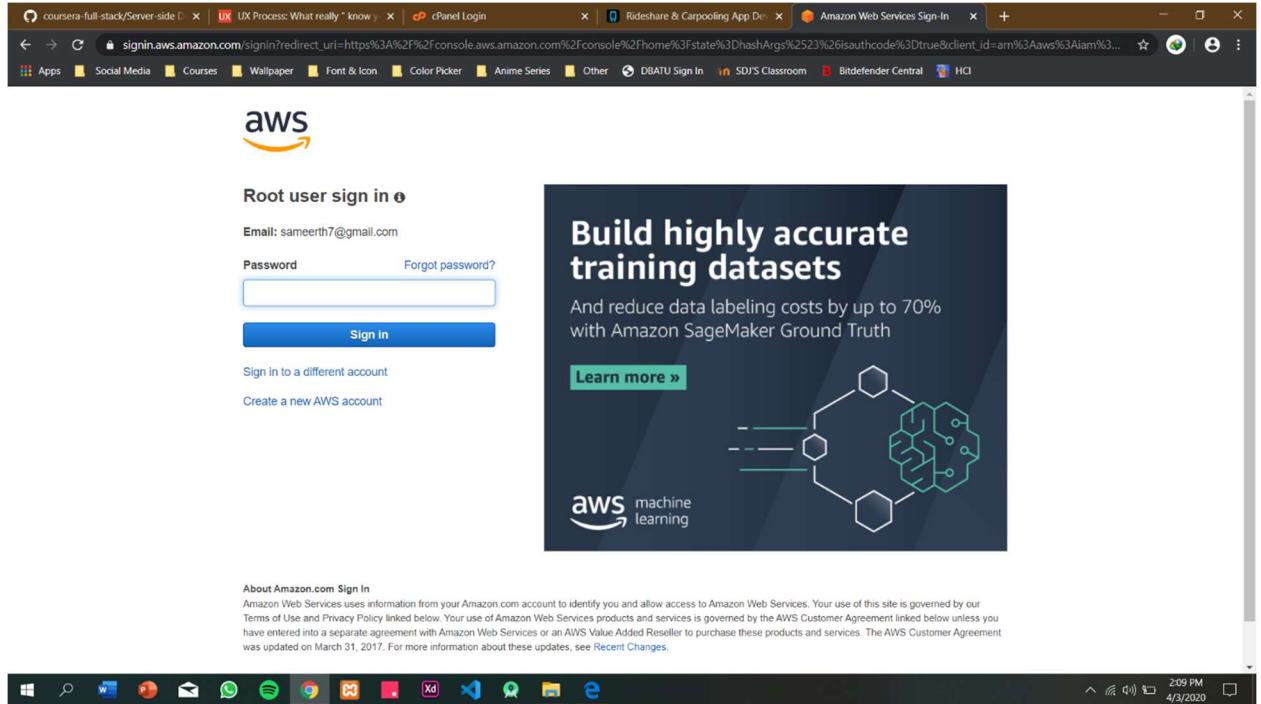


❖ Name : Sameer Thakare

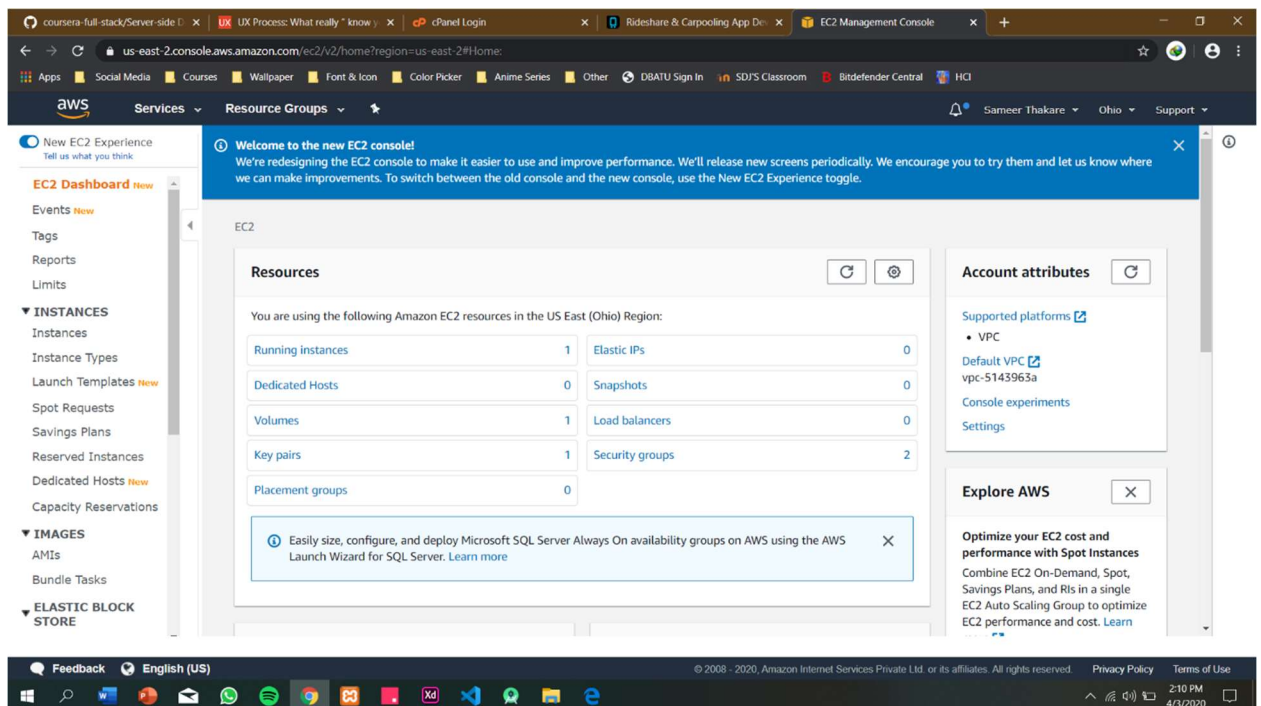
❖ Email Id : sameerth7@gmail.com

Screenshots needed for Dashboards -

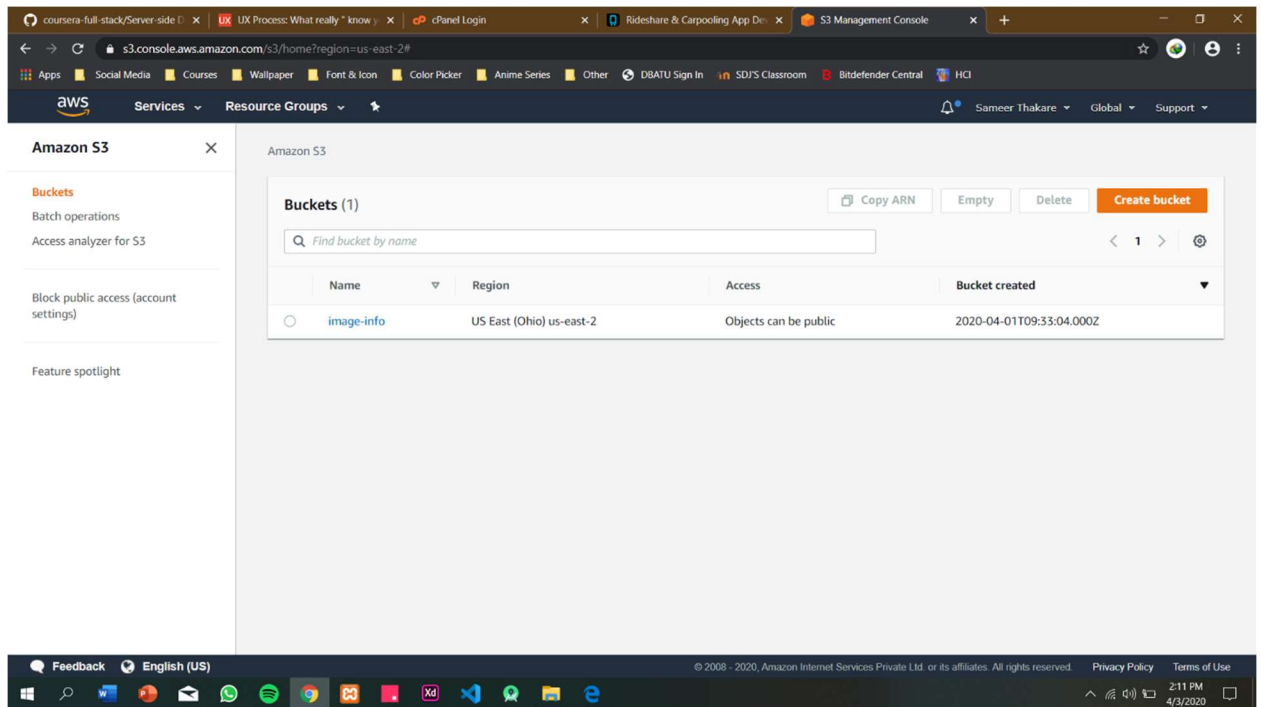
1. AWS Login screen with username



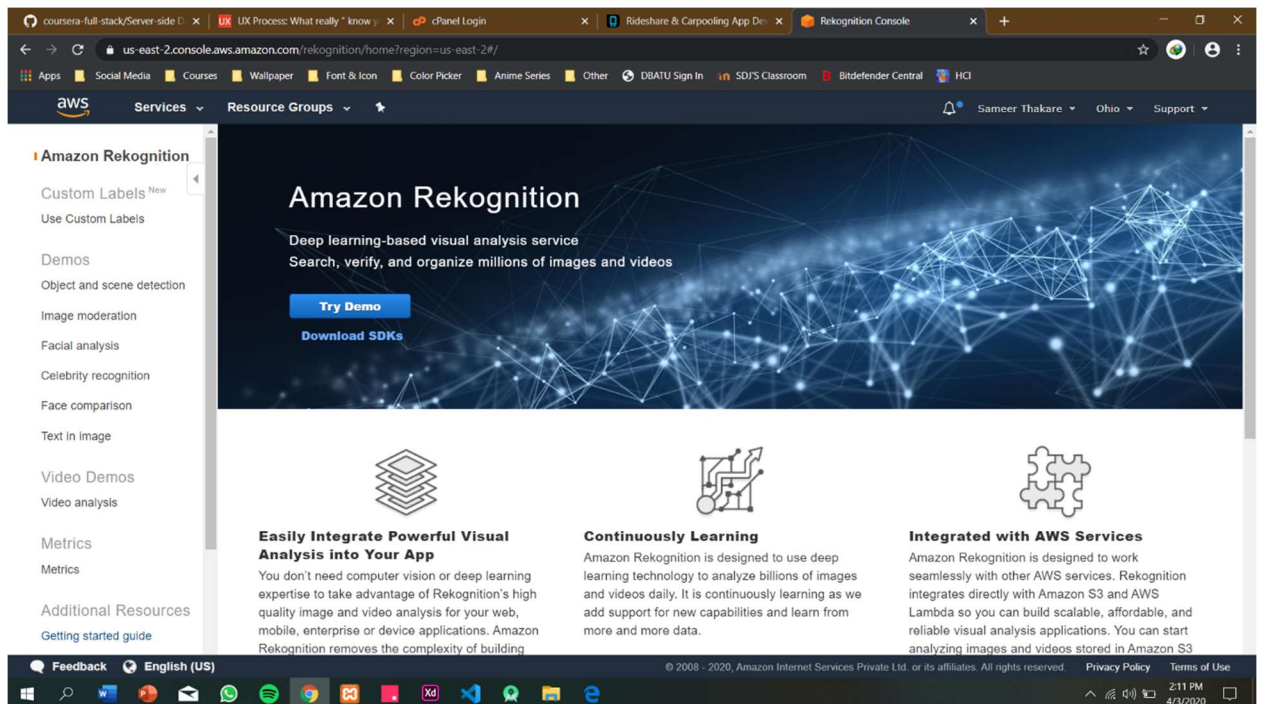
2. EC2 Dashboard



3. S3 Dashboard

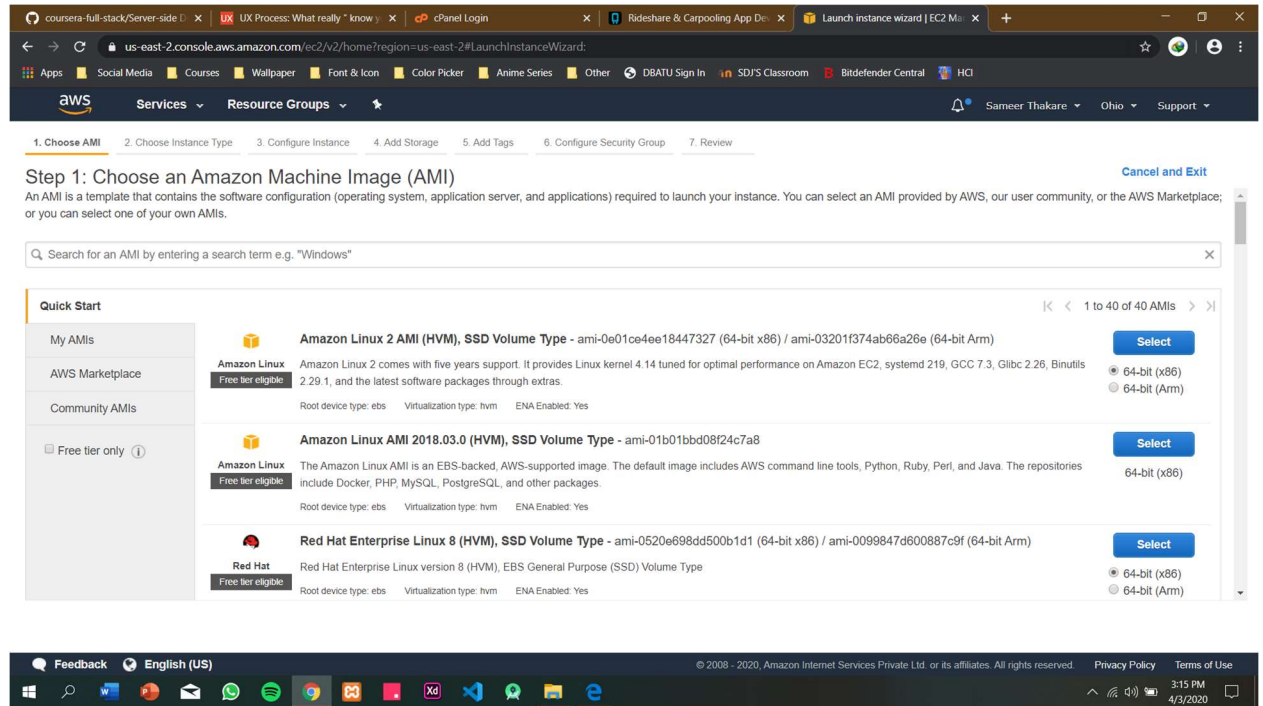


4. Rekognition Dashboard

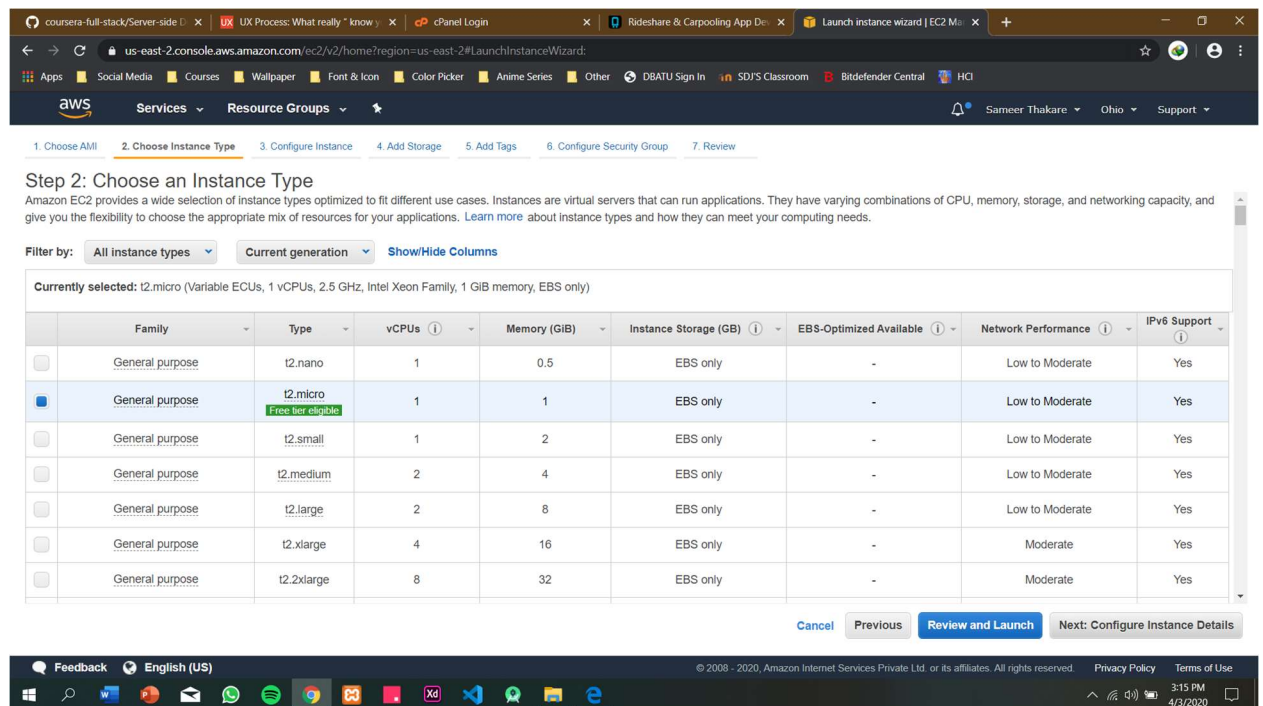


Screenshots needed for EC2

1. Choosing an AMI



2. Choosing an Instance Type



3. Adding Storage

The screenshot shows the 'Add Storage' step of the AWS EC2 Launch Wizard. The breadcrumb trail at the top indicates the current step is '4. Add Storage'. The main heading is 'Step 4: Add Storage'. Below the heading, a paragraph explains that storage device settings can be configured here, including attaching EBS volumes and instance store volumes. A table lists the current storage configuration for the 'Root' volume. The table has columns for Volume Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, Throughput (MB/s), Delete on Termination, and Encryption. The 'Root' volume is a 'General Purpose SSD (gp2)' with a size of 8 GiB, 100 IOPS, and 3000 MB/s throughput. The 'Delete on Termination' checkbox is checked, and the encryption status is 'Not Encrypted'. Below the table is a button labeled 'Add New Volume'. A blue information box states: '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.' At the bottom, there are navigation buttons: 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags'.

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

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

The screenshot shows the 'Configure Security Group' step of the AWS EC2 Launch Wizard. The breadcrumb trail at the top indicates the current step is '6. Configure Security Group'. The main heading is 'Step 6: Configure Security Group'. Below the heading, a paragraph explains that a security group is a set of firewall rules that control traffic to and from an instance. It provides instructions on how to assign a security group, either by creating a new one or selecting an existing one. The 'Assign a security group' section shows the 'Create a new security group' option selected. The 'Security group name' field contains 'launch-wizard-2', and the 'Description' field contains 'launch-wizard-2 created 2020-04-03T15:16:07.607+05:30'. Below this is a table for adding rules. The table has columns for Type, Protocol, Port Range, Source, and Description. A rule is added for 'SSH' using 'TCP' protocol on port '22', with the source set to 'Anywhere' (0.0.0.0/0, :::0). The description is 'e.g. SSH for Admin Desktop'. Below the table is a button labeled 'Add Rule'. A yellow warning box states: '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.' At the bottom, there are navigation buttons: 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags'.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name: launch-wizard-2

Description: launch-wizard-2 created 2020-04-03T15:16:07.607+05:30

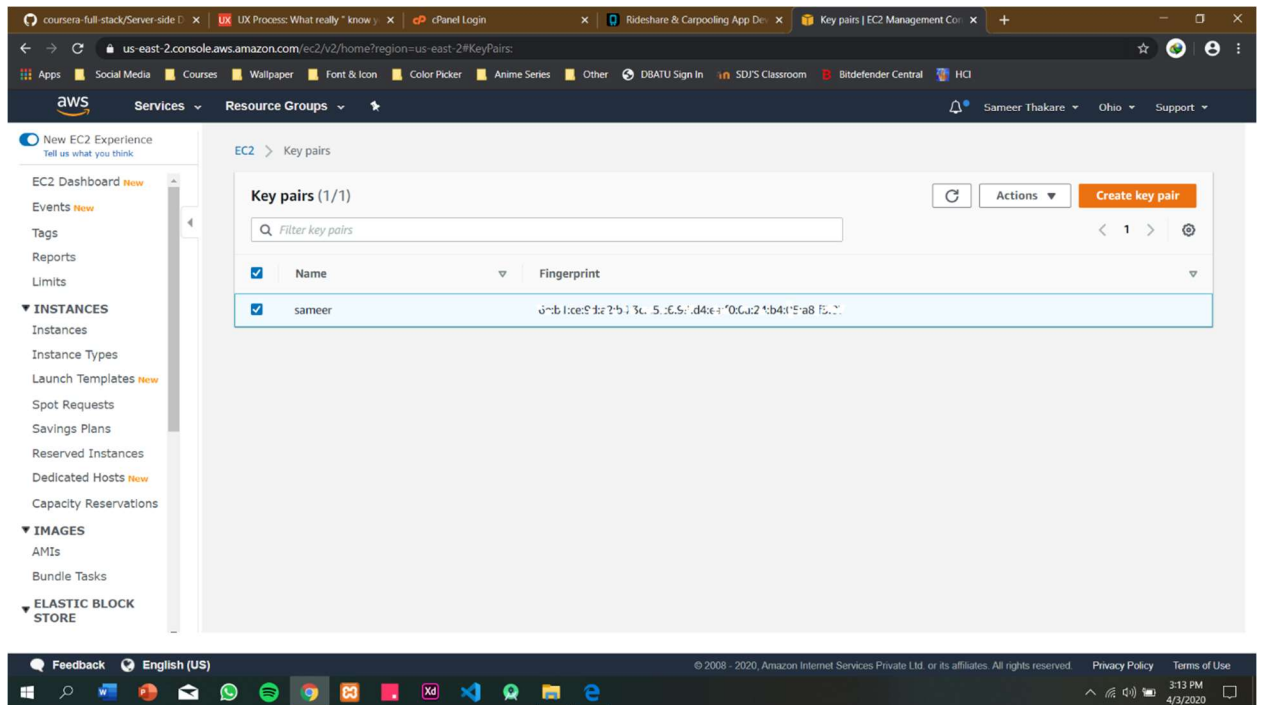
Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere (0.0.0.0/0, :::0)	e.g. SSH for Admin Desktop

Add Rule

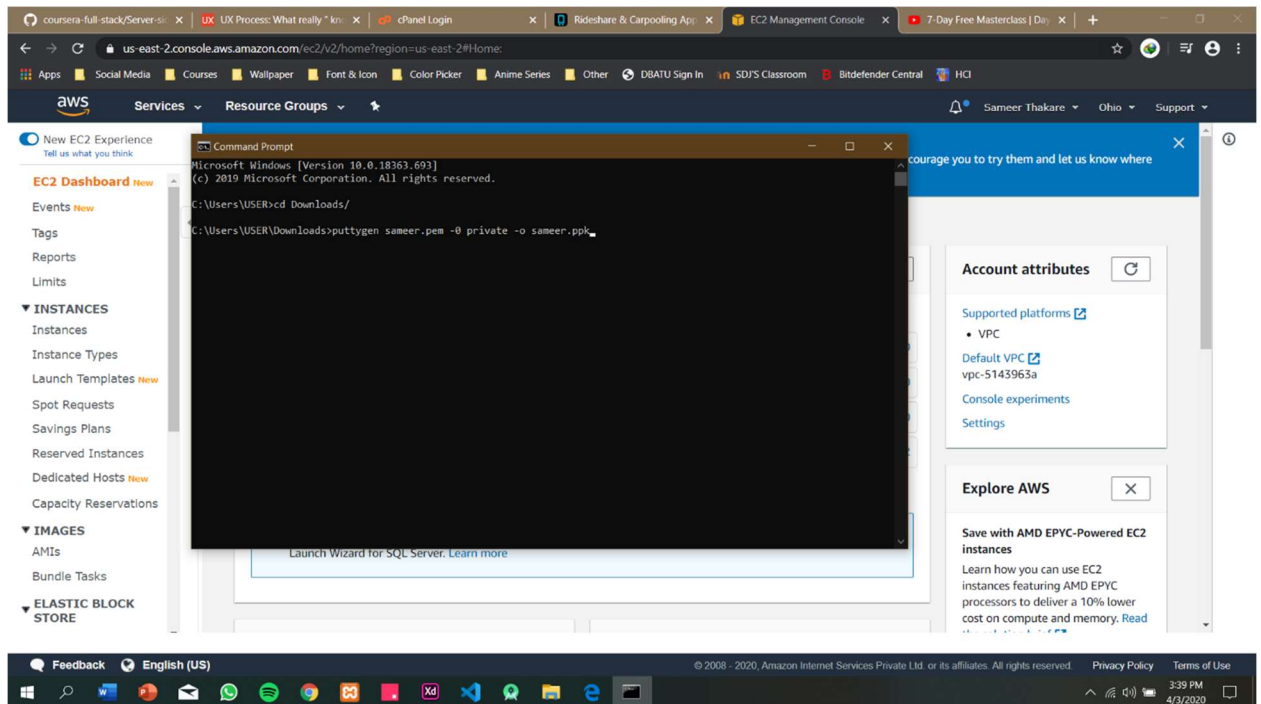
Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous **Review and Launch** Next: Add Tags

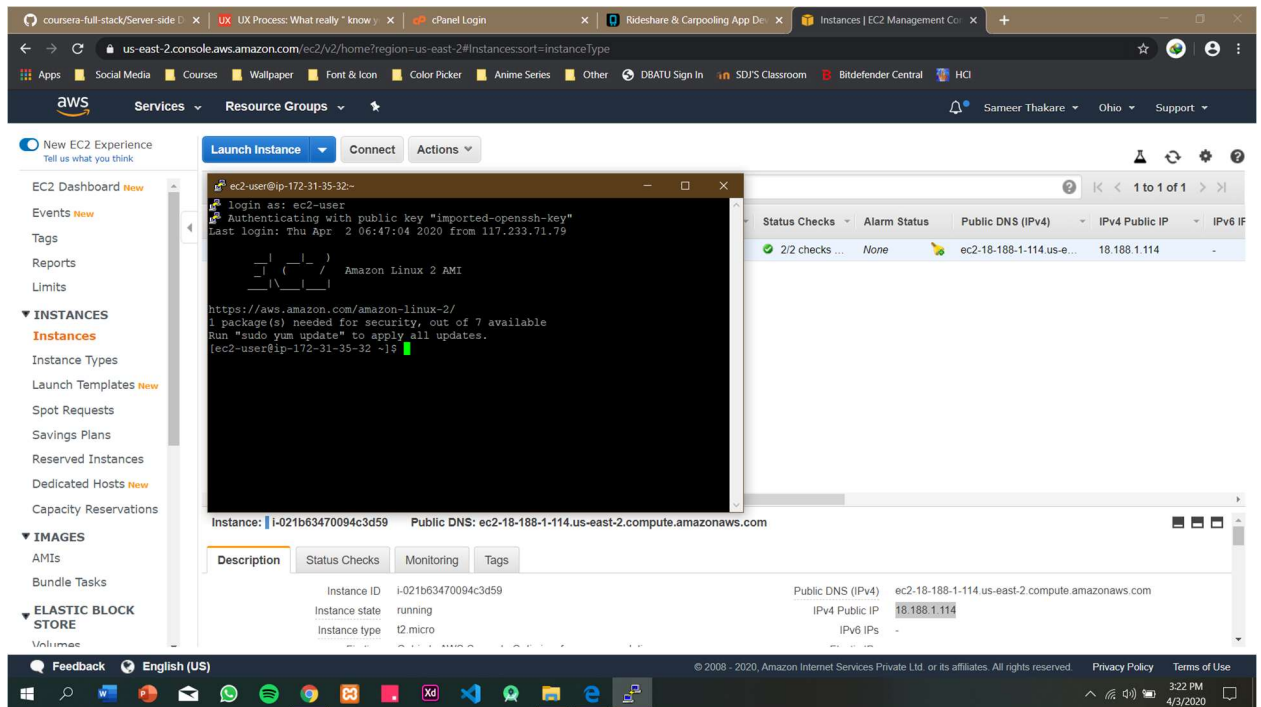
5. Key Pair Download



6. PuTTYgen conversion from pem to ppk

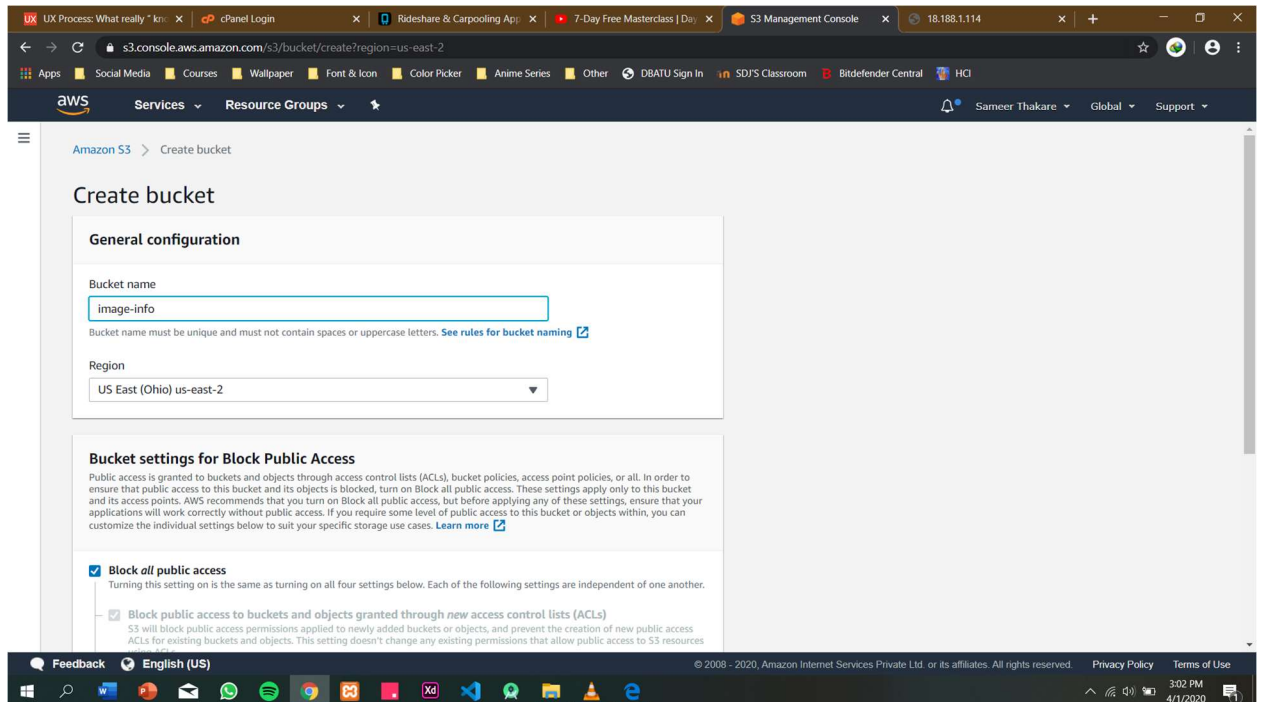


7. Logged in EC2 black screen

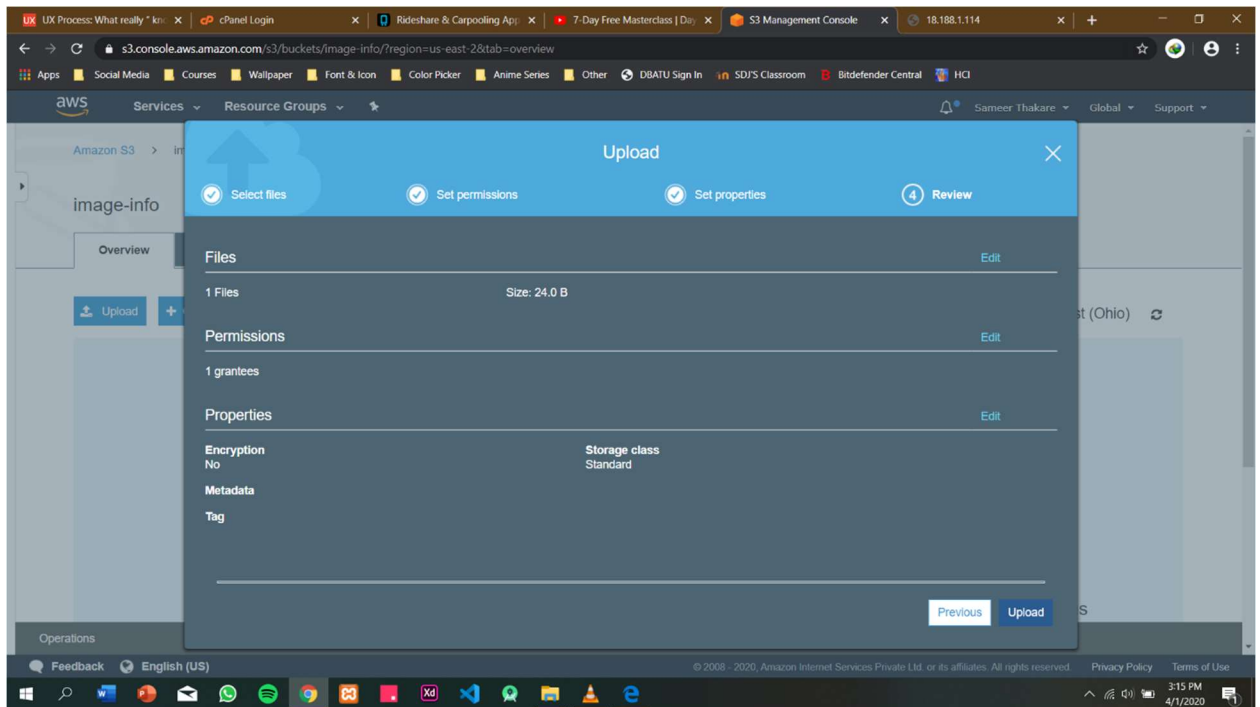


Screenshots needed for S3

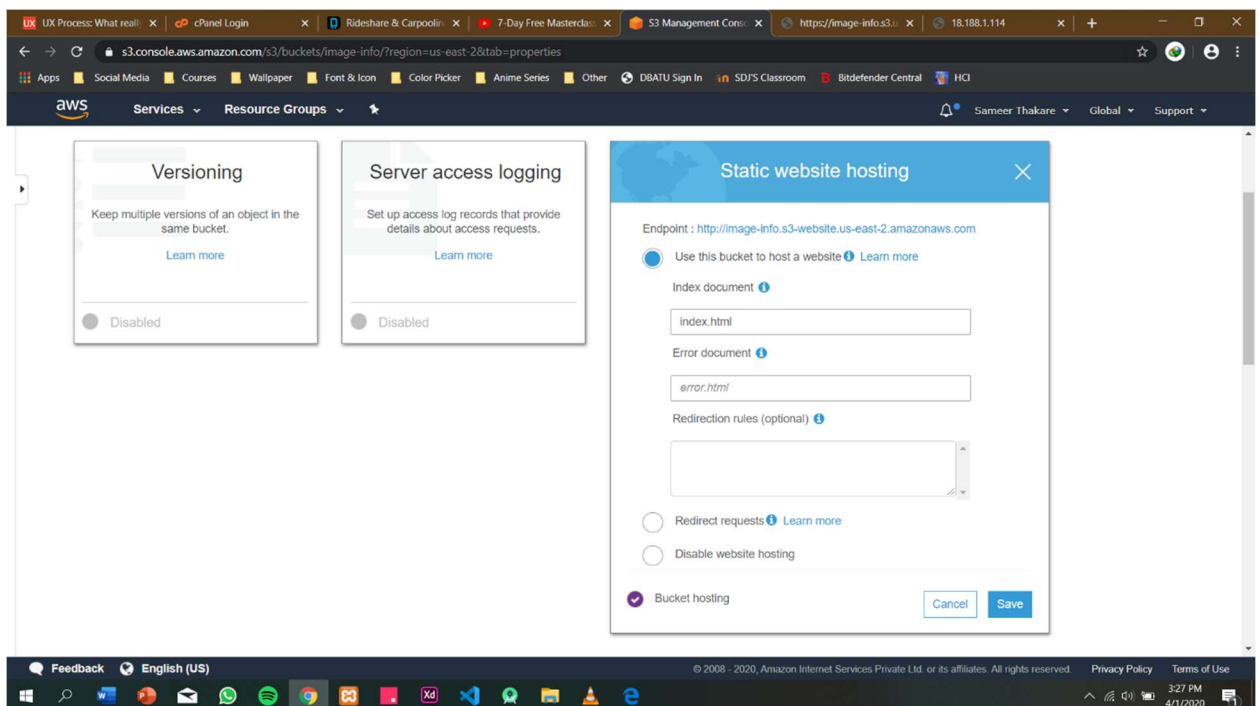
1. Creating a bucket



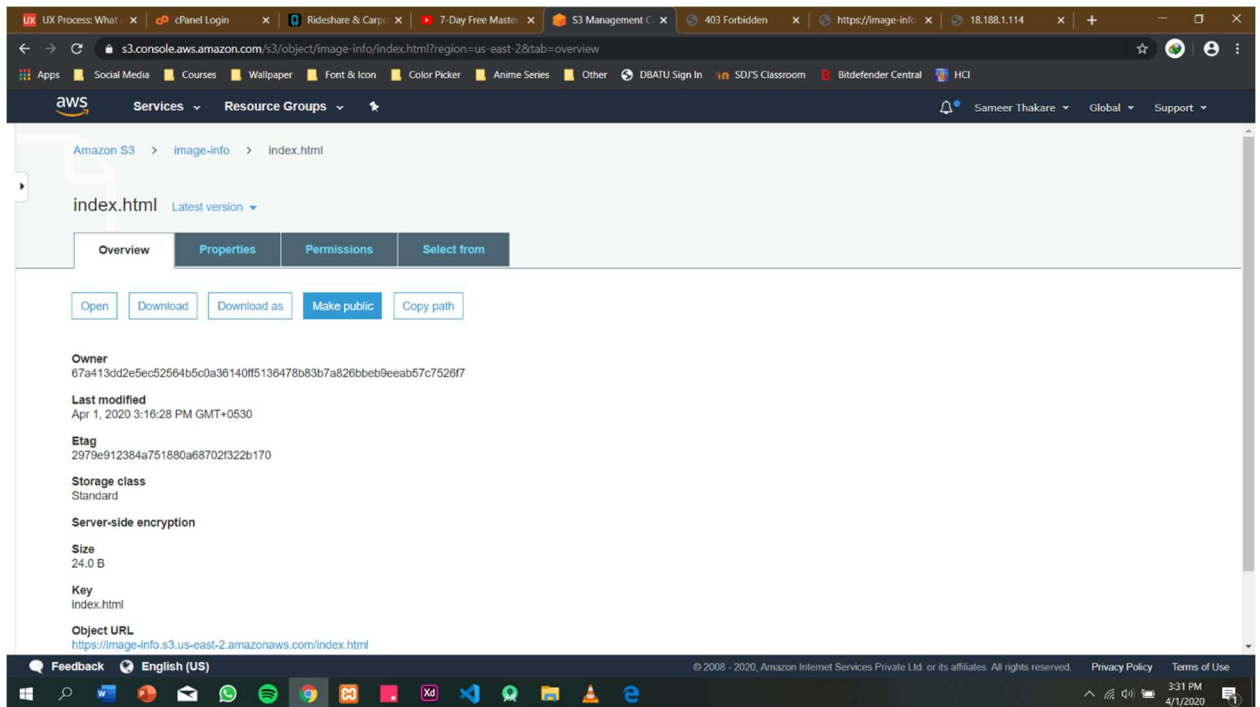
2. Uploading an Object



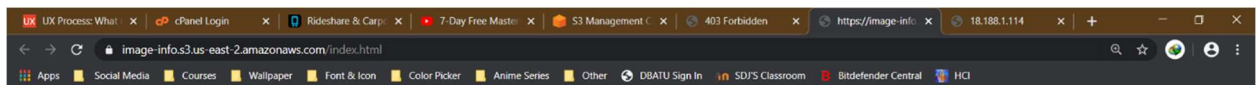
3. Enabling Static Website



4. Making the Object Public



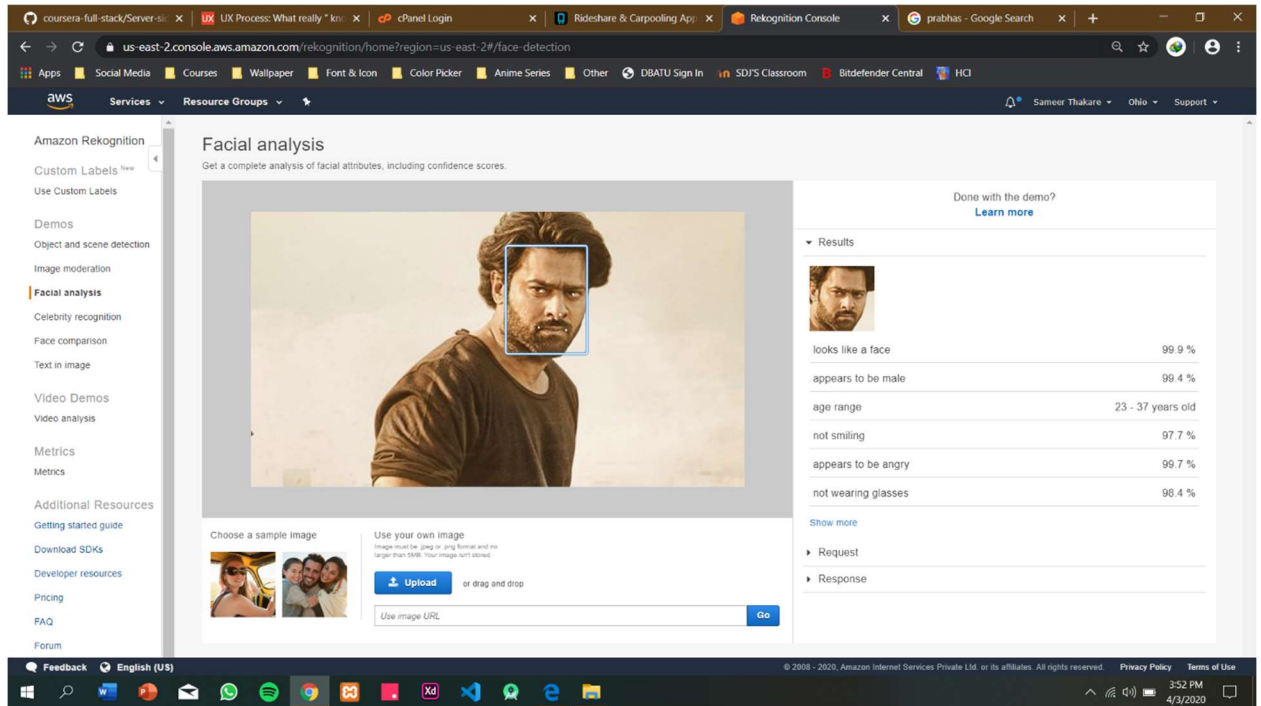
5. Checking the S3 link on the browser



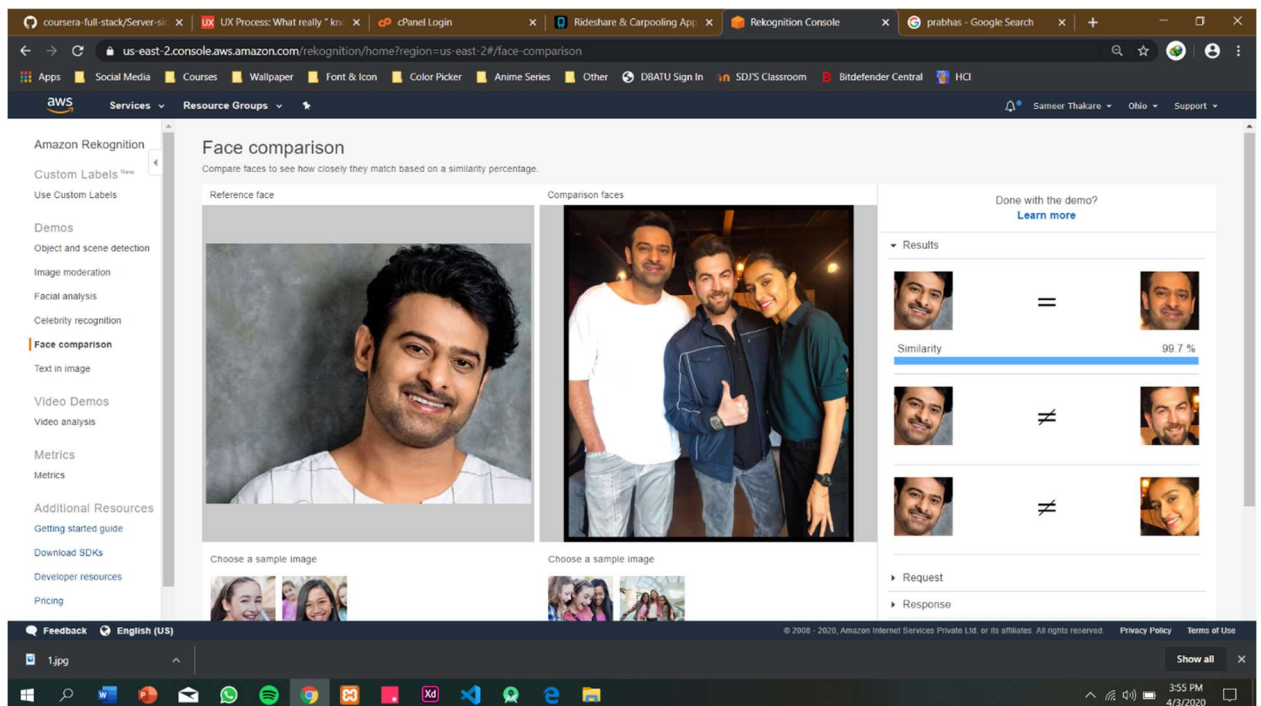
!! Hello, I am Sameer !!

Screenshots needed for Rekognition

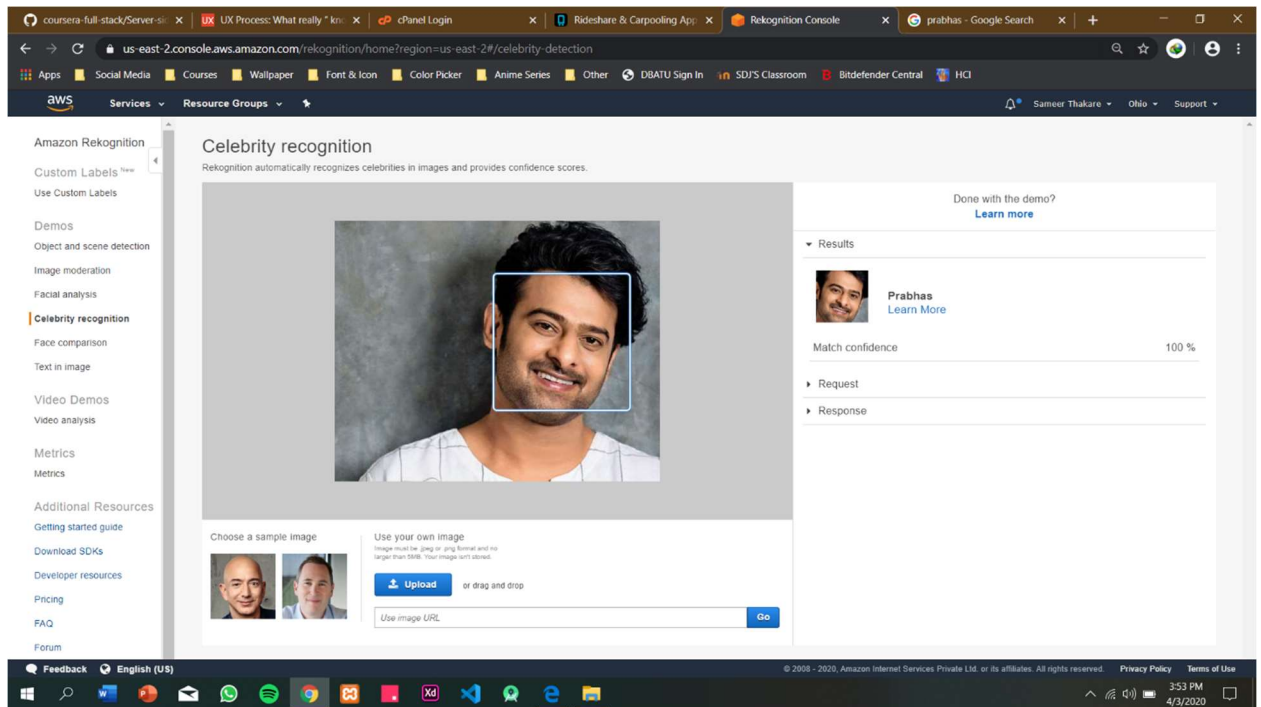
1. Face Detect



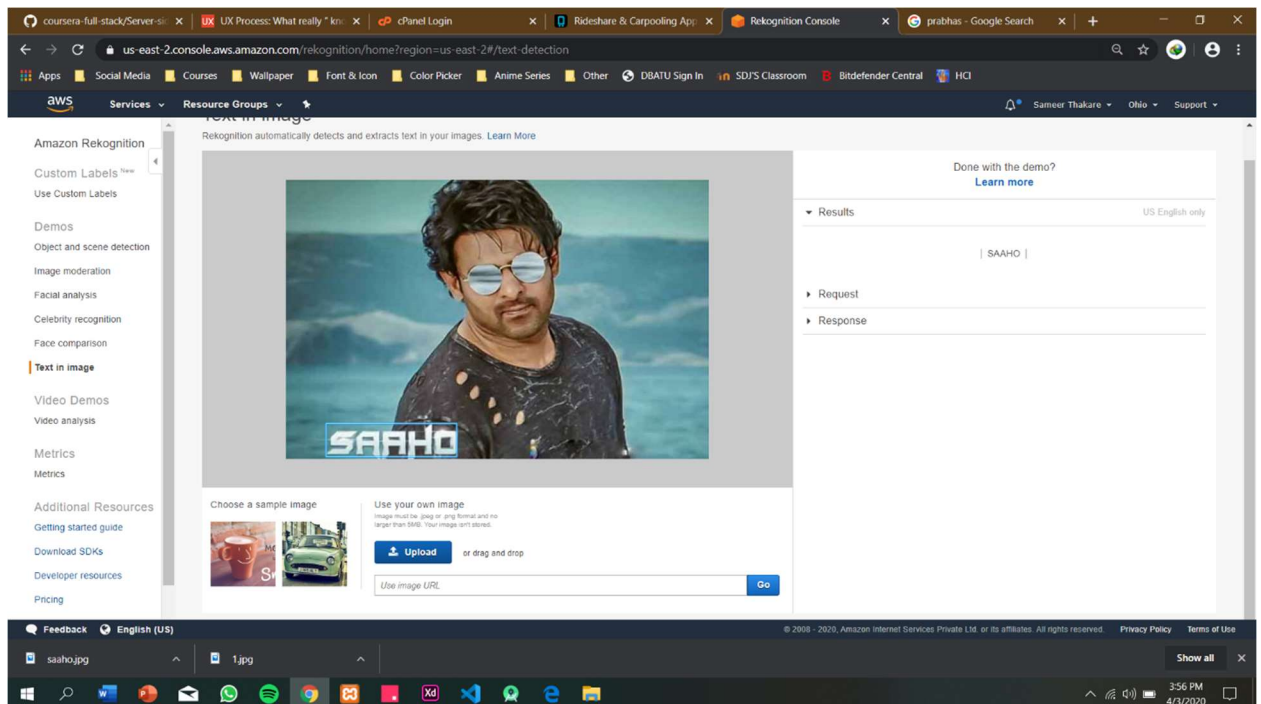
2. Face Compare



3. Celebrity Recognition



4. Text in Image



Screenshots needed for EC2 & S3

1. Installing aws-sdk

```
ec2-user@ip-172-31-35-32-  
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-172-31-35-32 ~]$ 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  
--> Processing Dependency: httpd filesystem for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: libaprutil1.so.0()(64bit) for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.41-1.amzn2.0.1.x86_64  
--> Running transaction check  
--> Package apr.x86_64 0:1.6.3-5.amzn2.0.2 will be installed  
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64  
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed  
--> Package httpd filesystem.noarch 0:2.4.41-1.amzn2.0.1 will be installed  
--> Package httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1 will be installed  
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed  
--> Package mod_http2.x86_64 0:1.15.3-2.amzn2 will be installed  
--> Running transaction check  
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
=====
```

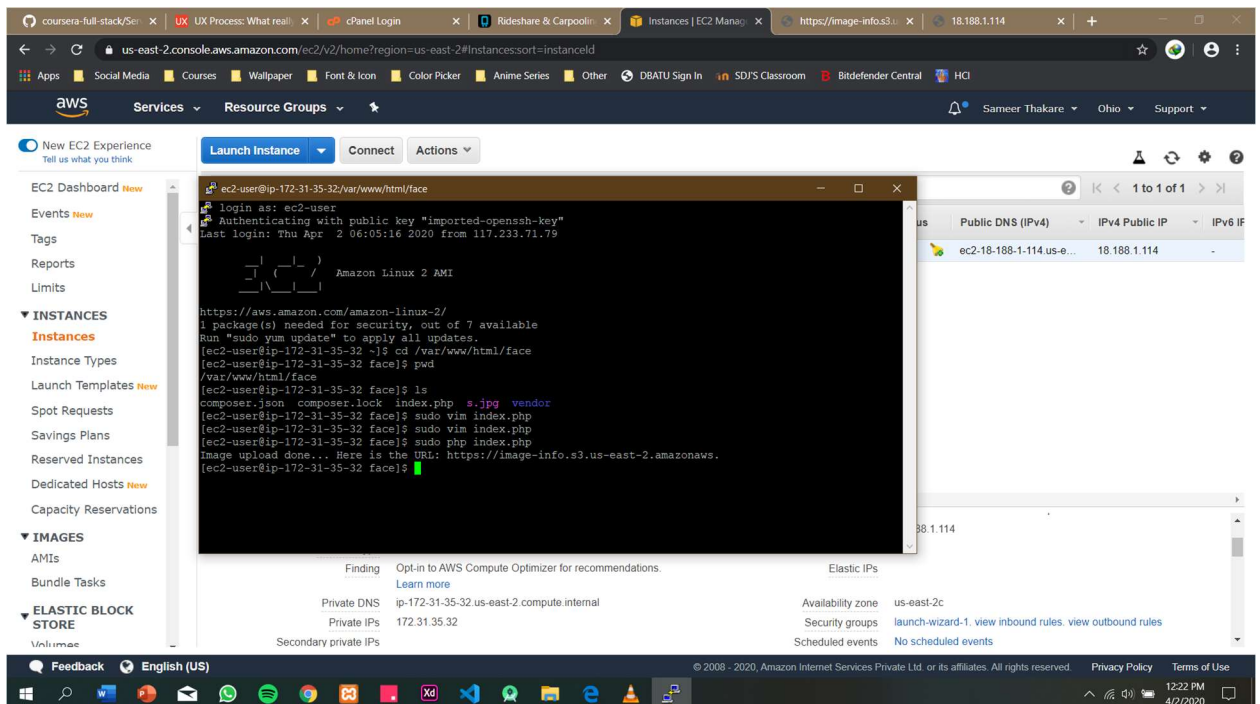
2. Installing php

```
ec2-user@ip-172-31-35-32-  
Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 5/9  
Installing : mailcap-2.1.41-2.amzn2.noarch 6/9  
Installing : httpd filesystem-2.4.41-1.amzn2.0.1.noarch 7/9  
Installing : mod_http2-1.15.3-2.amzn2.x86_64 8/9  
Installing : httpd-2.4.41-1.amzn2.0.1.x86_64 9/9  
Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9  
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9  
Verifying : httpd-2.4.41-1.amzn2.0.1.x86_64 3/9  
Verifying : httpd filesystem-2.4.41-1.amzn2.0.1.noarch 4/9  
Verifying : mod_http2-1.15.3-2.amzn2.x86_64 5/9  
Verifying : apr-1.6.3-5.amzn2.0.2.x86_64 6/9  
Verifying : mailcap-2.1.41-2.amzn2.noarch 7/9  
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 8/9  
Verifying : httpd-tools-2.4.41-1.amzn2.0.1.x86_64 9/9  
  
Installed:  
httpd.x86_64 0:2.4.41-1.amzn2.0.1  
  
Dependency Installed:  
apr.x86_64 0:1.6.3-5.amzn2.0.2  
apr-util.x86_64 0:1.6.1-5.amzn2.0.2  
apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2  
generic-logos-httpd.noarch 0:18.0.0-4.amzn2  
httpd filesystem.noarch 0:2.4.41-1.amzn2.0.1  
httpd-tools.x86_64 0:2.4.41-1.amzn2.0.1  
mailcap.noarch 0:2.1.41-2.amzn2  
mod_http2.x86_64 0:1.15.3-2.amzn2  
  
Complete!  
[ec2-user@ip-172-31-35-32 ~]$ sudo service httpd start  
Redirecting to /bin/systemctl start httpd.service  
[ec2-user@ip-172-31-35-32 ~]$ sudo service httpd status  
Redirecting to /bin/systemctl status httpd.service  
● httpd.service - The Apache HTTP Server  
Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)  
Active: active (running) since Tue 2020-03-31 15:03:01 UTC; 45s ago  
Docs: man:httpd.service(8)  
Main PID: 499 (httpd)  
Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"  
CGroup: /system.slice/httpd.service  
├─499 /usr/sbin/httpd -DFOREGROUND  
├─500 /usr/sbin/httpd -DFOREGROUND  
├─501 /usr/sbin/httpd -DFOREGROUND  
├─502 /usr/sbin/httpd -DFOREGROUND  
├─503 /usr/sbin/httpd -DFOREGROUND  
└─504 /usr/sbin/httpd -DFOREGROUND  
  
Mar 31 15:03:00 ip-172-31-35-32.us-east-2.compute.internal systemd[1]: Starting The Apache HTTP Server...  
Mar 31 15:03:01 ip-172-31-35-32.us-east-2.compute.internal systemd[1]: Started The Apache HTTP Server.  
[ec2-user@ip-172-31-35-32 ~]$
```

3. index.php file code

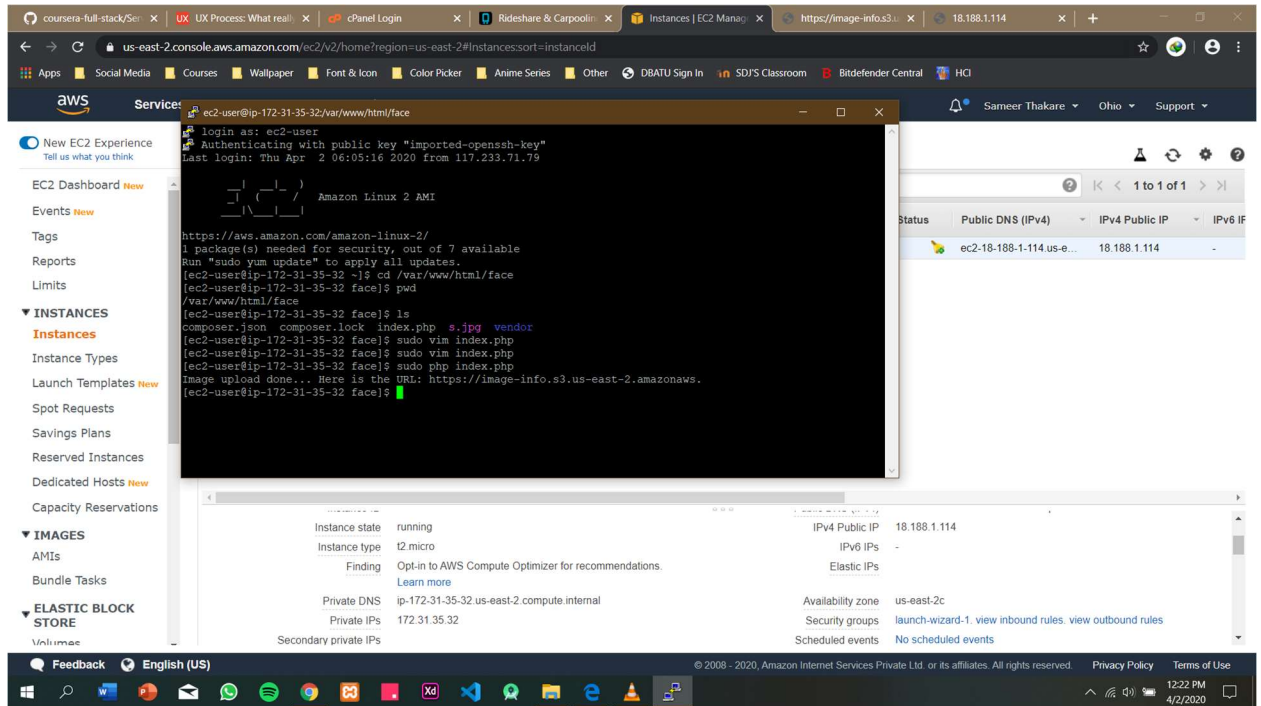
```
index.php
1 <?php
2 /*
3  *
4  *
5  * Install.php - sudo yum install php
6  * curl -s https://getcomposer.org/installer | php
7  * cd /var/www/html
8  * sudo mkdir face
9  * cd face
10 * sudo php -d memory_limit=1 -d composer.phar require aws/aws-sdk-php
11 *
12 * In case if you get memory error ...
13 * sudo /bin/dd if=/dev/zero of=/var/www/html bs=1M count=1024
14 * sudo /bin/mkswap /var/www/html
15 * sudo /bin/swapon /var/www/html
16 *
17 * sudo wget https://s3.amazonaws.com/ec2-images/latest/manifest.json
18 * sudo mv 097e31059417b94004913ac9f508.jpg sample.jpg
19 *
20 *
21 * error_reporting(0);
22 *
23 * require_once($_DIR_ . '/vendor/autoload.php');
24 *
25 * use Aws\S3\S3Client;
26 * use Aws Rekognition\RekognitionClient;
27 *
28 * $bucket = 'image-info';
29 * $keyname = 's.jpg';
30 *
31 *
32 * $s3 = S3Client::factory([
33 *     'profile' => 'default',
34 *     'region' => 'us-east-2',
35 *     'version' => '2006-03-01',
36 *     'signature' => 'v4',
37 *     'key' => 'yourkey',
38 *     'secret' => 'yoursecretkey',
39 * ]);
40 *
41 * try {
42 *     // Upload data
43 *     $result = $s3->putObject([
44 *         'Bucket' => $bucket,
45 *         'Key' => $keyname,
46 *         'SourceFile' => $_FILES_['/sample.jpg'],
47 *         'ACL' => 'public-read'
48 *     ]);
49 * }
50 *
51 * // Print the URL to the object.
52 * $signature = $result['ObjectURL'];
53 * if($signature){
54 *     echo "Image upload done... Here is the URL: " . $signature;
55 * }
56 * } catch (Exception $e) {
57 *     echo $e->getMessage() . PHP_EOL;
58 * }
```

4. Upload success screenshot



Screenshots needed for EC2 & Rekognition

1. Face Detect success screenshot



-----Thank You-----