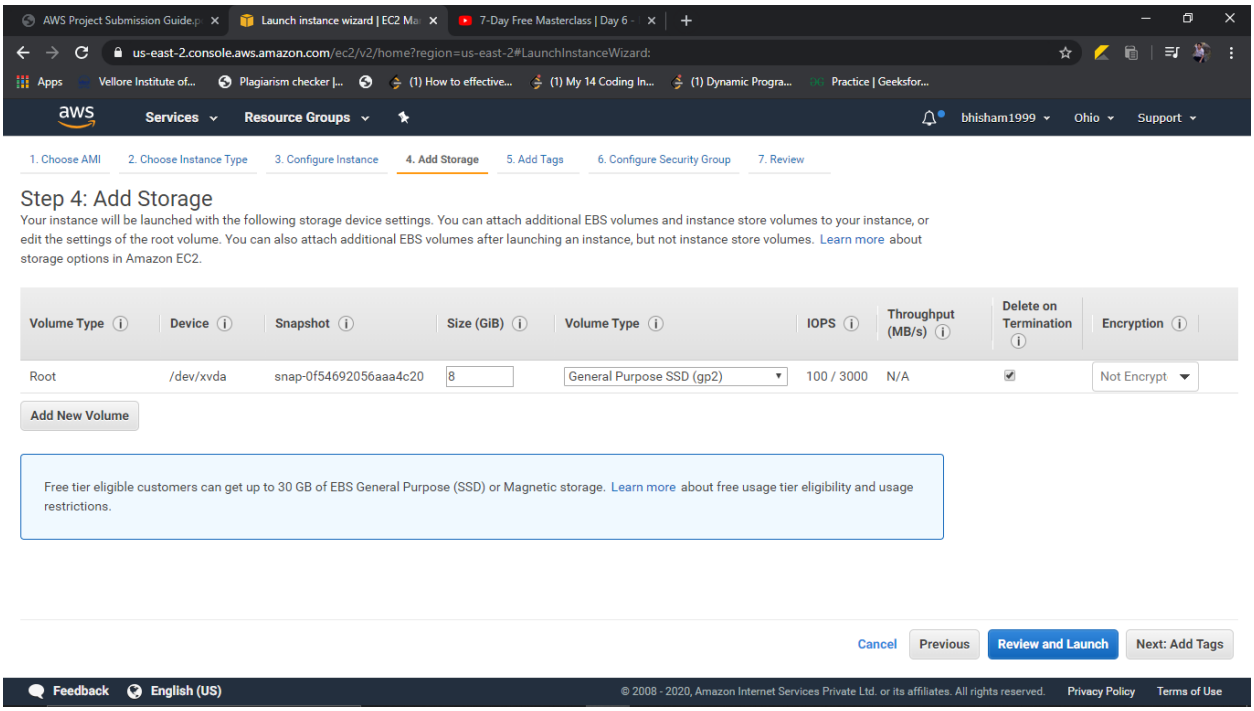
5: Choosing an AMI



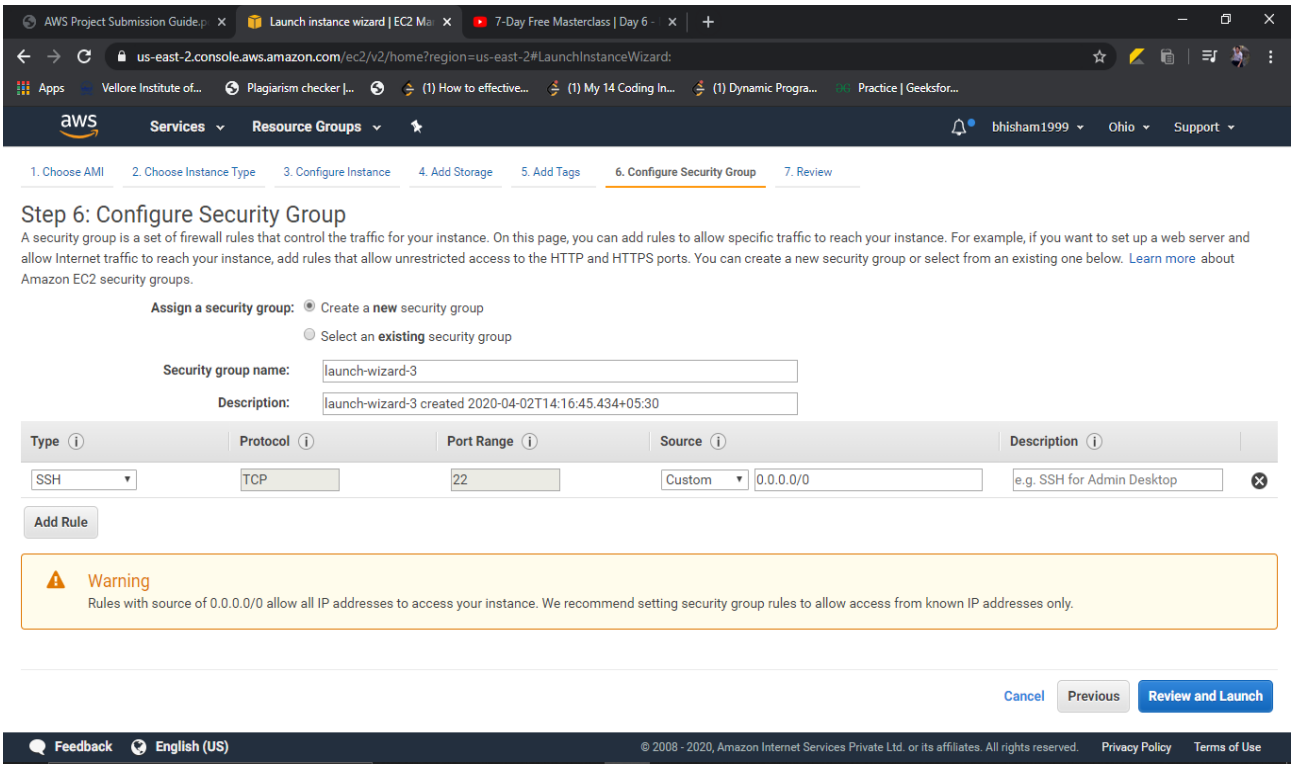
6.)


AWS Project Submission Guide.p
Launch instance wizard | EC2 Ma
7-Day Free Masterclass | Day 6 -
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
Apps
Vellore Institute of...
Plagiarism checker |...
(1) How to effective...
(1) My 14 Coding In...
(1) Dynamic Progra...
Practice | Geeksfor...
aws
Services
Resource Groups
bhisham1999
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.
Filter by:
All instance types
Current generation
Show/Hide Columns
Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)
Family
Type
vCPUs
Memory (GiB)
Instance Storage (GB)
EBS-Optimized Available
Network Performance
IPv6 Support
General purpose
t2.nano
1
0.5
EBS only
-
Low to Moderate
Yes
General purpose
t2.micro
Free tier eligible
1
1
EBS only
-
Low to Moderate
Yes
General purpose
t2.small
1
2
EBS only
-
Low to Moderate
Yes
General purpose
t2.medium
2
4
EBS only
-
Low to Moderate
Yes
General purpose
t2.large
2
8
EBS only
-
Low to Moderate
Yes
General purpose
t2.xlarge
4
16
EBS only
-
Moderate
Yes
Cancel
Previous
Review and Launch
Next: Configure Instance Details
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

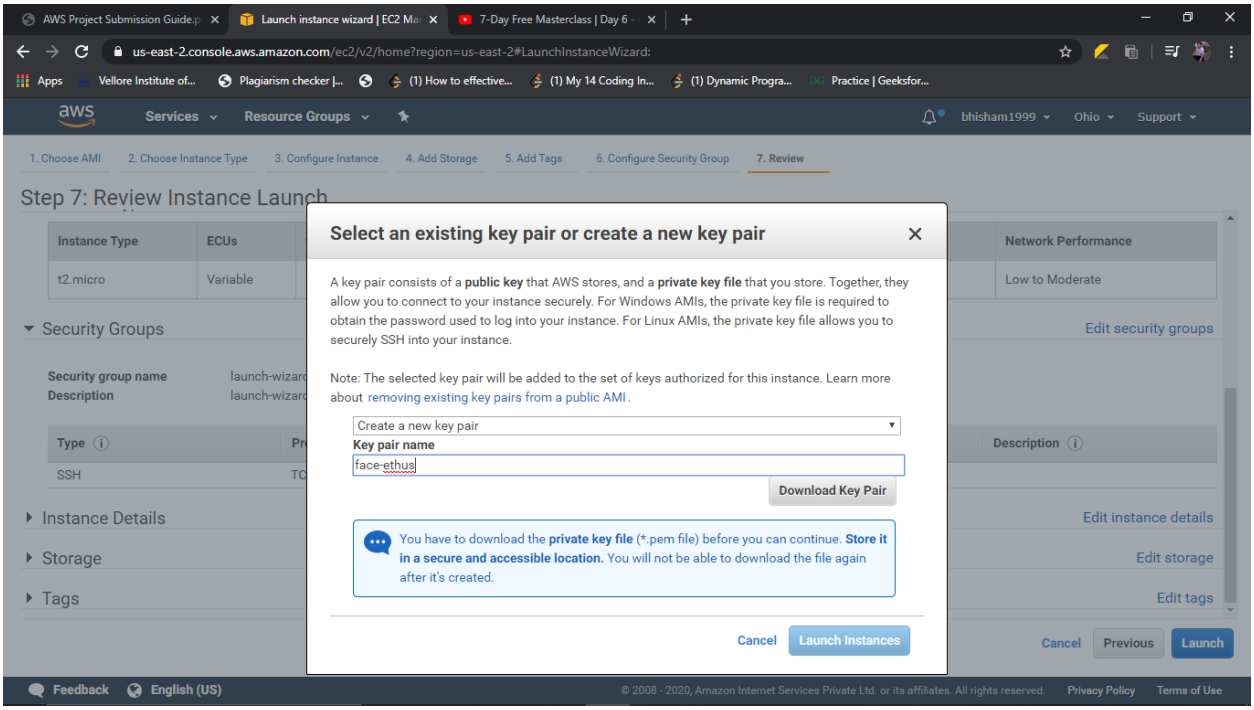
7.)


AWS Project Submission Guide.p
Launch instance wizard | EC2 Ma
7-Day Free Masterclass | Day 6 -
us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstanceWizard:
Apps
Vellore Institute of...
Plagiarism checker |...
(1) How to effective...
(1) My 14 Coding In...
(1) Dynamic Progra...
Practice | Geeksfor...
aws
Services
Resource Groups
bhisham1999
Ohio
Support
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
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
Not Encrypt
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
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

8.)

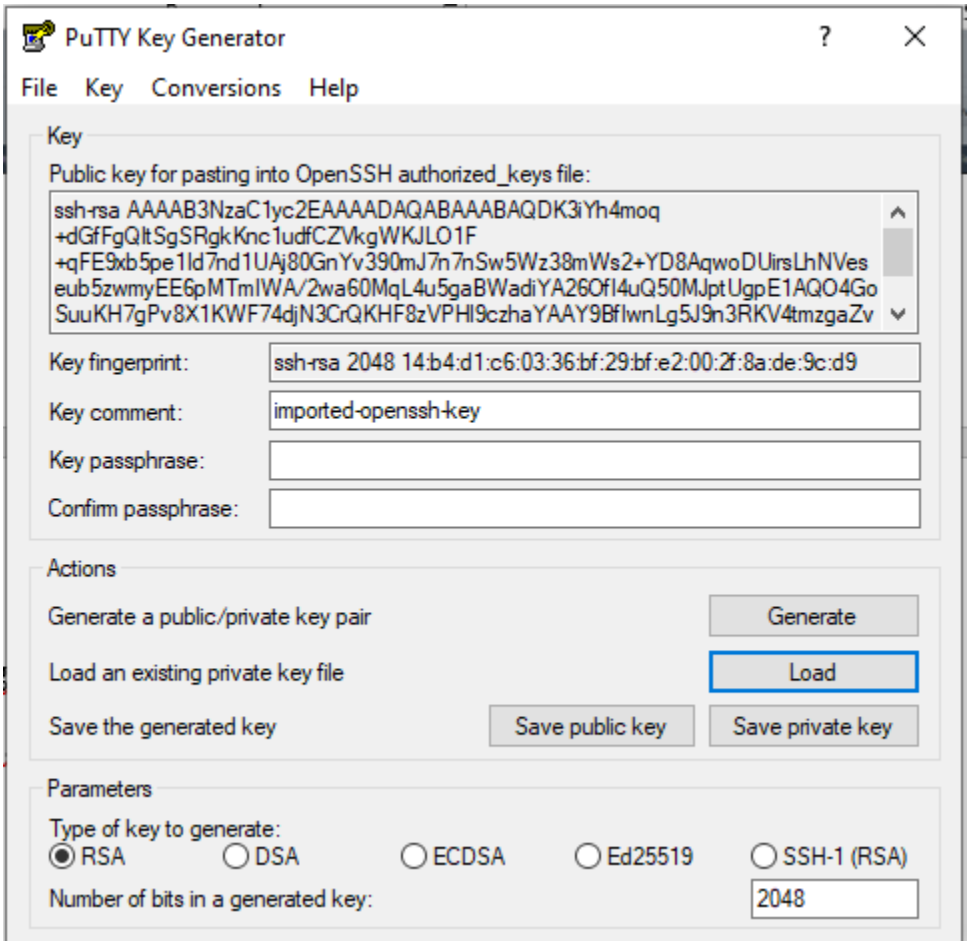


9.) face-ethus key pair



10.) puttygen conversion

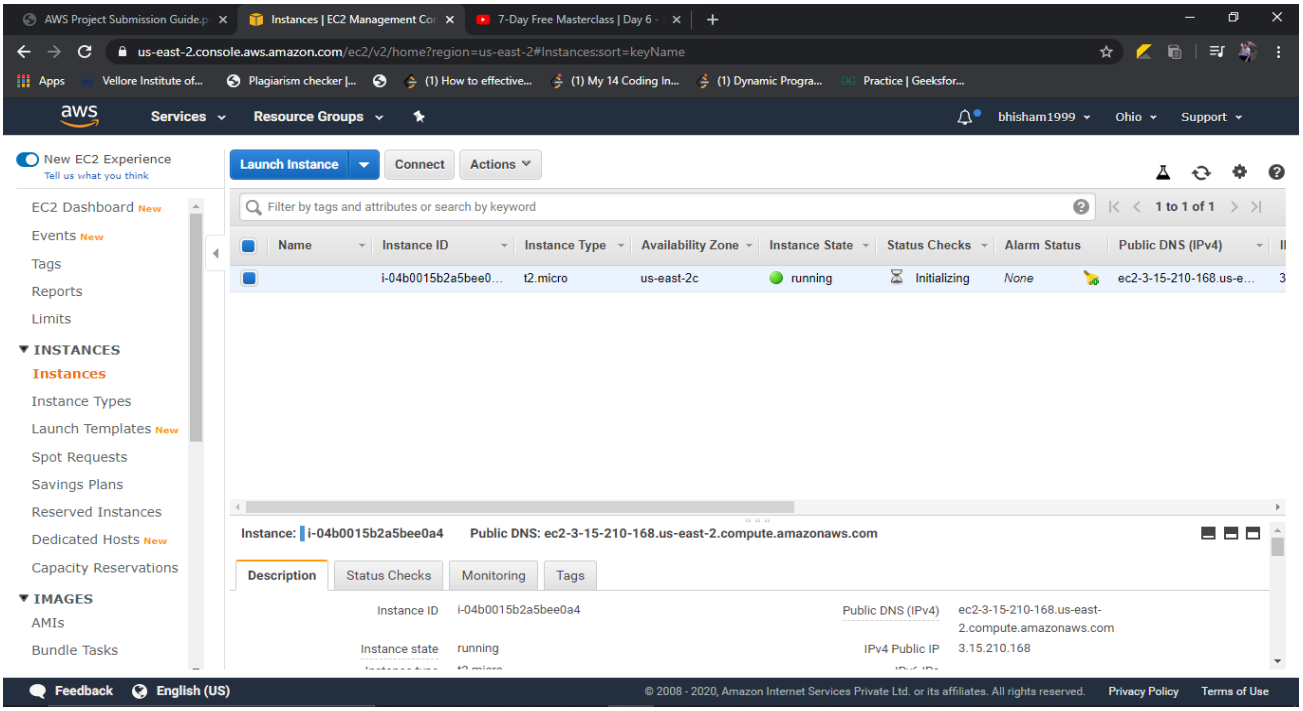
face-ethus.pem to face-ethus.ppk



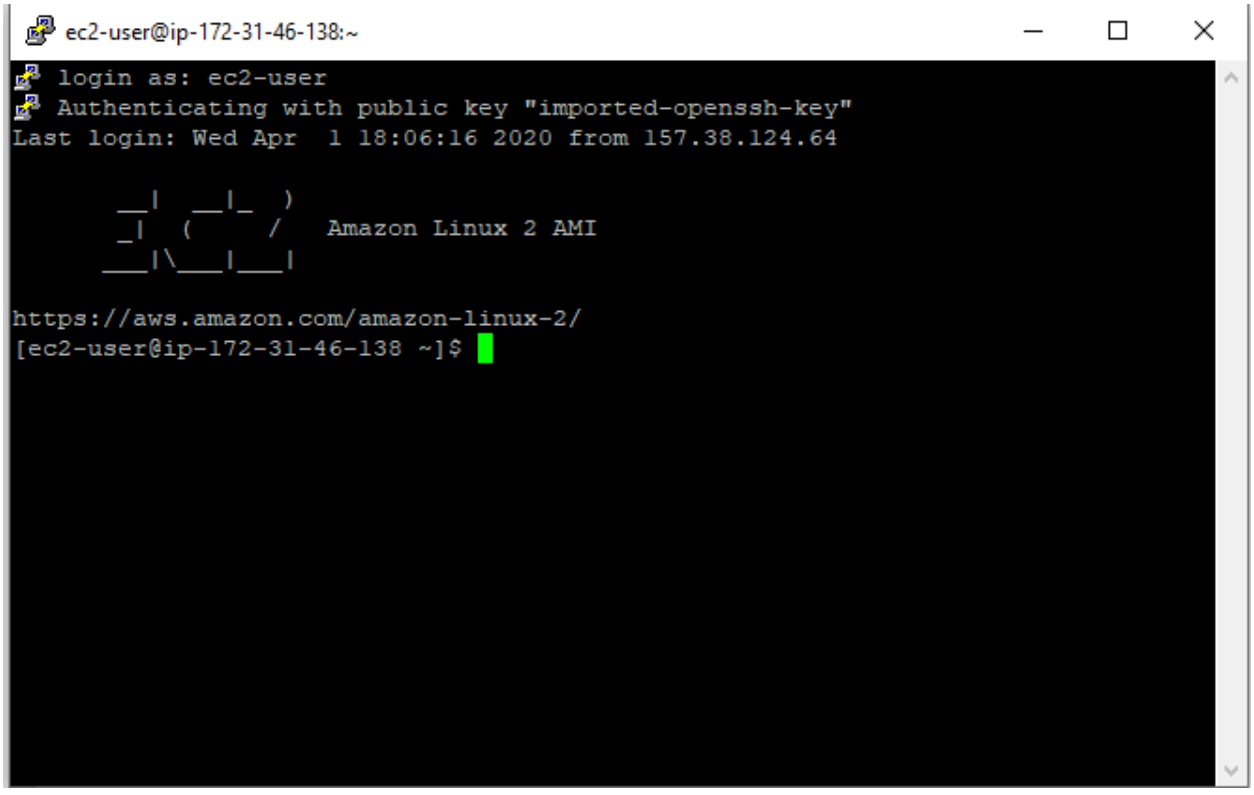
Key-loaded now save it as private key face-ethus

11.) Logged in Screen:

My Instance:

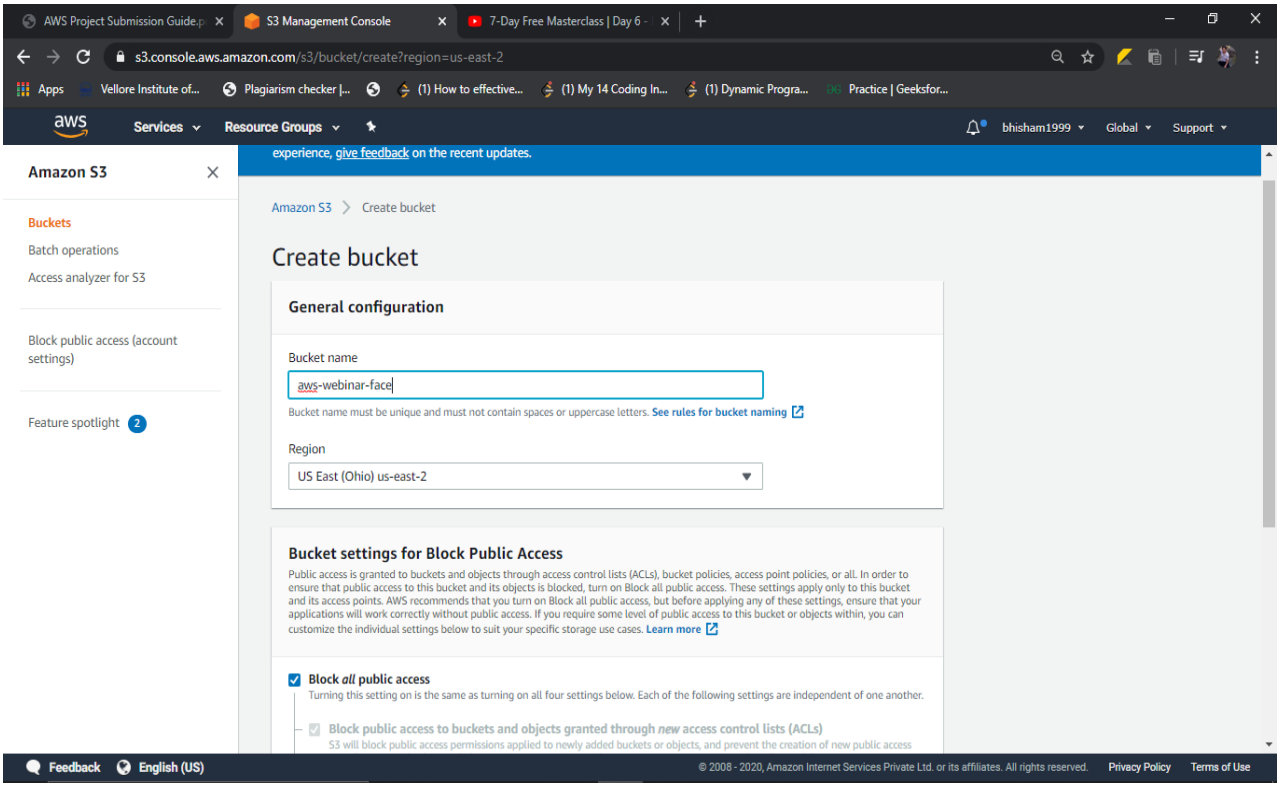


Now Copying the ip address in host in putty and passing ppk key pair in Auth under SSH.

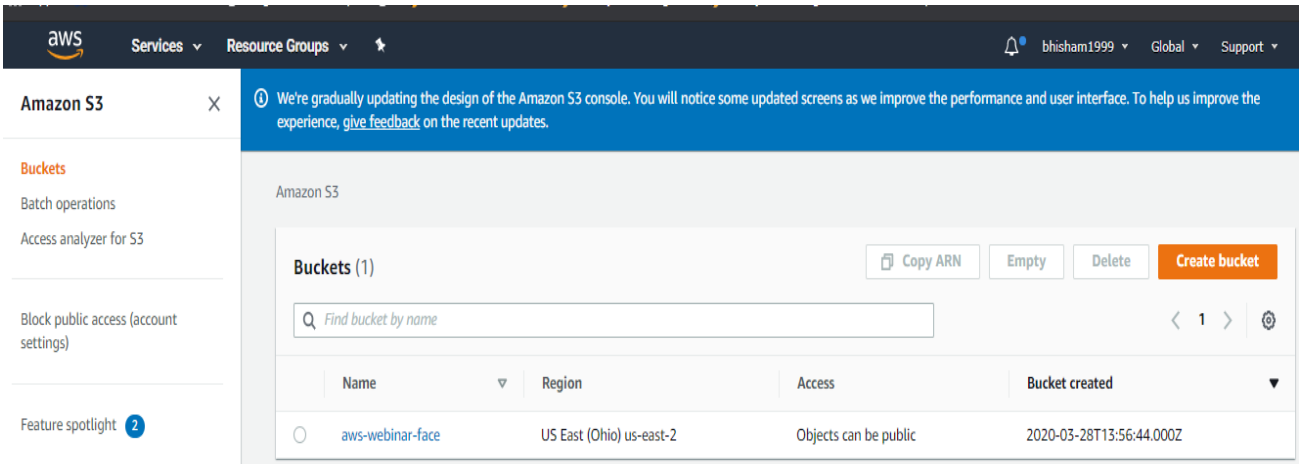


Screenshots needed for S3:

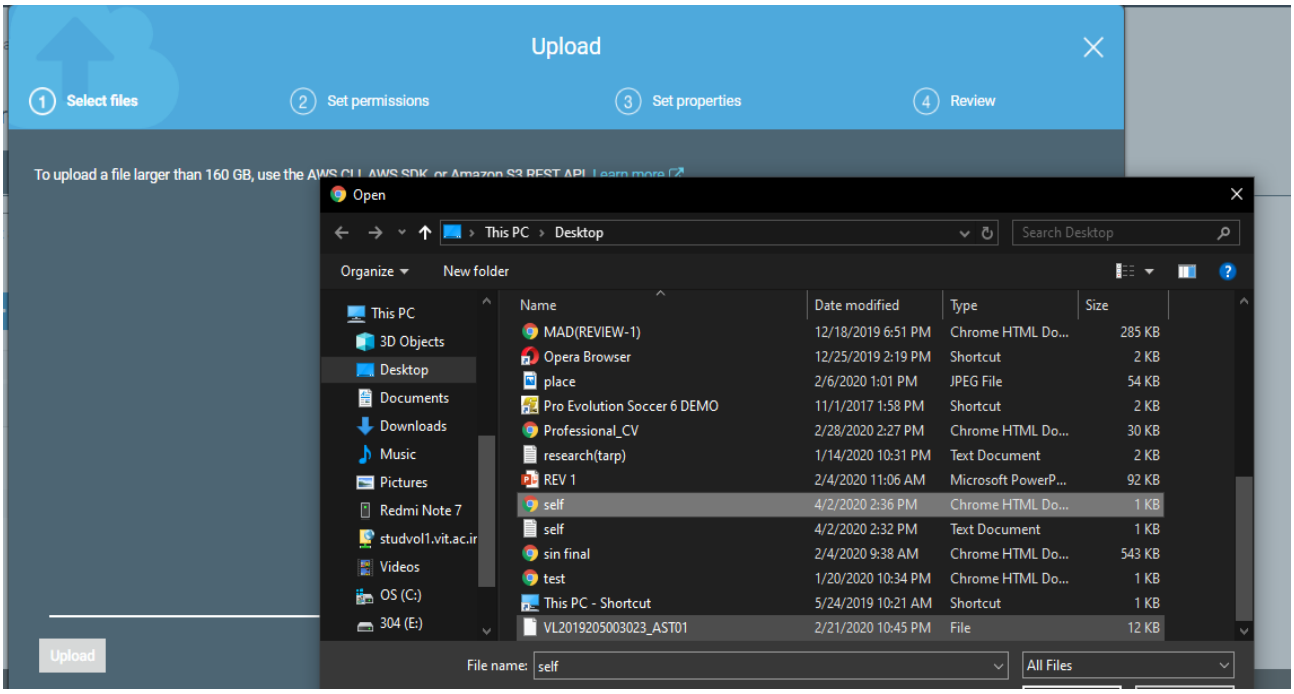
12.) Creating a bucket

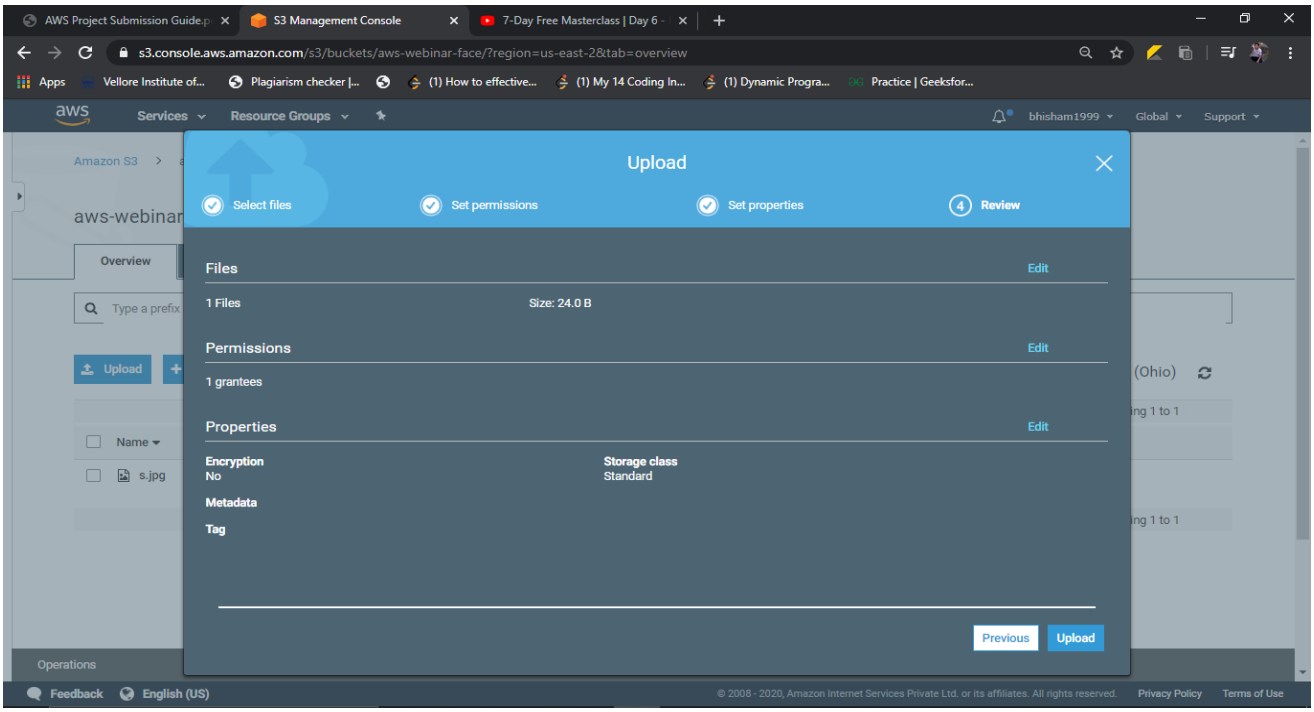


aws-webinar-face (bucket created)

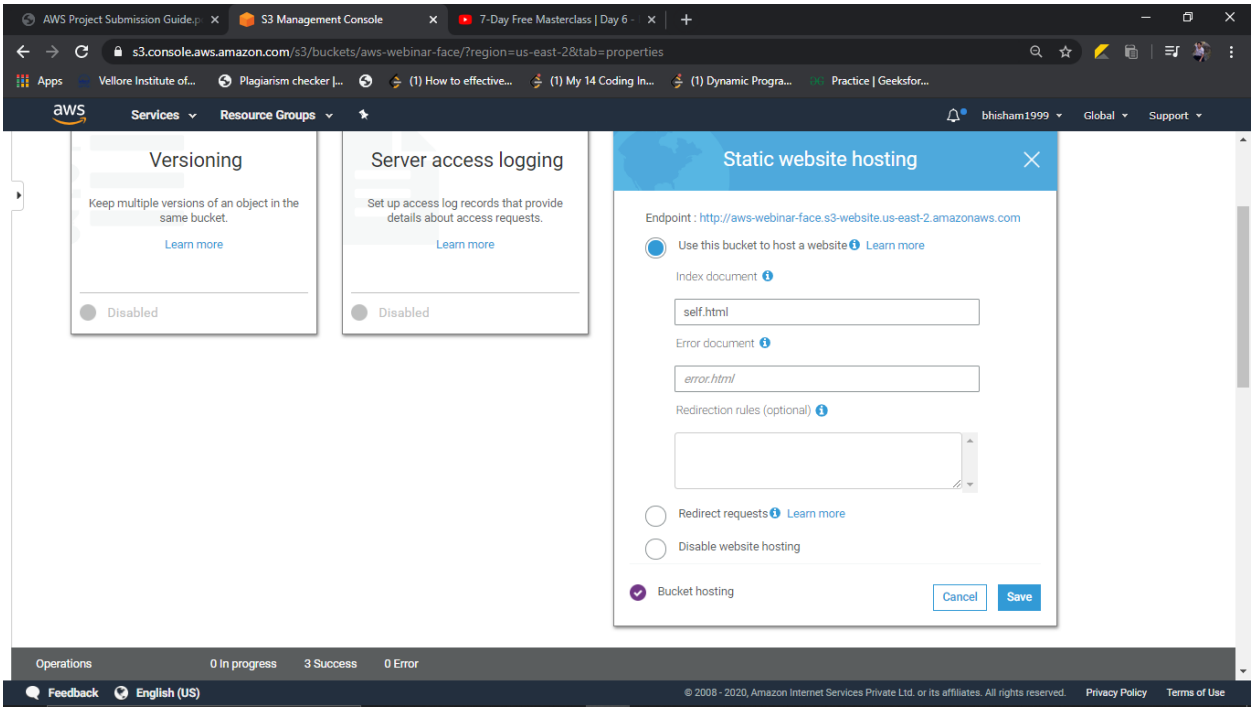


13.) Uploading a File

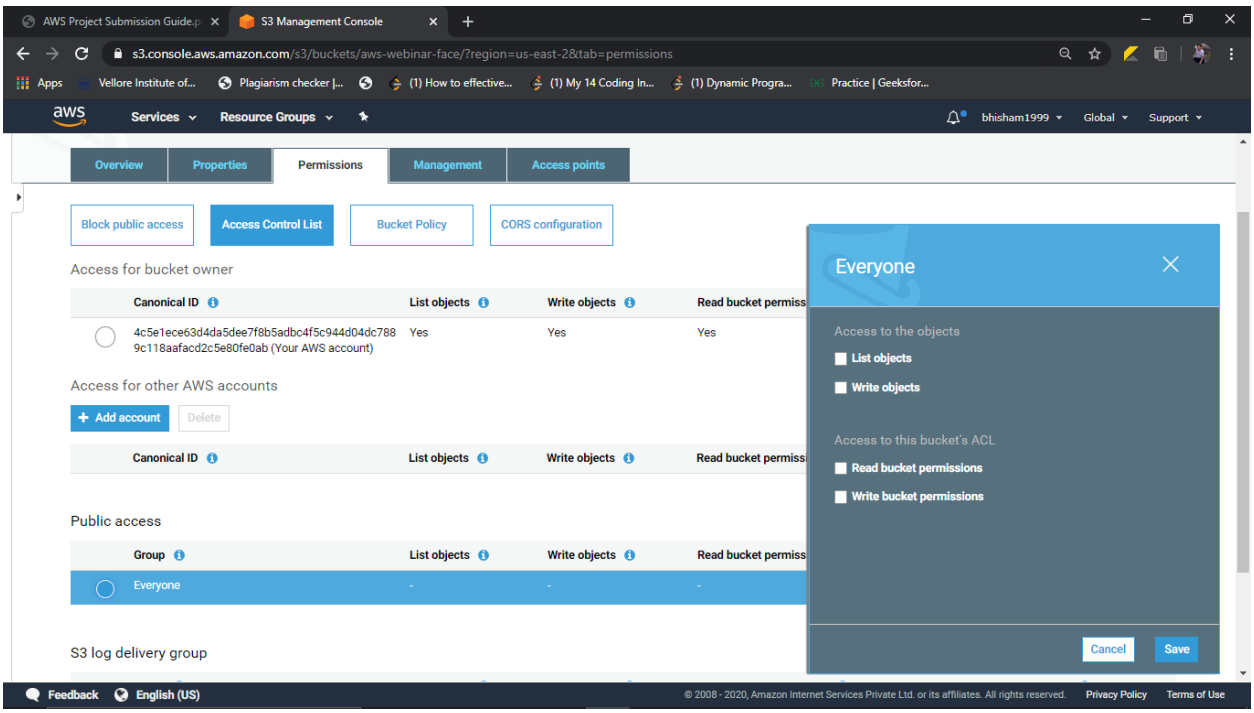


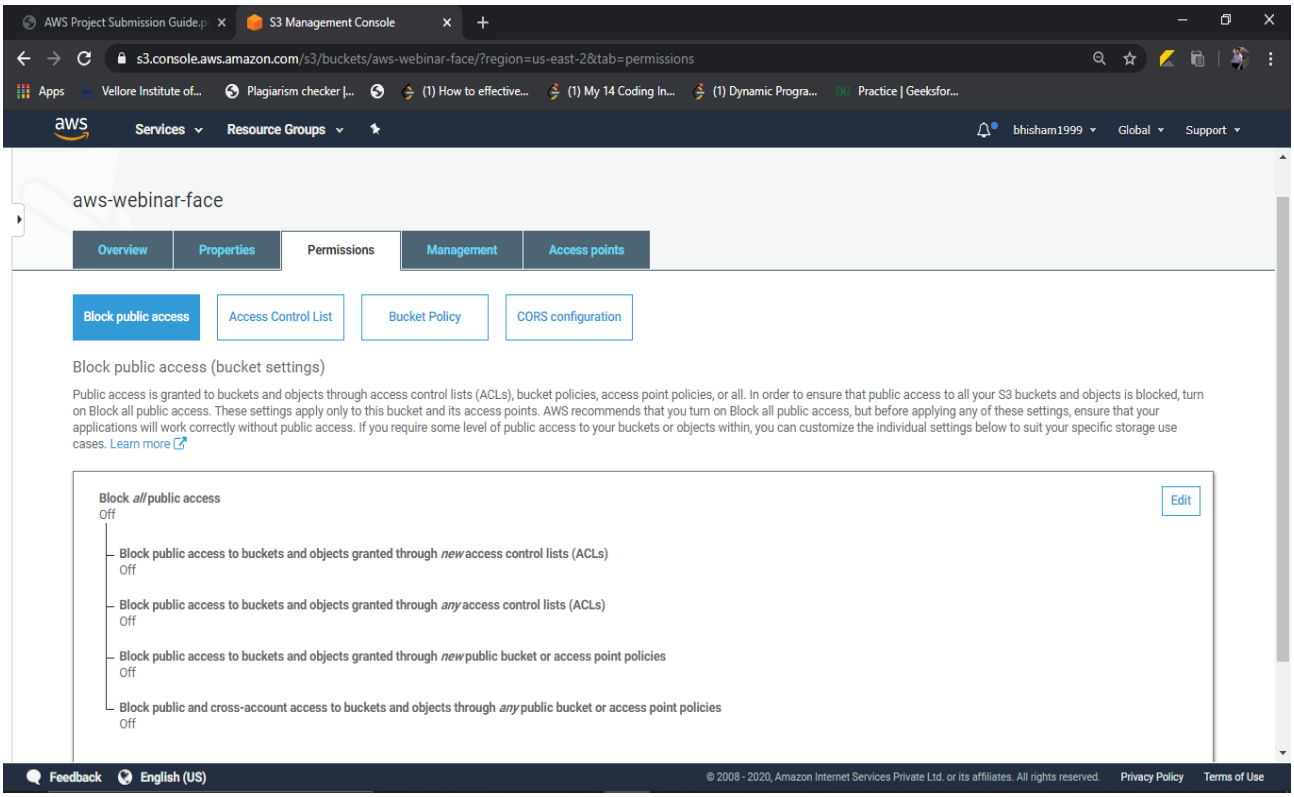


14.) Static website hosting

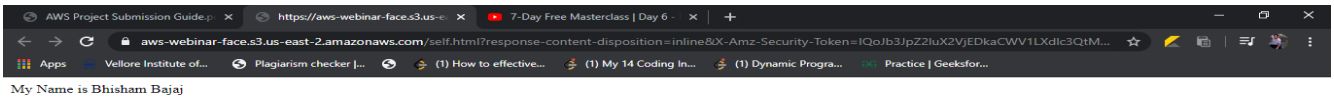


15.) Public access



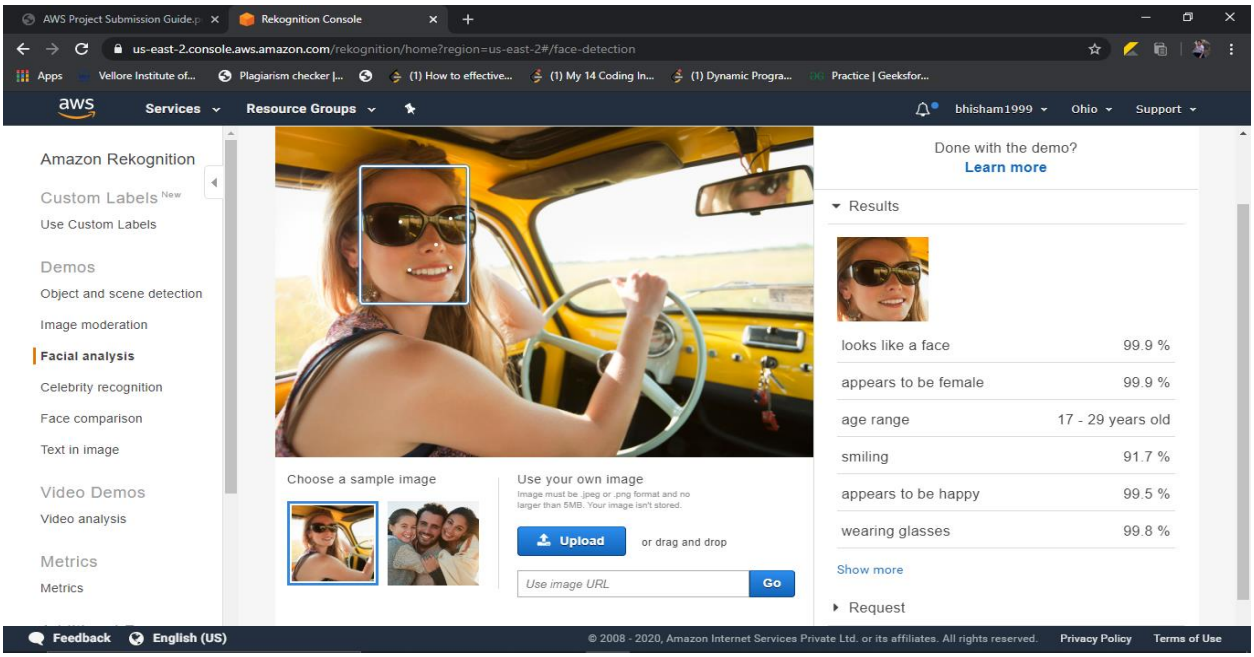


16.) Checking the link on Browser (My static website)

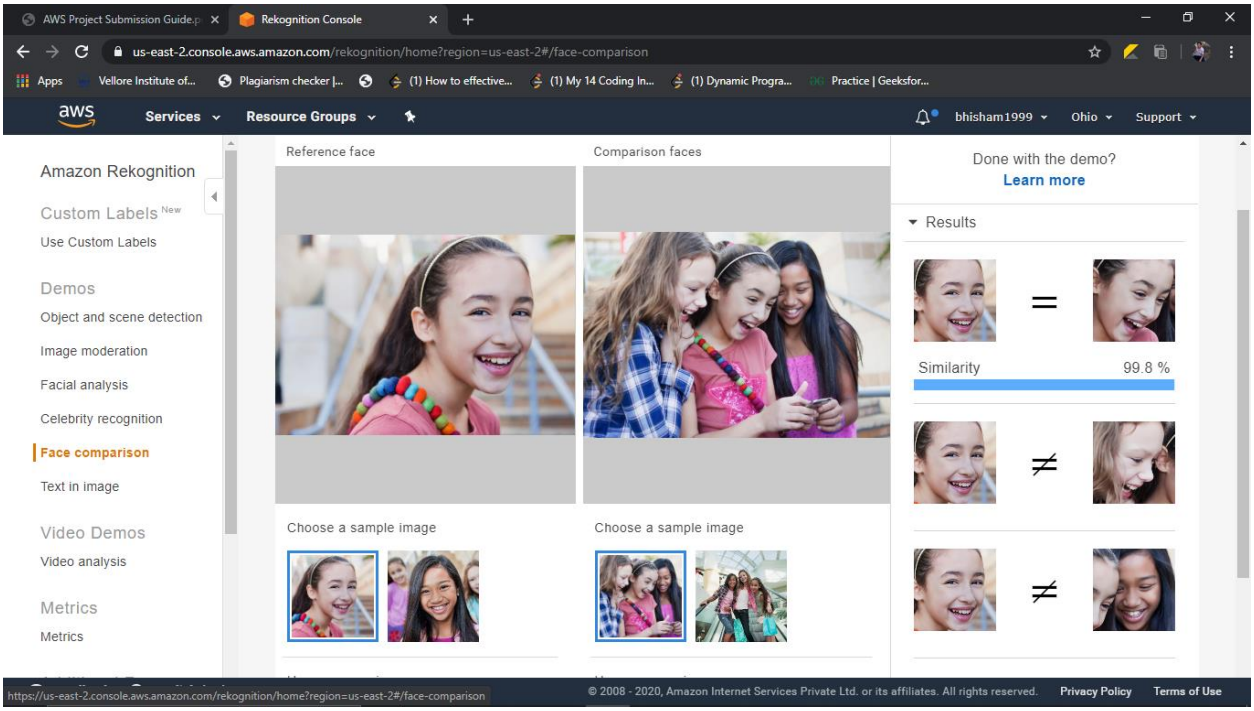


Screenshots needed for Rekognition:

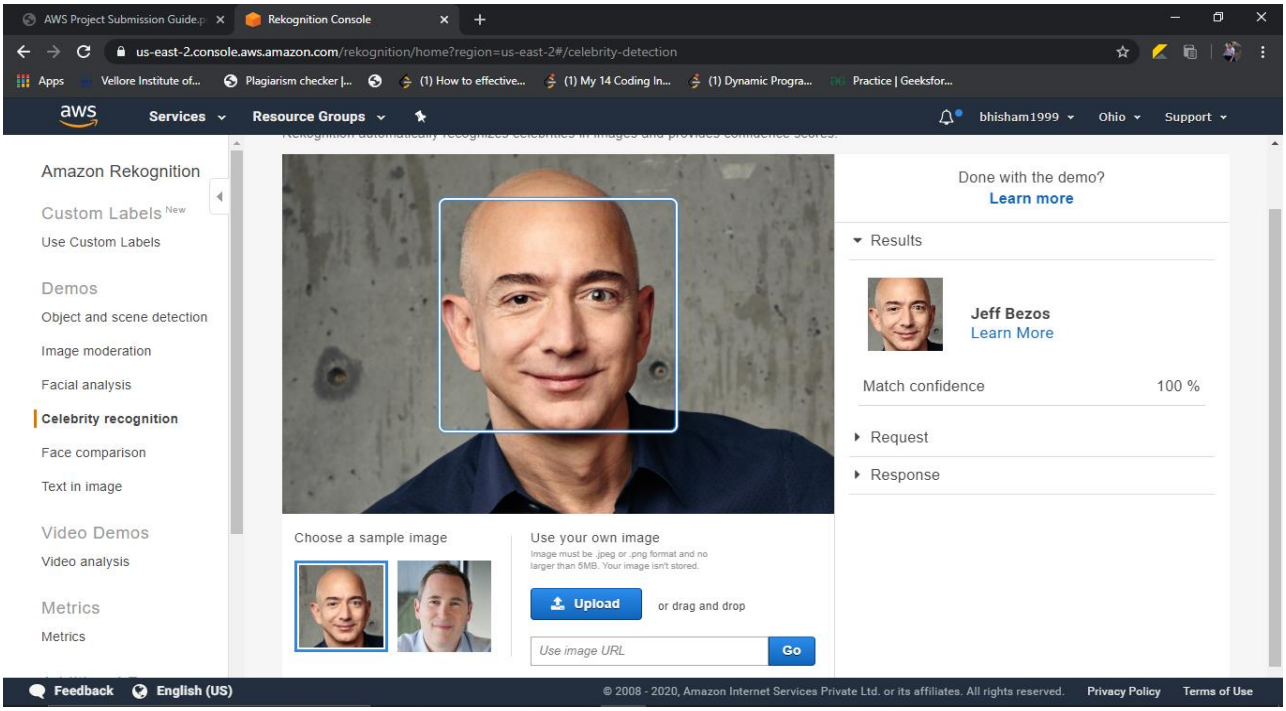
17.) Face Detection



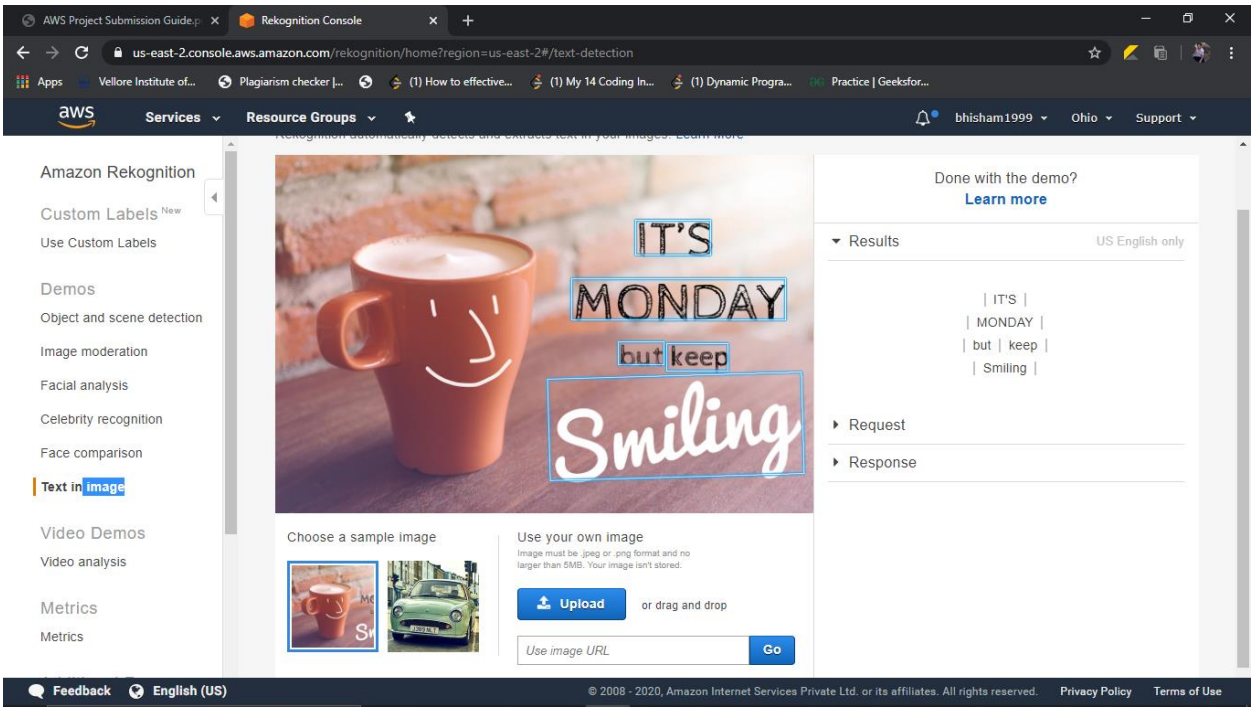
18.) Face comparison



19.) Celebrity Recognition



20.) Text in Image:



Screenshots needed for EC2 & S3:

21.) Installing aws-sdk:

```
ec2-user@ip-172-31-46-138:~
 login as: ec2-user
 Authenticating with public key "imported-openssh-key"
Last login: Thu Apr  2 08:56:34 2020 from 157.39.218.70

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-46-138 ~]$ sudo php -d memory_limit=-1 ~/composer.phar requi
re aws/aws-sdk-php
Using version ^3.134 for aws/aws-sdk-php
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 8 installs, 0 updates, 0 removals
  - Installing symfony/polyfill-mbstring (v1.15.0): Loading from cache
  - Installing mtdowling/jmespath.php (2.5.0): Loading from cache
  - Installing guzzlehttp/promises (v1.3.1): Loading from cache
  - Installing ralouphie/getallheaders (3.0.3): Loading from cache
  - Installing psr/http-message (1.0.1): Loading from cache
  - Installing guzzlehttp/psr7 (1.6.1): Loading from cache
  - Installing guzzlehttp/guzzle (6.5.2): Loading from cache
  - Installing aws/aws-sdk-php (3.134.1): Downloading (100%)
guzzlehttp/psr7 suggests installing zendframework/zend-httphandlerrunner (Emit P
SR-7 responses)
guzzlehttp/guzzle suggests installing psr/log (Required for using the Log middle
ware)
guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized D
omain Name (IDN) support)
aws/aws-sdk-php suggests installing doctrine/cache (To use the DoctrineCacheAdap
ter)
aws/aws-sdk-php suggests installing aws/aws-php-sns-message-validator (To valida
te incoming SNS notifications)
Writing lock file
Generating autoload files
1 package you are using is looking for funding.
Use the `composer fund` command to find out more!
[ec2-user@ip-172-31-46-138 ~]$
```

22.) Installing Php

```
ec2-user@ip-172-31-46-138:~
[ec2-user@ip-172-31-46-138 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 2.4 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package php.x86_64 0:7.2.28-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package     Arch          Version                 Repository                Size
================================================================================
Installing:
 php         x86_64        7.2.28-1.amzn2          amzn2extra-php7.2        2.9 M

Transaction Summary
================================================================================
Install  1 Package

Total download size: 2.9 M
Installed size: 9.1 M
Is this ok [y/d/N]: y
```

23.) Code File:

ec2-user@ip-172-31-46-138:~

```
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 = 'aws-webinar-face';
$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;
}
"index.php" 55L, 1223C                                                  55,1          Bot
```

24.) Image upload Done..

ec2-user@ip-172-31-46-138:/var/www/html/face

```
[ec2-user@ip-172-31-46-138 ~]$ cd /var/www/html
[ec2-user@ip-172-31-46-138 html]$ cd face
[ec2-user@ip-172-31-46-138 face]$ pwd
/var/www/html/face
[ec2-user@ip-172-31-46-138 face]$ ls
composer.json  composer.lock  index.php  s.jpg  vendor
[ec2-user@ip-172-31-46-138 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar-face.s3.us-east-2.amaz
```

25.) EC2-Recognition (facial detection)

```
[ec2-user@ip-172-31-46-138 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-webinar-face.s3.us-east-2.amazonaws.com/s.jpgTotally there are 9 faces[ec2-user@ip-172-31-46-138 face]$
```