

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 :

The screenshot shows the AWS Management Console login page. The URL in the address bar is `us-east-2.console.aws.amazon.com/console/home?region=us-east-2`. The page features the AWS logo and navigation links for 'Services' and 'Resource Groups'. On the left, there's a sidebar titled 'AWS services' with sections for 'Find Services' (containing a search bar), 'Recently visited services' (listing S3, EC2, Billing, and IAM), and 'All services' (listing Compute, Blockchain, Security, Identity, & Compliance, and various AWS services like Lambda, Batch, and Amazon Rekognition). On the right, there's a 'Access resources on the go' section with a link to the AWS Mobile App, and an 'Explore AWS' section featuring 'Free Digital Training' (with a link to 350+ courses), 'AWS IQ' (with a link to connect with experts), and 'Amazon SageMaker Studio' (described as 'The first visual integrated development environment').

#### 2. EC2 Dashboard :

The screenshot shows the new EC2 Dashboard. The URL in the address bar is `us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#Home`. A blue banner at the top says 'Welcome to the new EC2 console!' and encourages users to try the new interface. The main dashboard has a sidebar with sections for 'Events', 'Tags', 'Reports', 'Limits', 'INSTANCES' (listing Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, and Capacity Reservations), 'IMAGES' (listing AMIs and Bundle Tasks), and 'ELASTIC BLOCK STORE' (listing Volumes). The central area displays 'Resources' (listing Running instances, Dedicated Hosts, Volumes, Key pairs, Placement groups, Elastic IPs, Snapshots, Load balancers, and Security groups) and a callout for launching Microsoft SQL Server Always On availability groups. On the right, there's an 'Account attributes' section showing supported platforms (VPC), default VPC (vpc-41885d2a), and console experiments, along with an 'Explore AWS' section for optimizing EC2 costs.

### 3. S3 Dashboard :

The screenshot shows the AWS S3 Management console. On the left, there's a sidebar with options like 'Buckets', 'Batch operations', 'Access analyzer for S3', 'Block public access (account settings)', and a 'Feature spotlight'. The main area is titled 'Amazon S3' and shows a table for 'Buckets (1)'. The table has columns for 'Name', 'Region', 'Access', and 'Bucket created'. A single row is listed: 'varzzbuck1' (Region: US East (Ohio) us-east-2), 'Access: Objects can be public', and 'Bucket created: 2020-03-28T07:29:04.000Z'. There are buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'.

### 4. Rekognition Dashboard :

The screenshot shows the Amazon Rekognition console. On the left, there's a sidebar with sections for 'Custom Labels' (with a 'New' badge), 'Demos' (including 'Object and scene detection', 'Image moderation', 'Facial analysis', 'Celebrity recognition', 'Face comparison', 'Text in image'), 'Video Demos' (with 'Video analysis'), 'Metrics' (with 'Metrics'), and 'Additional Resources' (with 'Getting started guide'). The main area is titled 'Amazon Rekognition' with a sub-headline 'Deep learning-based visual analysis service'. It features a 'Try Demo' button and a 'Download SDKs' button. Below this, there are three sections: 'Easily Integrate Powerful Visual Analysis into Your App' (with an icon of three overlapping squares), 'Continuously Learning' (with an icon of a gear and arrows), and 'Integrated with AWS Services' (with an icon of interlocking puzzle pieces). At the bottom, there's a footer with 'Feedback', 'English (US)', and copyright information.

## Screen Shots for EC2 :

### 1. Choosing an AMI :

The screenshot shows the 'Step 1: Choose an Amazon Machine Image (AMI)' page. At the top, there's a search bar with placeholder text 'Q. Search for an AMI by entering a search term e.g. "Windows"'. Below it, a 'Quick Start' sidebar lists 'My AMIs', 'AWS Marketplace', and 'Community AMIs'. The main area displays a list of AMIs:

Image	Name	Description	Root device type	Virtualization type	ENA Enabled	Select
Amazon Linux icon	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0e01ce4ee18447327 (64-bit x86) / ami-03201f374ab66a26e (64-bit Arm)	Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.	Root device type: ebs	Virtualization type: hvm	ENA Enabled: Yes	<input checked="" type="button"/> Select
Amazon Linux icon	Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-01b01bbd08f24c7a8	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	Root device type: ebs	Virtualization type: hvm	ENA Enabled: Yes	<input type="button"/> Select
Red Hat icon	Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)	Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type	Root device type: ebs	Virtualization type: hvm	ENA Enabled: Yes	<input type="button"/> Select
SUSE icon	SUSE Linux Enterprise Server 15 SP1 (HVM), SSD Volume Type - ami-04r5hah51cc146925 (64-bit x86) / ami-02e7390201801R171 (64-bit)	SUSE Linux Enterprise Server 15 SP1 (HVM), SSD Volume Type	Root device type: ebs	Virtualization type: hvm	ENA Enabled: Yes	<input type="button"/> Select

### 2. Choosing an instance type :

The screenshot shows the 'Step 2: Choose an Instance Type' page. At the top, there's a note: '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.' Below it, a table lists instance types:

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	<b>t2.micro</b> <small>Free tier eligible</small>	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
General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
General purpose	t3a.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes

At the bottom, there are buttons: 'Cancel', 'Previous', 'Review and Launch', and 'Next: Configure Instance Details'.

**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	<input type="checkbox"/> Request Spot Instances	
Network	vpc-41885d2a (default)	<input type="button"/> Create new VPC
Subnet	No preference (default subnet in any Availability Zone)	<input type="button"/> Create new subnet
Auto-assign Public IP	Use subnet setting (Enable)	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	Open	<input type="button"/> Create new Capacity Reservation
IAM role	None <input type="button"/> Create new IAM role	
Shutdown behavior	Stop	
Stop - Hibernate behavior	<input type="checkbox"/> Enable hibernation as an additional stop behavior	
Enable termination protection	<input type="checkbox"/> Protect against accidental termination	
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring Additional charges apply.	

**Buttons:** 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"/>	<input 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.

**Buttons:** 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:	<input checked="" type="radio"/> Create a new security group	<input type="radio"/> 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)

**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.

**Buttons:** Cancel Previous Review and Launch

5. Key Pair Download :

Select an existing key pair or create a new key pair X

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**

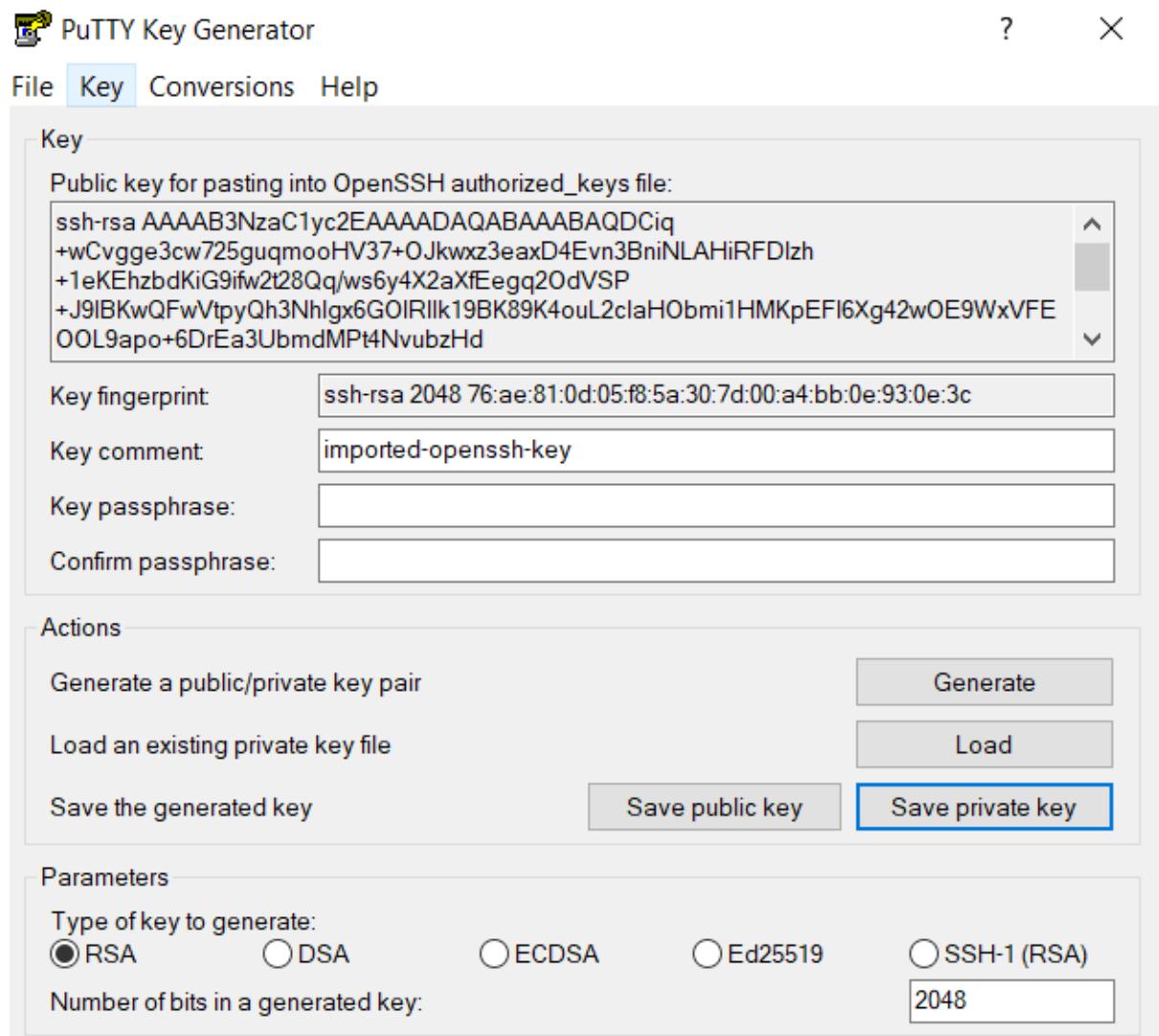
aws-webinar-key ▼

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

The screenshot shows the AWS Launch Status page. At the top, there's a navigation bar with the AWS logo, Services, Resource Groups, and user information (Varush\_H, Ohio, Support). Below the navigation, the title is "Launch Status". A green box contains the message "Your instances are now launching" with a checkmark icon. It also includes the text "The following instance launches have been initiated: i-0b92774169777b3f6" and a link "View launch log". Another box below it says "Get notified of estimated charges" with a help icon, explaining how to receive email notifications for billing alerts. The main content area is titled "How to connect to your instances" and includes a note about instances launching and becoming ready. It lists helpful resources like "How to connect to your Linux instance", "Learn about AWS Free Usage Tier", and links to the "Amazon EC2: User Guide" and "Amazon EC2: Discussion Forum". At the bottom, there's a note about creating status check alarms and attaching EBS volumes, along with links for "Manage security groups" and a "View instances" button.

- ## 6. PuTTYgen conversion from pem to ppk:



- ## 7. Logged in EC2 black screen

```
[ec2-user@ip-172-31-42-137:~]
[ec2-user@ip-172-31-42-137:~] login as: ec2-user
[ec2-user@ip-172-31-42-137:~] 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
4
```

## Screen Shots for S3:

### 1. Creating a bucket :

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with 'Amazon S3' selected under 'Buckets'. A message at the top says, 'We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.' The main area is titled 'Amazon S3' and shows a table for 'Buckets (1)'. The table has columns for Name, Region, Access, and Bucket created. One row is listed: 'varzzbuck1' (Region: US East (Ohio) us-east-2, Access: Objects can be public, Bucket created: 2020-03-28T07:29:04.000Z). At the bottom right of the table, there's a 'Create bucket' button.

### 2. Uploading an object :

The screenshot shows the AWS S3 console interface for the 'varzzbuck1' bucket. The left sidebar shows the bucket name. The main area has tabs for Overview, Properties, Permissions, Management, and Access points, with 'Properties' selected. Below the tabs is a search bar with placeholder text 'Type a prefix and press Enter to search. Press ESC to clear.' Underneath are buttons for Upload, Create folder, Download, and Actions. To the right, it shows the location 'US East (Ohio)' and a note 'Viewing 1 to 3'. A table lists three objects: 'note1.html' (Last modified: Mar 28, 2020 1:04:44 PM GMT+0530, Size: 22.0 B, Storage class: Standard), 'note2.html' (Last modified: Mar 28, 2020 6:34:20 PM GMT+0530, Size: 162.0 B, Storage class: Standard), and 's.jpg' (Last modified: Apr 2, 2020 11:36:39 PM GMT+0530, Size: 210.5 KB, Storage class: Standard). At the bottom, there's a note 'Viewing 1 to 3' and the standard footer with links for Feedback, English (US), Privacy Policy, and Terms of Use.

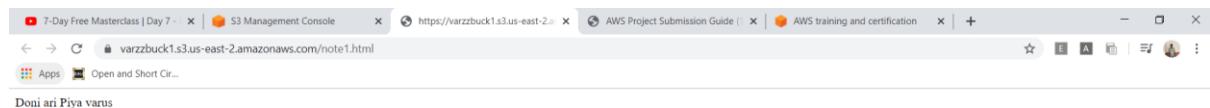
### 3. Enabling Static Website :

The screenshot shows the AWS S3 console for a bucket named 'varzzbuck1'. The 'Properties' tab is selected. Under the 'Static website hosting' section, the status is set to 'Bucket hosting'. Other options like 'Versioning', 'Server access logging', and 'Object-level logging' are also listed but disabled.

### 4. Making the Object Public:

The screenshot shows the 'Permissions' tab of the AWS S3 console for the same bucket. The 'Block public access' section is active. The 'Block all public access' checkbox is checked. Below it, four sub-options are listed, each with a detailed description. The 'Save' button is visible at the bottom right.

## 5. Checking the S3 link on the browser :



## ScreenShots for Rekognition :

### 1. Face Detect :

A screenshot of the AWS Rekognition console in a Microsoft Edge browser. The URL is https://us-east-2.console.aws.amazon.com/rekognition/home?region=us-east-2#/label-detection. The left sidebar shows navigation options like "Custom Labels", "Demos", "Object and scene detection", "Image moderation", "Facial analysis", "Celebrity recognition", "Face comparison", "Text in image", "Video Demos", "Video analysis", "Metrics", and "Additional Resources". The main content area is titled "Object and scene detection" and shows a street scene with a person performing a parkour move. Blue bounding boxes highlight various objects. To the right, there is a "Results" section with a table:

	Done with the demo?
Transportation	98.8 %
Automobile	98.8 %
Vehicle	98.8 %
Car	98.8 %
Person	98.3 %
Human	98.3 %

## 2. Face Compare:

The screenshot shows the AWS Rekognition Face Comparison page. On the left, a sidebar lists various services like Custom Labels, Demos, and Metrics. The main area is titled "Face comparison" and instructs users to "Compare faces to see how closely they match based on a similarity percentage." It features two sections: "Reference face" (a single photo of a girl) and "Comparison faces" (a group of three girls). Below these are two "Choose a sample image" buttons. To the right, a "Results" section displays four pairs of faces with similarity scores: 99.8% (the first pair), 99.8% (the second pair), 99.8% (the third pair), and 99.8% (the fourth pair). A "Done with the demo?" link is at the top right.

## 3. Celebrity Recognition

The screenshot shows the AWS Rekognition Celebrity Recognition page. The sidebar is identical to the Face Comparison page. The main area is titled "Celebrity recognition" and states that "Rekognition automatically recognizes celebrities in images and provides confidence scores." It features a large image of Jeff Bezos with a bounding box around his face. Below this are "Choose a sample image" and "Use your own image" buttons, along with an "Upload" button. To the right, a "Results" section shows a match for "Jeff Bezos" with a "Match confidence" of 100%. Below this are "Request" and "Response" sections. A "Done with the demo?" link is at the top right.

#### 4. Text In Image:

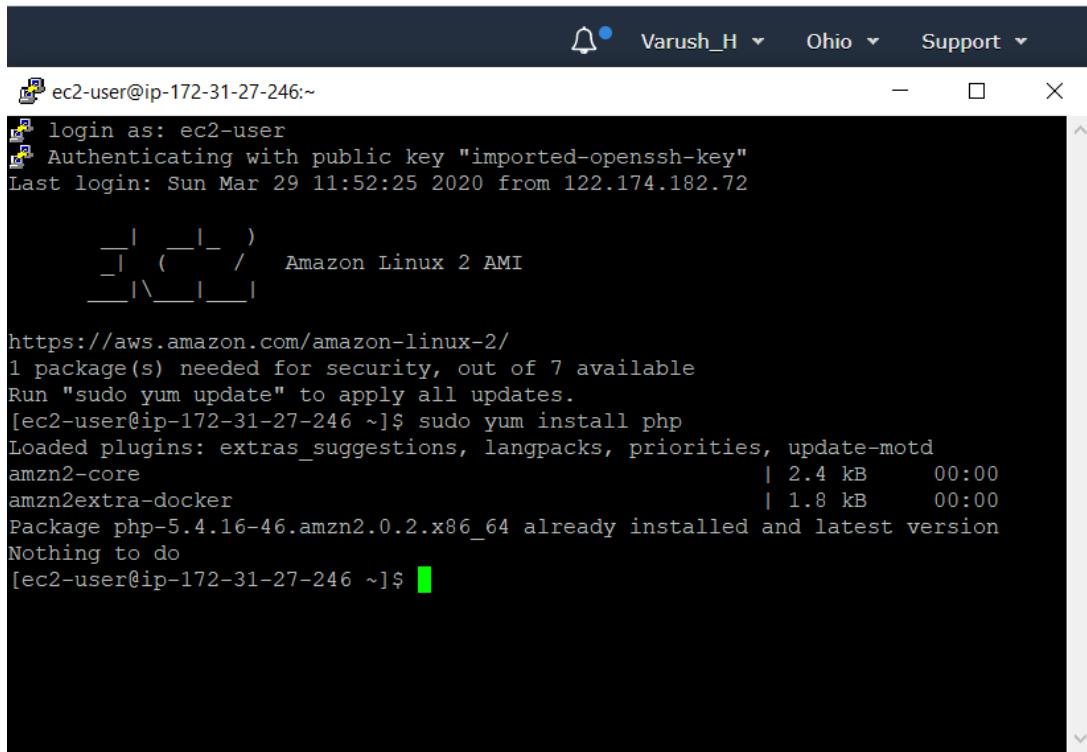
The screenshot shows the AWS Rekognition Text Detection interface. On the left, a sidebar lists various services like Custom Labels, Demos, and Metrics. The main area displays a photo of a coffee cup with a hand-drawn smiley face. Overlaid on the image is text: "IT'S MONDAY but keep Smiling". Below the image, there are sections for "Choose a sample image" and "Use your own image", both with upload buttons. To the right, a results panel shows the detected text: "IT'S", "MONDAY", "but keep", and "Smiling". A "Request" and "Response" section follows. At the bottom, there's a footer with links for Feedback, English (US), and various AWS support options.

#### 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 aws/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 credentials 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 requests and responses)
aws/aws-sdk-php suggests installing symfony/yaml (Eases the ability to write manifests for creating jobs in AWS Import/Export)
Package guzzle/guzzle is abandoned, you should avoid using it. Use guzzlehttp/guzzle instead.
Writing lock file
Generating autoload files
[ec2-user@ip-172-31-27-246 face]$
```

## 2. Installing PHP :



The screenshot shows a terminal window with the following output:

```
ec2-user@ip-172-31-27-246:~$ login as: ec2-user
[ec2-user@ip-172-31-27-246 ~]$ 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 aws/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]$ 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
```

Name	Last modified	Size	Storage class
note1.html	Mar 28, 2020 1:04:44 PM GMT+0530	22.0 B	Standard
note2.html	Mar 28, 2020 6:34:20 PM GMT+0530	162.0 B	Standard
s.jpg	Apr 2, 2020 11:36:39 PM GMT+0530	210.5 KB	Standard