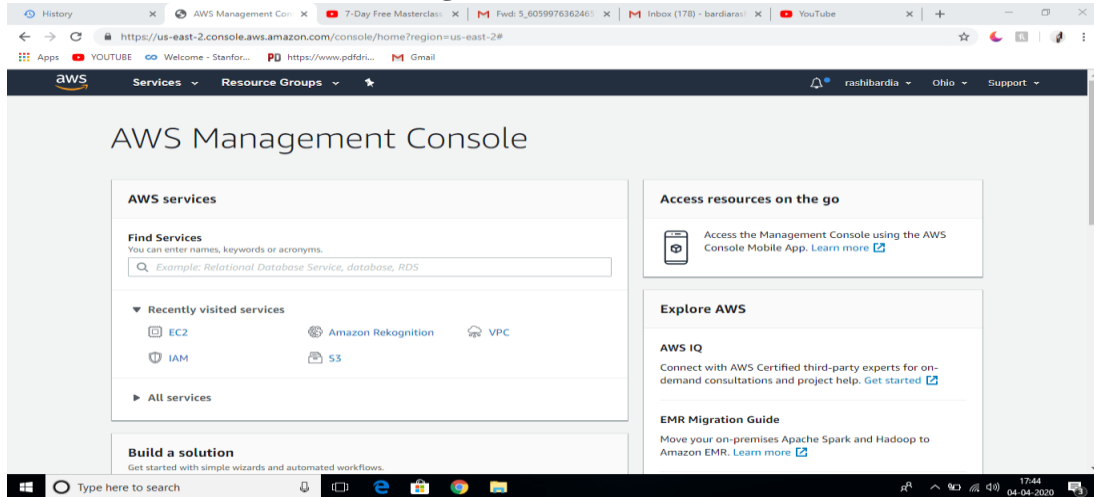
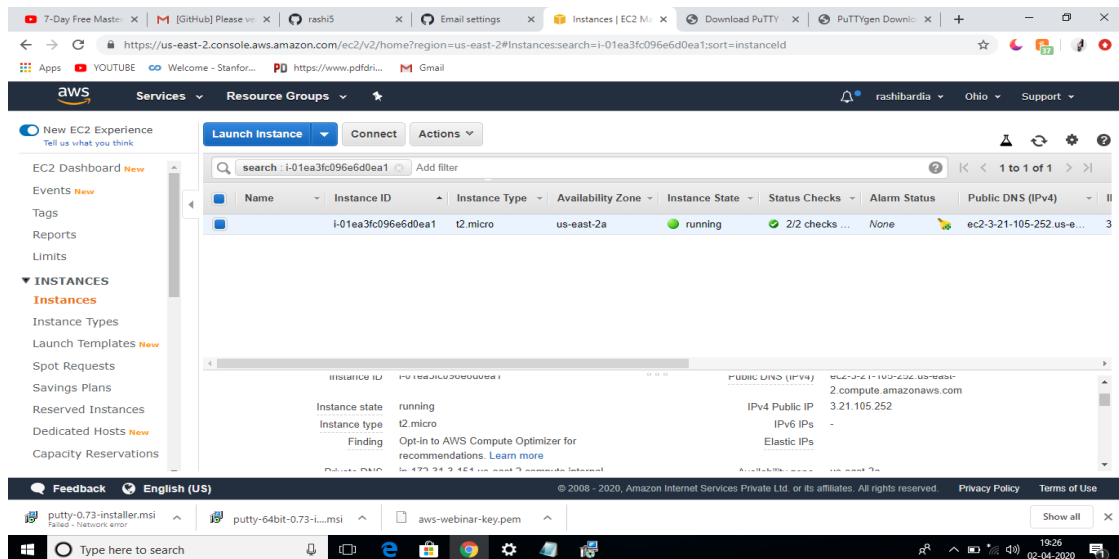


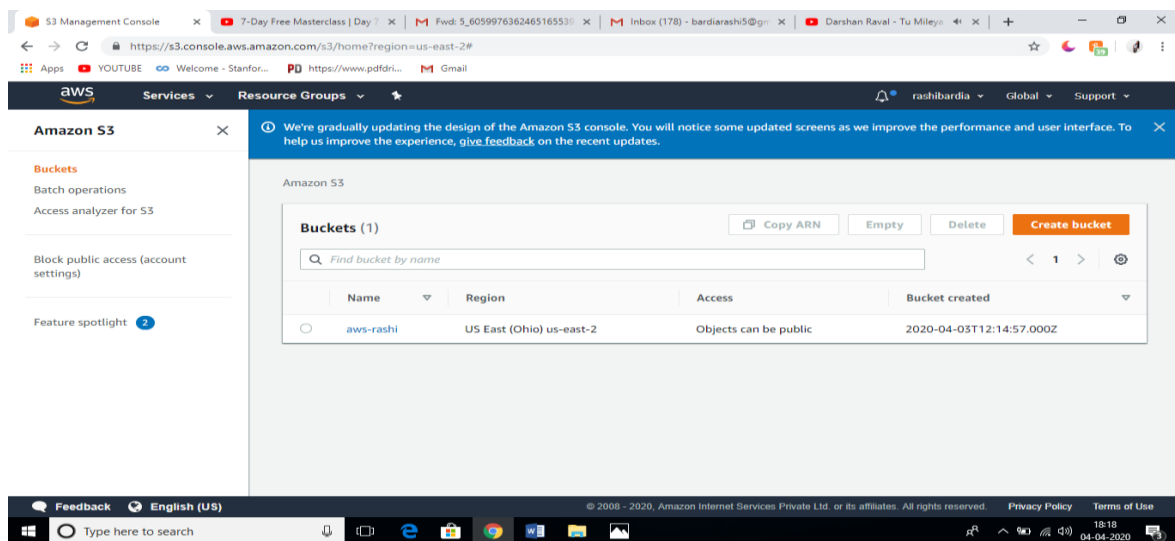
## 1. AWS Login screen with username



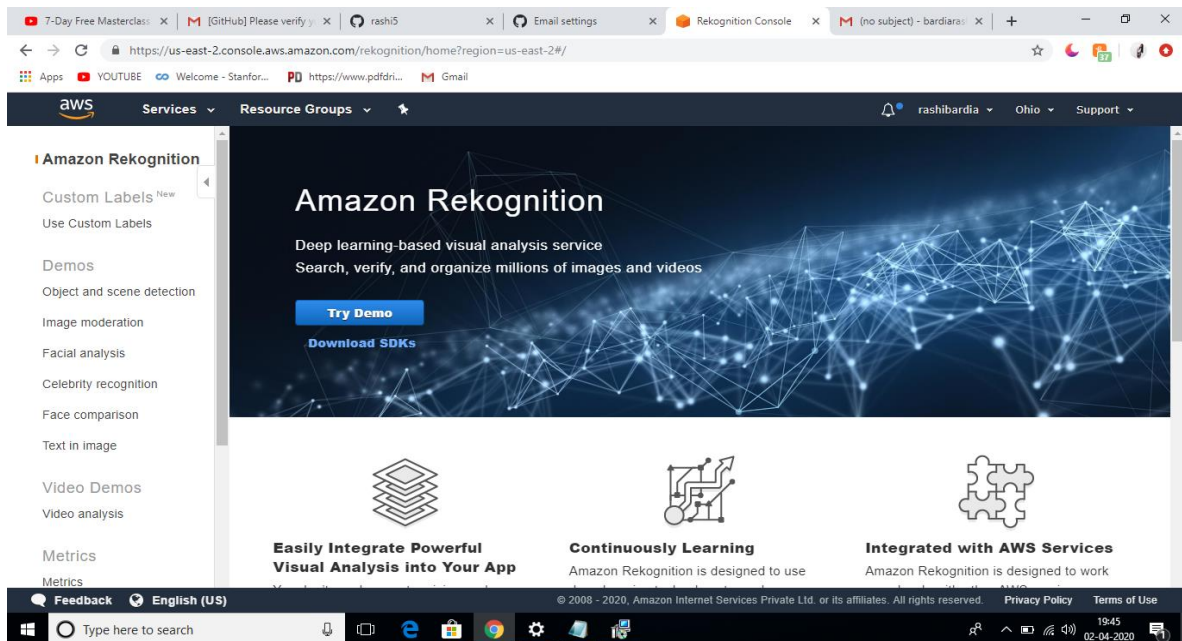
## 2. EC2 Dashboard



## 3. S3 Dashboard

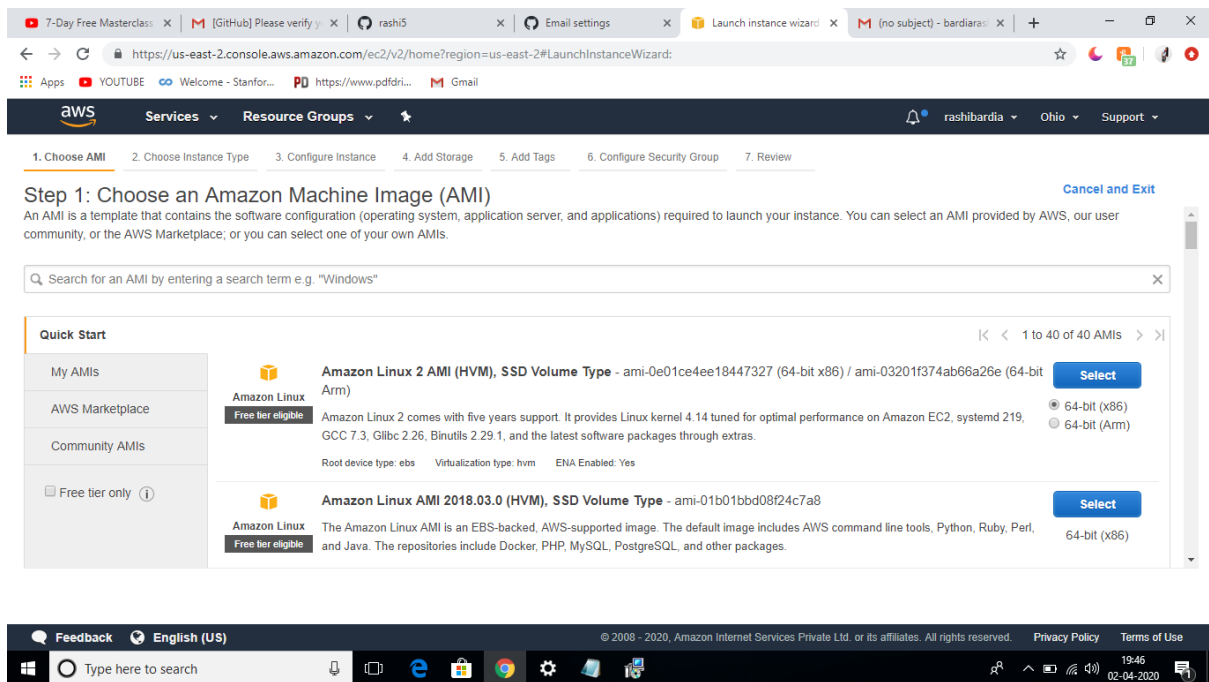


## 4. Rekognition Dashboard

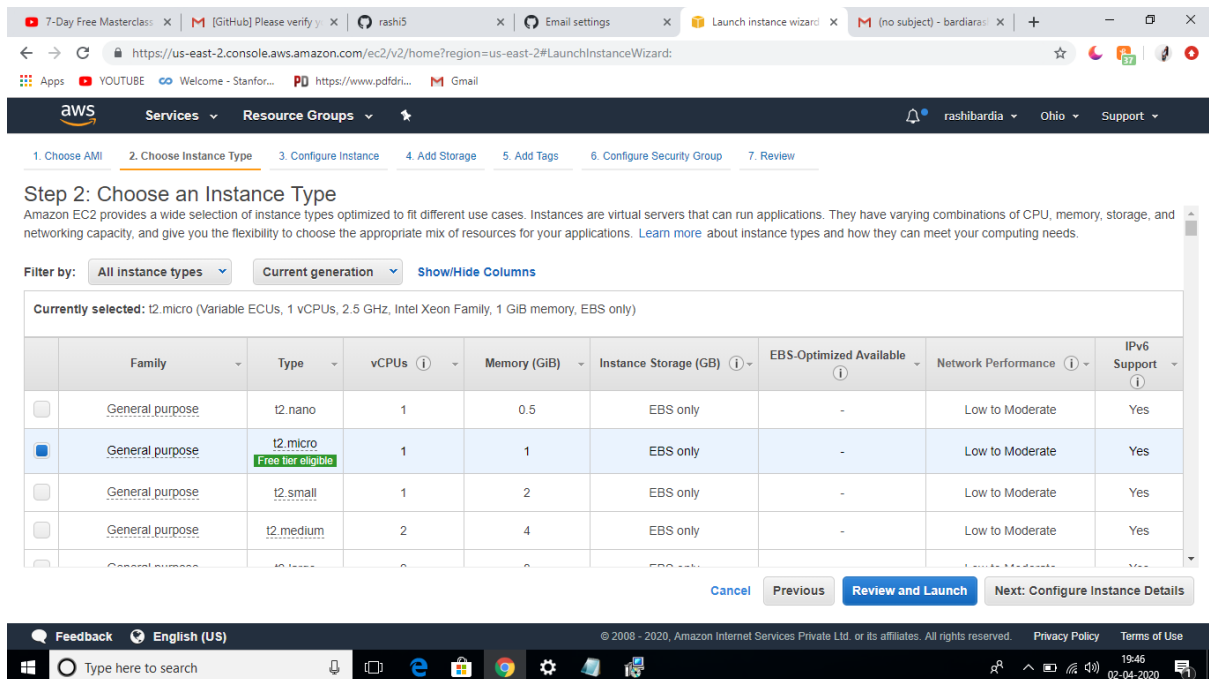


## Screenshots needed for EC2

### 1. Choosing an AMI



## 2. Choosing an Instance Type



**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.

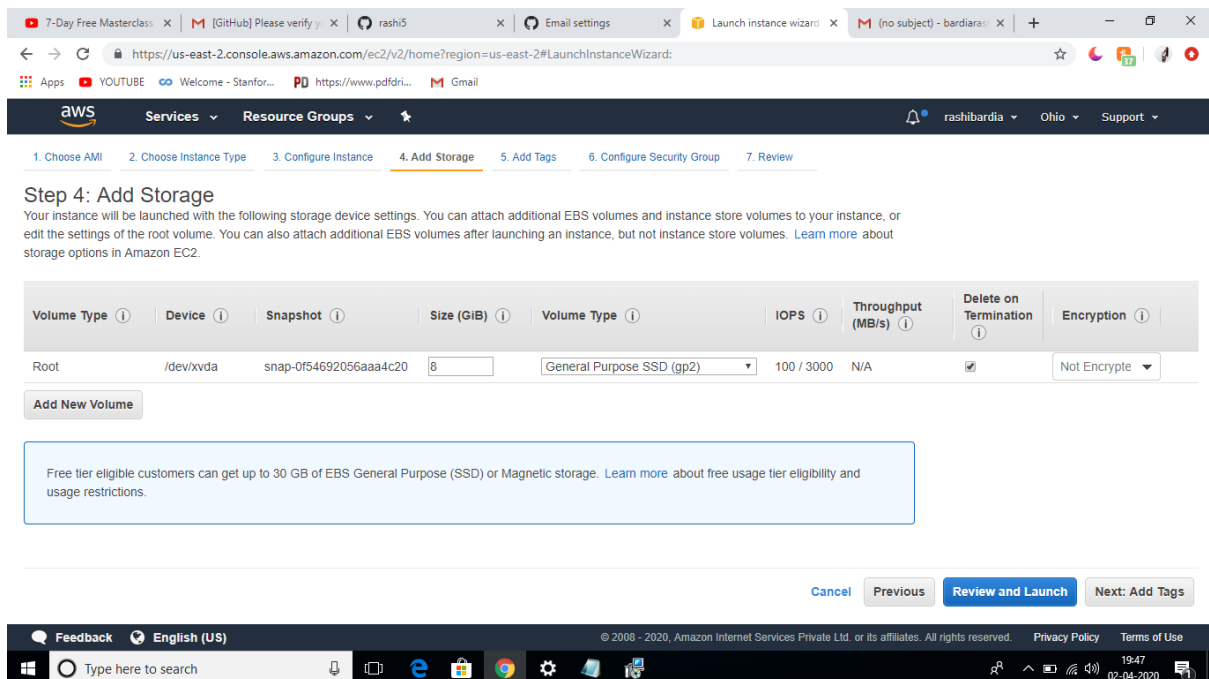
Filter by: **All instance types** **Current generation** [Show/Hide Columns](#)

**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
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

## 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 Encrypt

[Add New Volume](#)

Free tier eligible customers can get up to 30 GiB 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:   
Description:

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

Step 7: Review Instance Launch

**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](#).

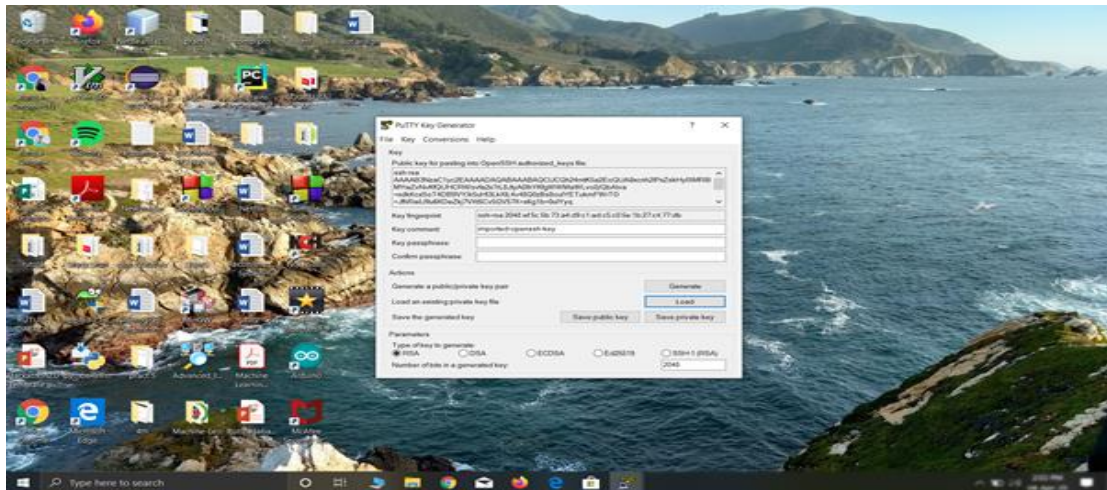
Create a new key pair ☒ ☐ Select an existing key pair

Key pair name:  [Download Key Pair](#)

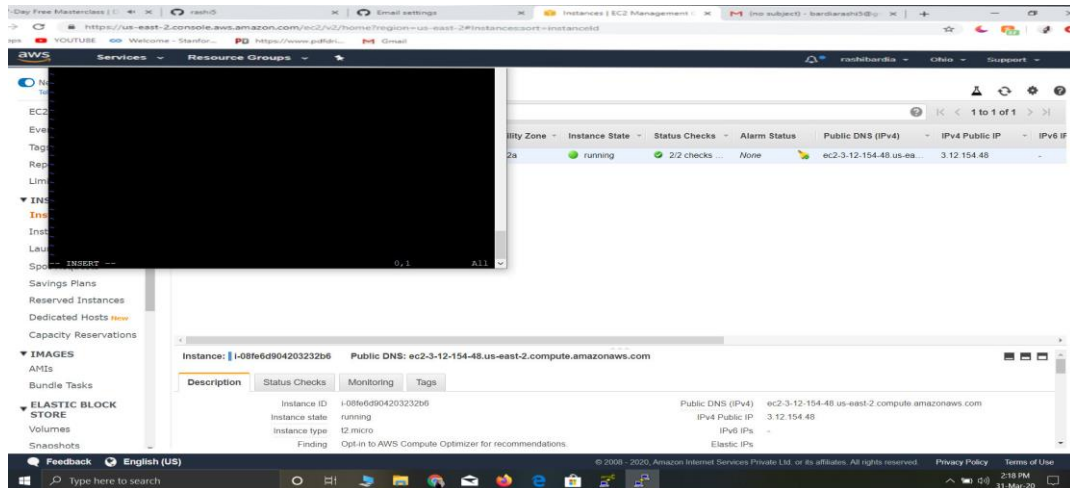
**You have to download the private key file (\*.pem file) before you can continue. Store it in a secure and accessible location. You will not be able to download the file again after it's created.**

[Cancel](#) [Launch Instances](#)

## 6. PuTTYgen conversion from pem to ppk

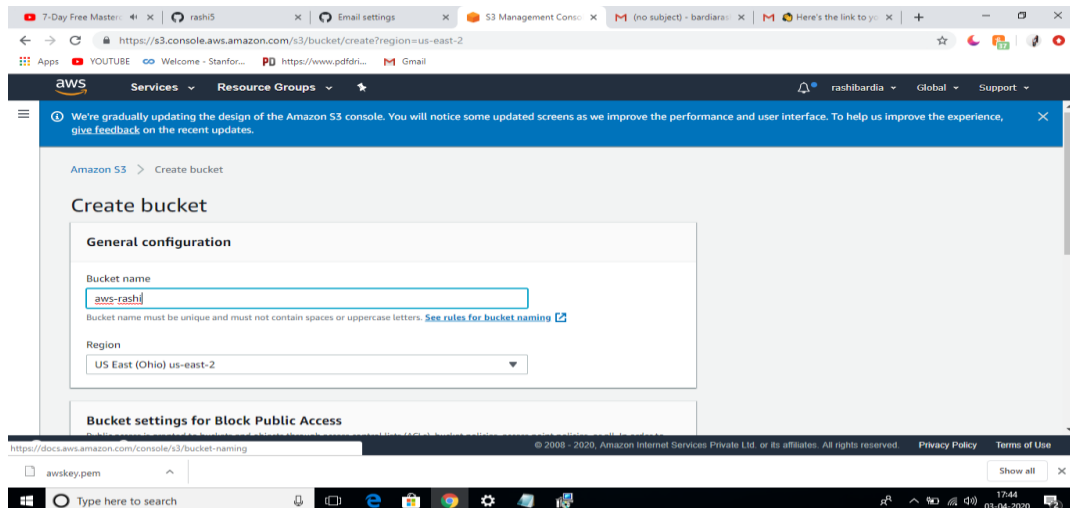


## 7. Logged in EC2 black screen

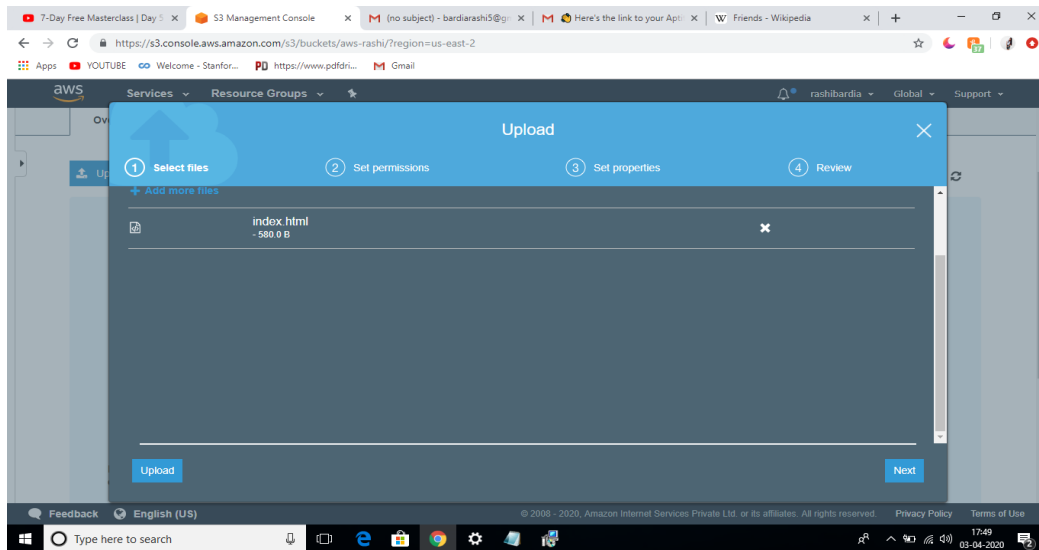


## Screenshots for S3

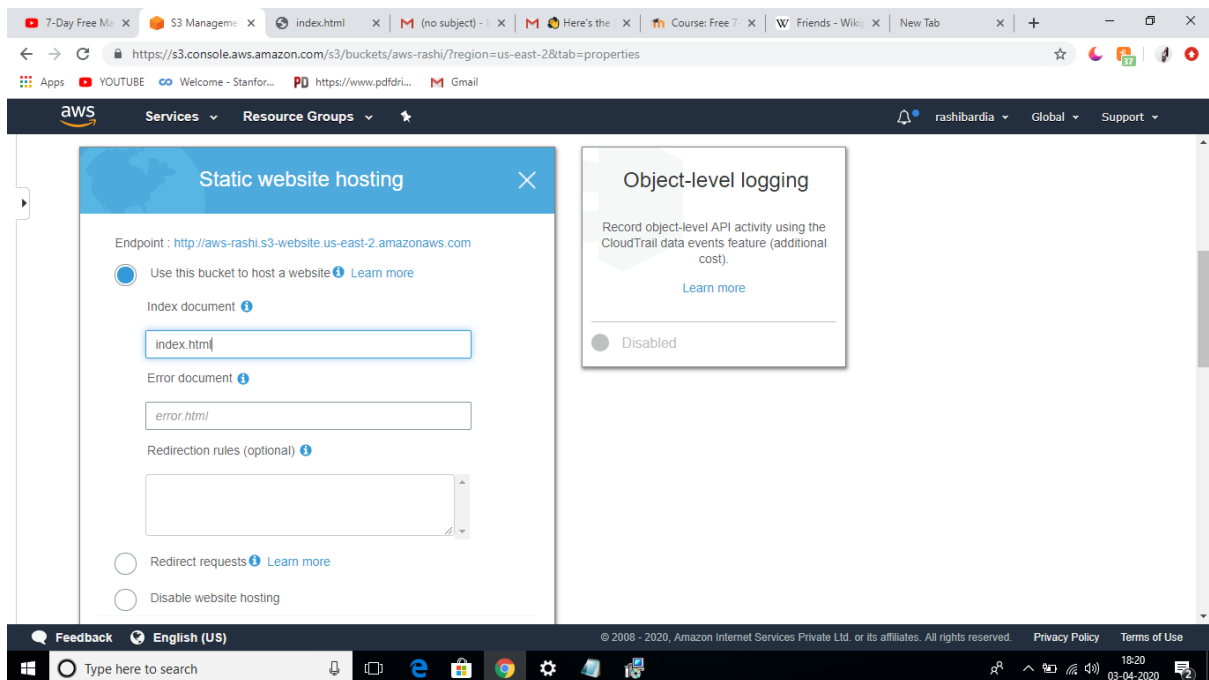
### 1. Creating a Bucket



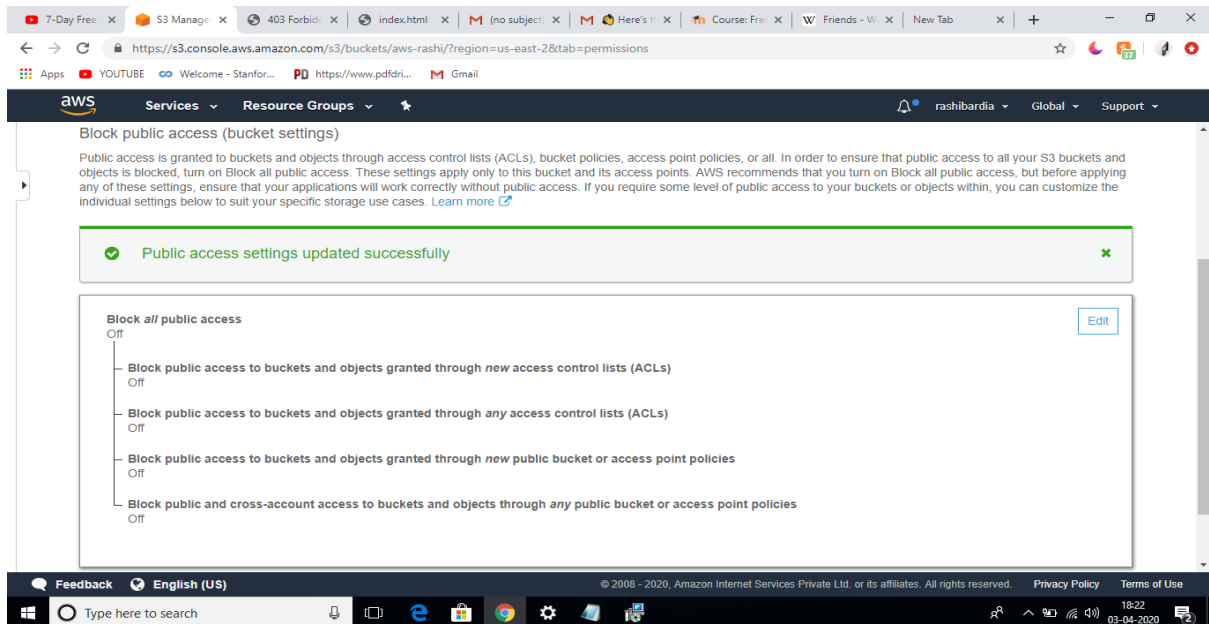
## 2. Uploading an object



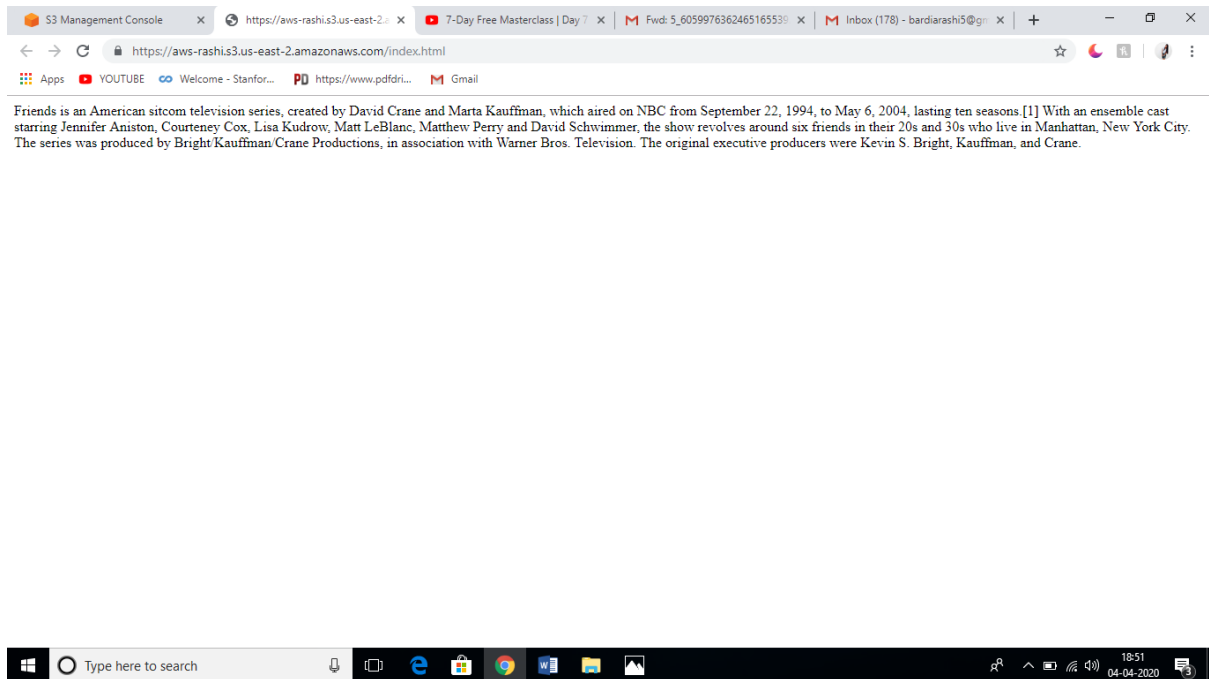
## 3. Enabling Static Website



## 4. Making the Object Public



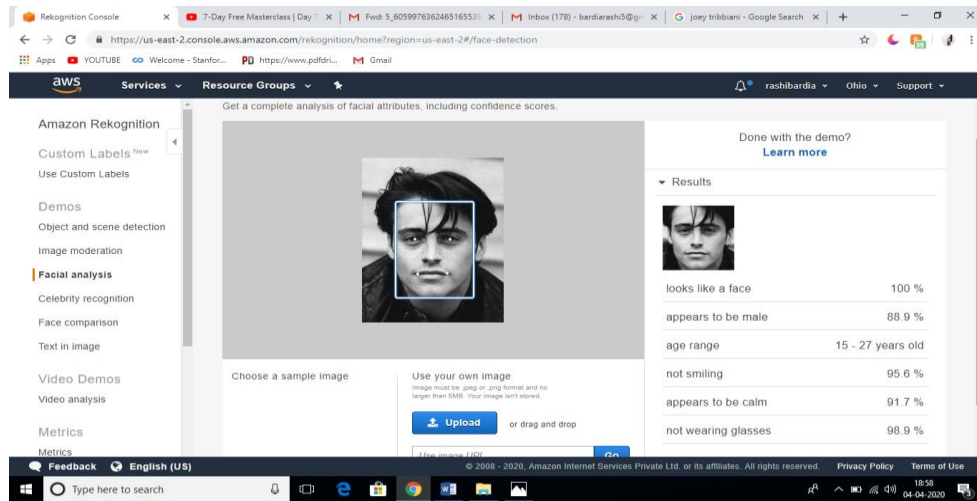
## 5. Checking the S3 link on the browser



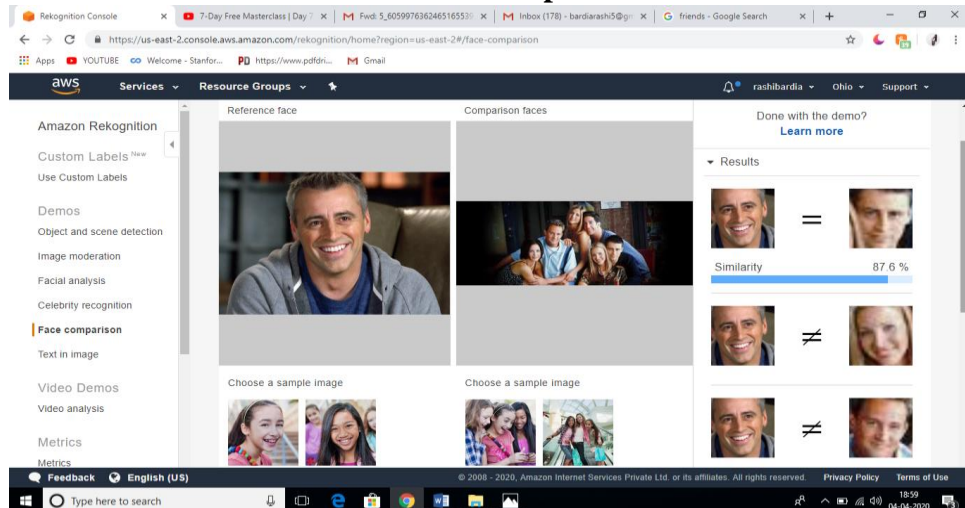


# SCREENSHOTS NEEDED FOR REKOGNITION

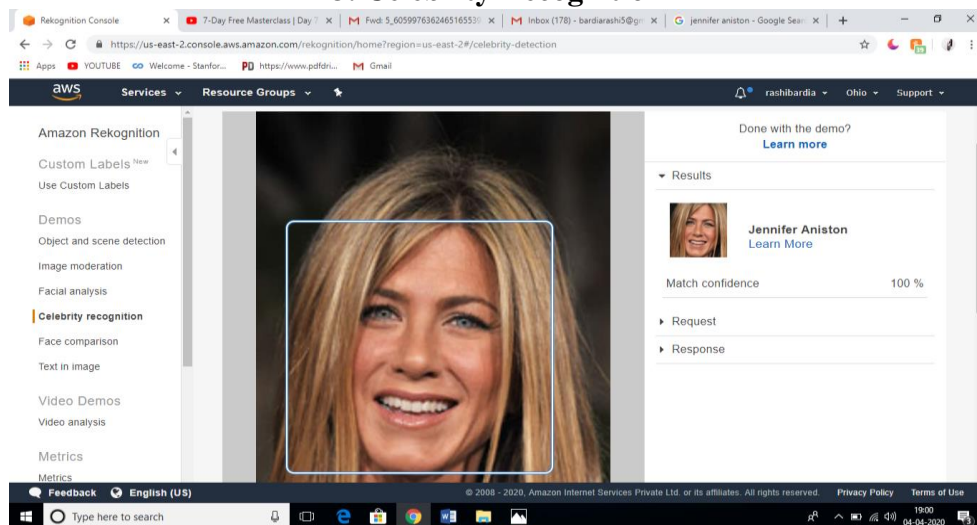
## 1. Face Detect



## 2. Face Compare

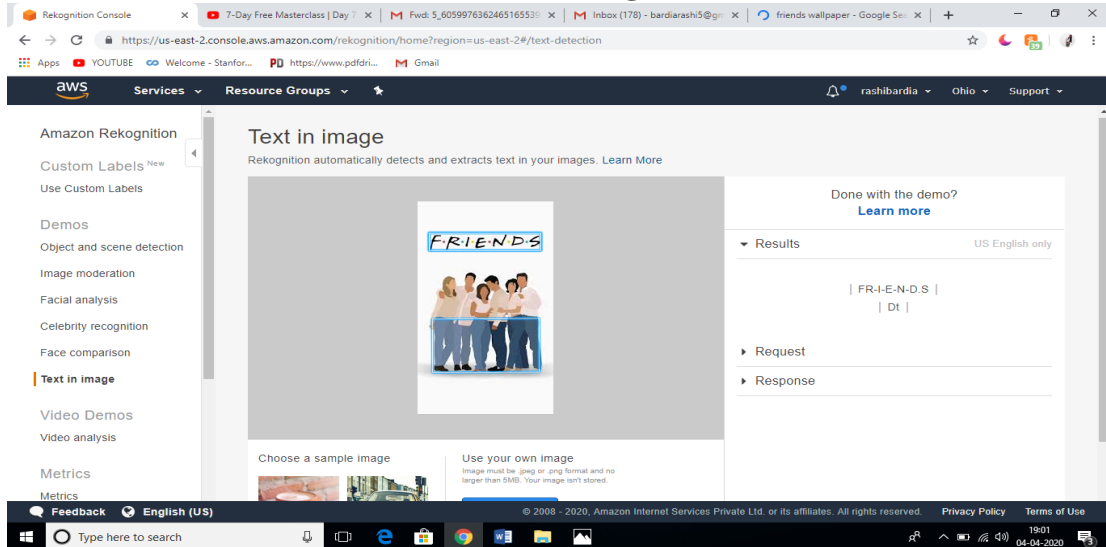


## 3. Celebrity Recognition





## 2. Text in Image



## Screenshots needed for EC2 & S3

### 1. Installing aws-sdk

```
ec2-user@ip-172-31-35-213:/var/www/html/face
kernel.x86_64 0:4.14.173-137.229.amzn2

Updated:
  gmpg2.x86_64 0:2.0.22-5.amzn2.0.4
  langtable.noarch 0:0.0.31-4.amzn2
  langtable-data.noarch 0:0.0.31-4.amzn2
  langtable-python.noarch 0:0.0.31-4.amzn2
  libfastjson.x86_64 0:0.95.4-3.amzn2
  libtirpc.x86_64 0:0.2.4-0.16.amzn2

Complete!
[ec2-user@ip-172-31-35-213 ~]$ sudo yum upgrade
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-172-31-35-213 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package php-5.4.16-46.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-35-213 ~]$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 1.10.1) successfully installed to: /home/ec2-user/composer.phar
Use it: php composer.phar

[ec2-user@ip-172-31-35-213 ~]$ cd /var/www/html
[ec2-user@ip-172-31-35-213 html]$ cd face
[ec2-user@ip-172-31-35-213 face]$ sudo php -d memory_limit=-1 ~/composer.phar re
quire aws/aws-sdk-php
Using version ^2.3 for aws/aws-sdk-php
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zle instead.
Generating autoload files
[ec2-user@ip-172-31-35-213 face]$ ^C
[ec2-user@ip-172-31-35-213 face]$ ls
composer.json  linkin-park-billboard-650-compressed.jpg  s.jpg
composer.lock  linkin-park.jpg?w=900                     vendor
index.php      sample.jpg
[ec2-user@ip-172-31-35-213 face]$ sudo php index.php
PHP Fatal error:  Argument 1 passed to Aws\Common\Client\AbstractClient
::__construct() must be an instance of Aws\Common\Credentials\CredentialsInterf
ace, array given, called in /var/www/html/face/index.php on line 46 and defined
in /var/www/html/face/vendor/aws/aws-sdk-php/src/Aws/Common/Client/AbstractClie
nt.php on line 73
[ec2-user@ip-172-31-35-213 face]$
```

## 2. Installing php

```
2. installing php
kernel.x86_64 0:4.14.173-131.223.amzn2

Updated:
  gnupg2.x86_64 0:2.0.22-5.amzn2.0.4
  langtable.noarch 0:0.0.31-4.amzn2
  langtable-data.noarch 0:0.0.31-4.amzn2
  langtable-python.noarch 0:0.0.31-4.amzn2
  libfastjson.x86_64 0:0.99.4-3.amzn2
  libtirpc.x86_64 0:0.2.4-0.16.amzn2

Complete!
[ec2-user@ip-172-31-35-213 ~]$ sudo yum upgrade
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-172-31-35-213 ~]$ sudo yum install php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package php-5.4.16-46.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-35-213 ~]$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 1.10.1) successfully installed to: /home/ec2-user/composer.phar
Use it: php composer.phar

[ec2-user@ip-172-31-35-213 ~]$ cd /var/www/html
[ec2-user@ip-172-31-35-213 html]$ cd face
[ec2-user@ip-172-31-35-213 face]$ sudo php -d memory_limit=-1 ~/composer.phar re
quire aws/aws-sdk-php
Using version ^2.8 for aws/aws-sdk-php
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/gu
zle instead.
Installing command-line
[ec2-user@ip-172-31-35-213 face]$ ^C
[ec2-user@ip-172-31-35-213 face]$ ls
composer.json  linkin-park-billboard-650-compressed.jpg  s.jpg
composer.lock  linkin-park.jpg?w=800                     vendor
index.php      sample.jpg
[ec2-user@ip-172-31-35-213 face]$ sudo php index.php
PHP Catchable fatal error:  Argument 1 passed to Aws\Common\Client\AbstractClie
nt::__construct() must be an instance of Aws\Common\Credentials\CredentialsInterf
ace, array given, called in /var/www/html/face/index.php on line 46 and defined
in /var/www/html/face/vendor/aws/aws-sdk-php/src/Aws/Common/Client/AbstractClie
nt.php on line 73
[ec2-user@ip-172-31-35-213 face]$
```

## 3. index.php file code

```
ec2-user@ip-172-31-40-112:/var/www/html/face
aws Aws\Rekognition\RekognitionClient;

$bucket = 'aws-web-07';
$keyname = 's.jpg';

$s3 = S3Client::factory([
    'region' => 'us-east-2',
    'version' => '2006-03-01',
    'signature' => 'v4'
]);

try {
    // Upload image
    $result = $s3->putObject([
        'bucket' => $bucket,
        'key' => $keyname,
        'SourceFile' => __DIR__ . "/" . $keyname,
        'ACL' => 'public-read-write'
    ]);

    // Return the URL to the object
    $imageUrl = $result['ObjectURL'];
    if($imageUrl) {
        echo "Image upload done... Here is the URL: " . $imageUrl;

        $rekognition = RekognitionClient::factory([
            'region' => 'us-east-2',
            'version' => 'latest',
        ]);

        $result = $rekognition->detectFaces([
            'Image' => [
                'uri' => $imageUrl,
            ],
        ]);

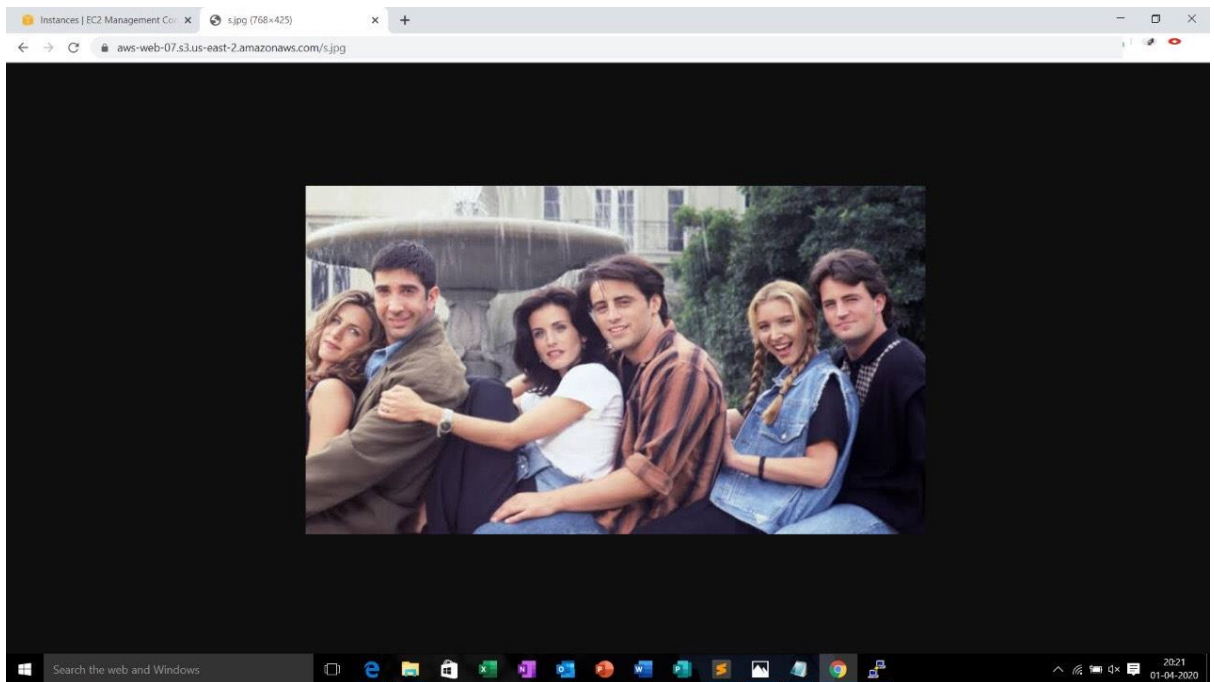
        echo "Totally there are " . count($result['FaceDetails']) . " Faces";
    }
} catch (Exception $e) {
    echo $e->getMessage() . PHP_EOL;
}
```

## 4. Upload success screenshot

1.

```
ec2-user@ip-172-31-40-112:~$ curl http://www.html5face.com
ec2-user@ip-172-31-40-112:~$ cd /var/www/html/
ec2-user@ip-172-31-40-112:~$ sudo vim index.php
ec2-user@ip-172-31-40-112:~$ sudo php index.php
Image upload done... Here is the URL: https://aws-web-07.s3.us-east-2.amazonaws.com/s.jpg[ec2-user@ip-172-31-40-112:~]$
```

2.



## Screenshots needed for EC2 & Rekognition

### 1. Face Detect success screenshot

```
ec2-user@ip-172-31-40-112:/var/www/html/face
composer.json composer.lock index.php s.jpg vendor
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo php index.php
PHP Catchable fatal error: Argument 1 passed to Aws\Common\Client\AbstractClient
::__construct() must be an instance of Aws\Common\Credentials\CredentialsInterf
ace, array given, called in /var/www/html/face/index.php on line 46 and defined
in /var/www/html/face/vendor/aws/aws-sdk-php/src/Aws/Common/Client/AbstractClien
t.php on line 73
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo php index.php
PHP Catchable fatal error: Argument 1 passed to Aws\Common\Client\AbstractClient
::__construct() must be an instance of Aws\Common\Credentials\CredentialsInterf
ace, array given, called in /var/www/html/face/index.php on line 46 and defined
in /var/www/html/face/vendor/aws/aws-sdk-php/src/Aws/Common/Client/AbstractClien
t.php on line 73
[ec2-user@ip-172-31-40-112 face]$ sudo vim index.php
[ec2-user@ip-172-31-40-112 face]$ sudo php index.php
Image upload done... Here is the URL: https://aws-web-07.s3.us-east-2.amazonaws.
[ec2-user@ip-172-31-40-112 face]$ clear
[ec2-user@ip-172-31-40-112 face]$
```