

CLOUD COMPUTING

COURSE CODE:CSE4001

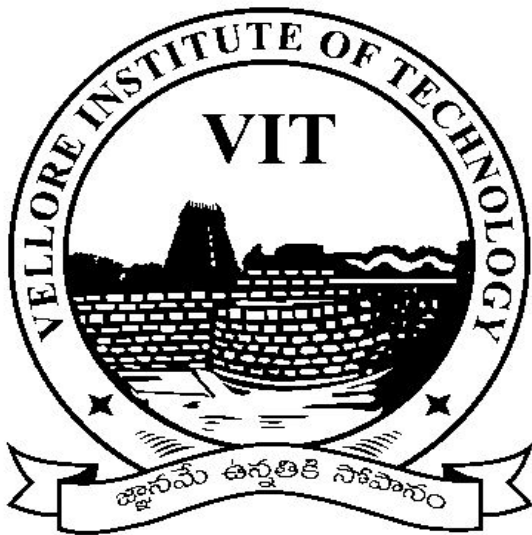
REPORT ON “ WEB APPLICATION HOSTING USING AWS SERVICES ”

Name

Reg.Number

SK Mohammad Younus

17BCE7147



VIT[®]

AP

INDEX

S.NO	TITLE	PAGE.NO
1	Abstract	3
2	Introduction	4
3	Implementation	5-21
	3.1 Steps to launch EC2 Instance	5-10
	3.2 Steps to connect our Instance	10-16
	3.3 .Steps to create PhpMyAdmin,Mysql using Putty:	16-19
	3.4 Steps to upload files to our PhpMyAdmin:	19-21
4	Results	22-27
5	Conclusion	28
6	Reference	28

Abstract:

Developed using PHP and MySQL, this application provides how to host our web application using AWS services. This project Court Case Management System is deployed as a web application using AWS cloud ec2 instance. It has client module, lawyer module and an admin login.

The common features of the Court Case Management includes functions such as client module, lawyer module. It also provides an admin module columns which helps to maintain the cases information of lawyer and client.

PROJECT PROFILE:

Project Name	Court Case Management System
Objective	Deploy Php Mysql project online using AWS services
Platform	AWS EC2 INSTANCE
Front End	PHP,HTML,CSS
BACK End	Mysql
Tools	Putty,Filezilla,Amazon Linux AMI 2018.03.0(Virtual Machine)

Introduction:

The Introduction gives a brief description of the applications addressed in this project and of the particular modules that have been implemented in the project. And also the tools that have been used for the Web deployment of the project.

We can use Amazon Elastic Beanstalk for static php application deployment.

As our project is dynamic, in this project Court Case Management System, we have used an EC2 instance to deploy AWS Cloud. An EC2 instance is nothing but a *virtual server* in Amazon Web services terminology. It stands for *Elastic Compute Cloud*. I have used Amazon Linux AMI 2018.03.0(HVM) virtual machine.

Project Implementation:

1.Steps to Launch AWS EC2 Instance:

i.Login and access to AWS services

- Login to your AWS account and go to the AWS Services tab at the top left corner.
- Here, you will see all of the AWS Services categorized as per their area viz. Compute, Storage, Database, etc. For creating an EC2 instance, we have to use EC2.
- Set Timezone
- Select Launch Instance option

ii.Choose AMI

1. You will be asked to choose an AMI of your choice. (An AMI is an Amazon Machine Image. It is a template basically of an Operating System platform which you can use as a base to create your instance). Once you launch an EC2 instance from your preferred AMI, the instance will automatically be booted with the desired OS. (We will see more about AMIs in the coming part of the tutorial).
2. Here we are choosing the Amazon Linux AMI 2018.03.0(64 bit).



Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-005956c5f0f757d37

Amazon Linux
Free tier eligible

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.

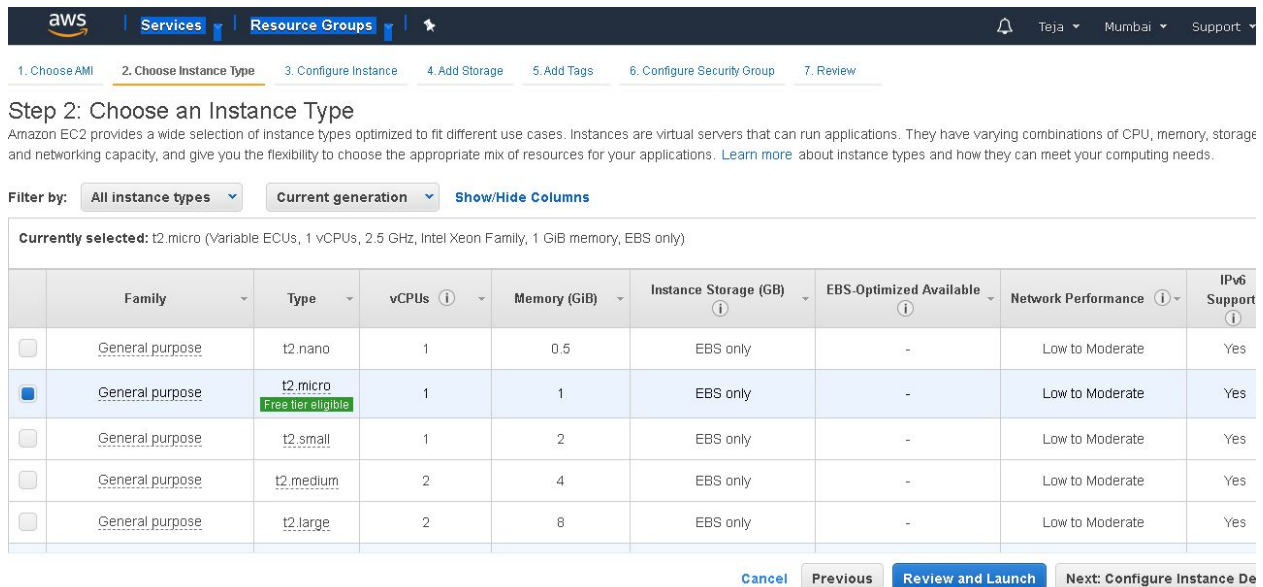
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

iii. Configure Instance

1. Choose Free tier Instance it doesn't cost more.



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 Free tier eligible	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
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes

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

iv. Add Storage

1. Free tier provides 30GB storage free. I choosen 20GB 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-0f5c831fcfd9c5b8e	20	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](#)

v.Tag Instance

1. You can tag your instance with a key-value pair. This gives visibility to the AWS account administrator when there are a lot of instances.
2. The instances should be tagged based on their department, environment like Dev/SIT/Prod. Etc. This gives a clear view of the costing on the instances under one common tag.
3. Here we have tagged the instance as a PHP site.

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances ⓘ	Volumes ⓘ
Name	PHP SITE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

vi. Configure Security Groups

In this next step of configuring Security Groups, you can restrict traffic on your instance ports. This is an added firewall mechanism provided by AWS apart from your instance's OS firewall.

You can define open ports and IPs.

1. Since our server is a web server, I will do following things
2. Creating a new Security Group
3. Create HTTP, HTTPS Protocols and make their source as “Anywhere” to access them for anywhere.
4. Once, the firewall rules are set– Review and launch

aws Services Resource Groups

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

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
HTTP	TCP	80	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop

Add Rule

Warning

Cancel Previous Review and Launch

vii. Review And Launch Instance:

1. We can see our instance in running instances.

aws Services Resource Groups

New EC2 Experience Tell us what you think

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
PHP SITE	i-0d1ee45844aaef806	t2.micro	ap-south-1a	running	2/2 checks ...	None	ec2-13-127-252-157.ap-south-1a...

Instance: i-0d1ee45844aaef806 (PHP SITE) Public DNS: ec2-13-127-252-157.ap-south-1.compute.amazonaws.com

Description	Status Checks	Monitoring	Tags
Instance ID	i-0d1ee45844aaef806	Public DNS (IPv4)	ec2-13-127-252-157.ap-south-1.compute.amazonaws.com
Instance state	running	IPv4 Public IP	13.127.252.157
Instance type	t2.micro	IPv6 IPs	-
Finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more	Elastic IPs	-
Private DNS	ip-172-31-40-14.ap-south-1.compute.internal	Availability zone	ap-south-1a

2.Steps to connect our Instance:

i.Download Keypair:

After Launching Instance you will be asked to create a key pair to login to your instance. A key pair is a set of public-private keys.

AWS stores the private key in the instance, and you are asked to download the private key. Make sure you download the key and keep it safe and secured; if it is lost you cannot download it again.

1. Create a new key pair
2. Give a name to your key
3. Download and save it in your secured folder
4. I gave name as “mykeypair”.
5. If we open the mykeypair.pem it is as given below.

```

mykeypair.pem
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAsK0UU7yi5ziSHnqr4WjXYEHm3/ieum20j9fAD1UziQvi6sq8dozwIbWy8IN
KxjV1nLD+m5Bvf/P91ke1BopybSTGhr67peE/FI+sxMQBA7bpSaCTfNi4cQ+qVHcO5hpEQBUhBgh
4CCRJM1oM+vaMTet9/YbRW8SBf3CjBMnYQNY7QM7zKqKwvyRqC8e76DxFTeqvZg4tdHYE1AQKcs
N2UujBHZxbtGZ4YSIXgB/SGZpbiYcmeKga2gtwHJWR9sdi2ZkOJP70A9QMqE2evEIYTiHNbpsE8x
xlzJ0yevVKy1HxI+nhaOkeiR5HHI/7QGwDJCUFo1UhD27IOQtIPmYQIDAQABaoIBAQCbGfckZBAN
KPSuKUdHhcC+Ulya9zvIx/Clttthx2+L1bcMHvGbhF4wjUryy+FKTKSRhNTNzYOP6Jt+5+E21xlk
aFBTtKfiWcZ3tMQydW00DE8rNx1h0W9Xf+76u1C7SNvD/yQzKvRiTHTfeCwomL9eQcp2/umGFTIf
bF0k15QUh3rTv9e1vYkAGUfRwMSAPcaE0KcpXtfxI2XzEY1t6f7C2mFM0ppfpgP59ZLLGs47m+M
Kg0L9rCFermc1FbI9IcOofyYs/mj6N0g9ZDrJupPXAXGfc0kNhGuRjEtJZPjDWMkkoZOajG1cZwV
2n5IC2hBec8RZHJA742JpLPpPyihAoGBAN4oPuEQFZWnUOb84YMpmoPj+8GFwBLVewAm5vj3PAb6
nBZXsleHJz5gbyhBvOAYQ71ejsgie7xL/xtze+NbzEASvDvirM3Uvd5bCMt1/NP02XXmMEw0XoLT
xj27ECvLa1PASepe5hG1CZuzvpj+Dk+YVfb4WjnOQ/zSQ0BtZ75NAoGBAMuXJbZ88qcS2a8omYQ9
Z7U818tPTt7GPHL15GRXo86p/SkcMM7e+Ixyq5mI7C+MUDxxceE6vYBcF/Fzsayp66QxEChxhOeC
V2/X1heuj90/5QV9Mh5ujPFi++2WJydf2s865pK7Zo7t9eNINXFeKeIiZW1DAABr6pmT5pF7Shp1
AoGAdYxMASBwM3IFr8M8u/8mgAUA8BvTR1voEmODMVB4TyUJtBTBC+k6jZnPFzj5bIXKR+pNmH40
hMqdeHP5qIqpH3oKWZG0J/caY+49UBL1LW3QJr8e7SOEogrVqVIys0mVwCztr3J5QANPyzAneKJt
Bhav3db5Pr7yYHMHKkHDL0CgYAcOo3pGrQ0hGeYiNmYY2dA0vD9KLb33qa9C7E6VIiK/Dj0/Uqr
NHfKDt7CICExiq4JY3V11E44Q0y01uTYjGyt7HvY12M1MoZrJIpUKEkbBC1+MFjFrXBve9kQnVQ6
emj9eN5F1CRpRwdkBzwZxwOprj1ACB008Cgqn75Fh8HdnQKBgQDdkPruc/fToznrjvG7FX3ARJaZ
VICsMKByIvCoZcd2oLMHs4c92Em9Hsuy21vbK1FERirL4ePc0n6CDNm+oFqUvF95cwWlLwboAv3j
XWGQaKdXLP8cHRvX+IJmg2EJEeOIBUW28d9EHdc/Ykma1cBij5ma76Sjd/a+wLxIKt9RPA==
-----END RSA PRIVATE KEY-----

```

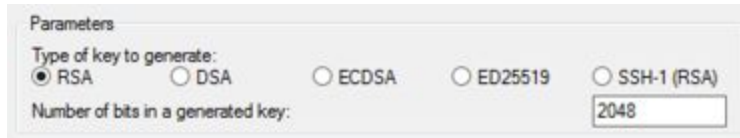
ii. Connect Instance Using Putty:

PuTTY does not natively support the private key format for SSH keys. PuTTY provides a tool named PuTTYgen, which converts keys to the required format for PuTTY. You must convert your private key (.pem file) into this format (.ppk file) as follows in order to connect to your instance using PuTTY.

To convert your private key

1. From the Start menu, choose All Programs, PuTTY, PuTTYgen.

2. Under Type of key to generate, choose RSA. If you're using an older version of PuTTYgen, choose SSH-2 RSA.



3. Choose Load. By default, PuTTYgen displays only files with the extension .ppk. To locate your .pem file, choose the option to display files of all types.



4. Select your .pem file for the key pair that you specified when you launched your instance and choose Open. PuTTYgen displays a notice that the .pem file was successfully imported. Choose OK.
5. To save the key in the format that PuTTY can use, choose Save private key. PuTTYgen displays a warning about saving the key without a passphrase. Choose Yes. A passphrase on a private key is an extra layer of protection. Even if your private key is discovered, it can't be used without the passphrase. The downside to using a passphrase is that it makes automation harder because human intervention is needed to log on to an instance, or to copy files to an instance.

6. Specify the same name for the key that you used for the key pair (for example, `my-key-pair`) and choose Save. PuTTY automatically adds the `.ppk` file extension.

Your private key is now in the correct format for use with PuTTY. You can now connect to your instance using PuTTY's SSH client.

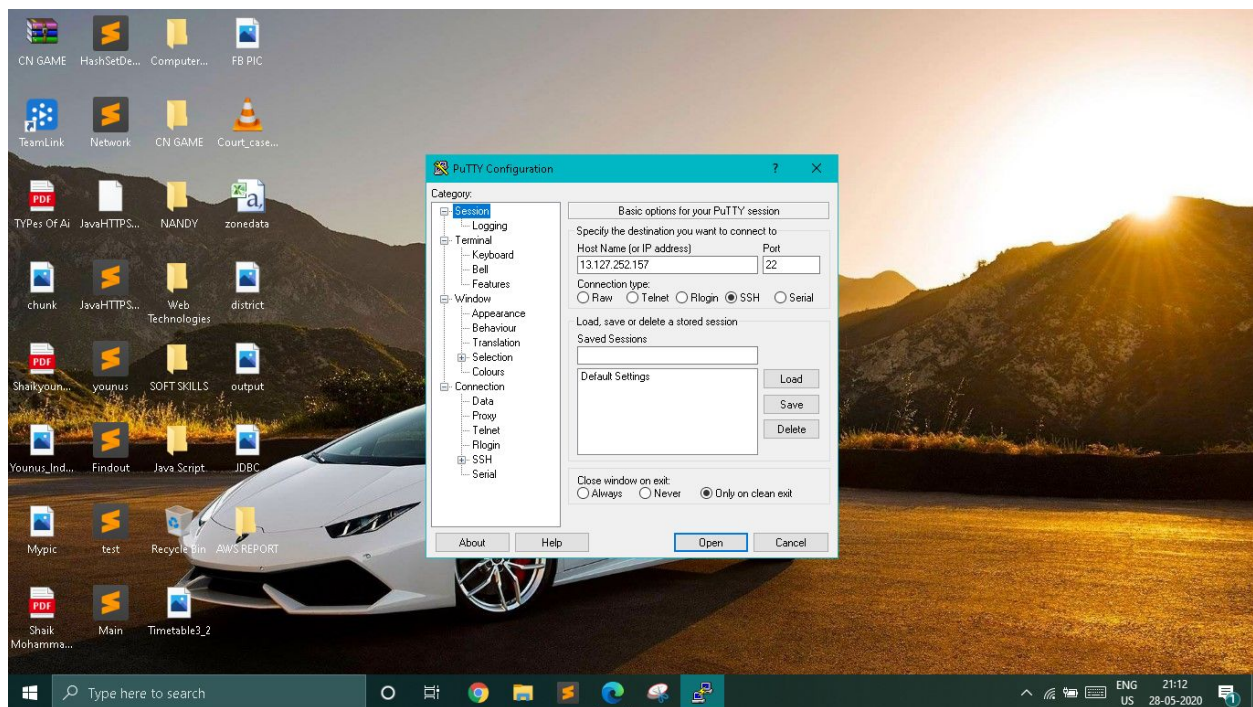
iii.Connecting to your Linux instance

Use the following procedure to connect to your Linux instance using PuTTY. You need the `.ppk` file that you created for your private key. For more information, see Convert your private key using PuTTYgen in the preceding section. If you receive an error while attempting to connect to your instance, see Troubleshooting Connecting to Your Instance.

To connect to your instance using PuTTY

1. Start PuTTY (from the Start menu, choose All Programs, PuTTY, PuTTY).
2. In the Category pane, choose Session and complete the following fields:
 - a. In the Host Name box, do one of the following:

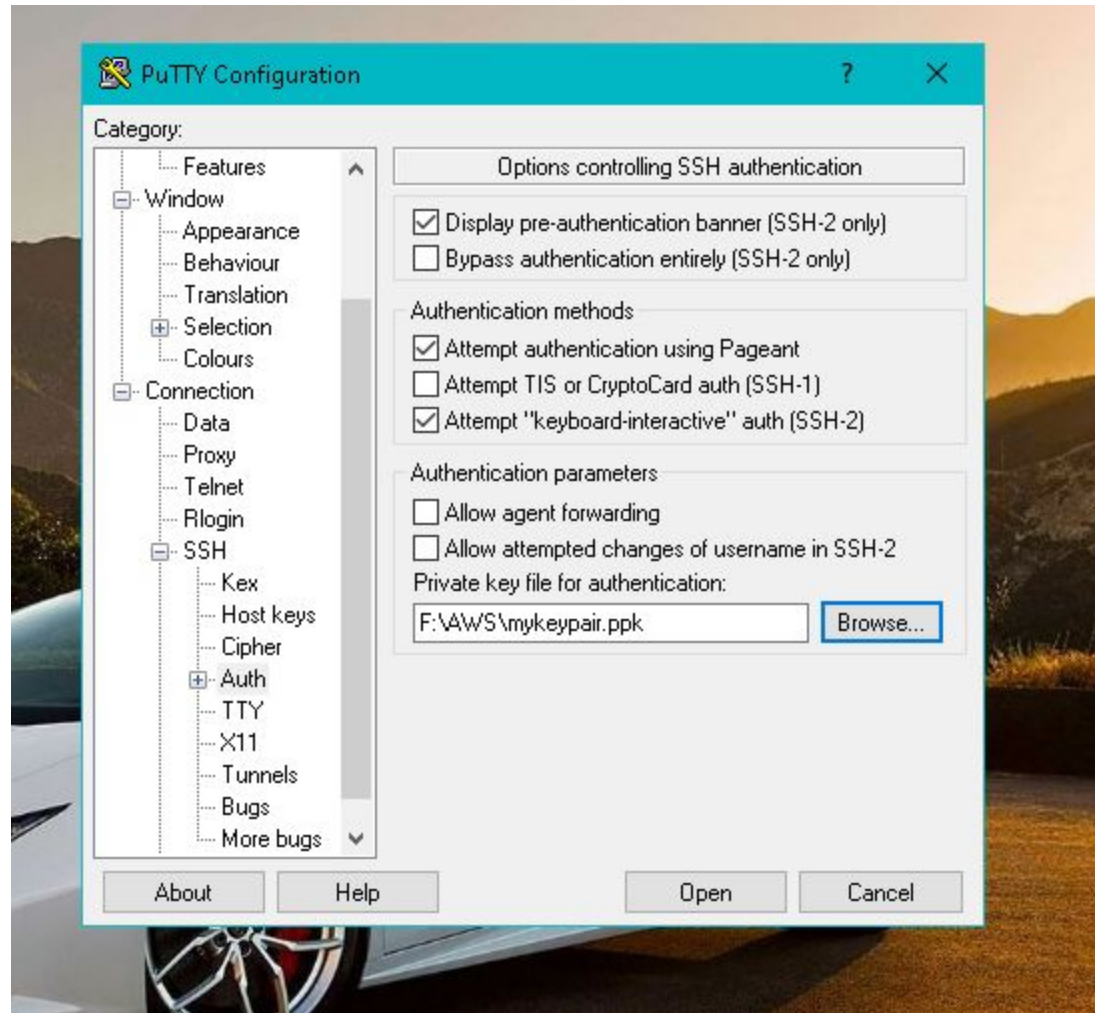
- (Public DNS) To connect using your instance's public DNS name, enter “13.127.252.157”(IP Address)
- b. Ensure that the Port value is 22
- c. Under Connection type, select SSH.



3. In the Category pane, expand Connection, expand SSH, and then choose Auth. Complete the following:
 - a. Choose Browse.
 - b. Select the .ppk file that you generated for your key pair and choose Open.
 - c. (Optional) If you plan to start this session again later, you can save the session information for future use.

Under Category, choose Session, enter a name for the session in Saved Sessions, and then choose Save.

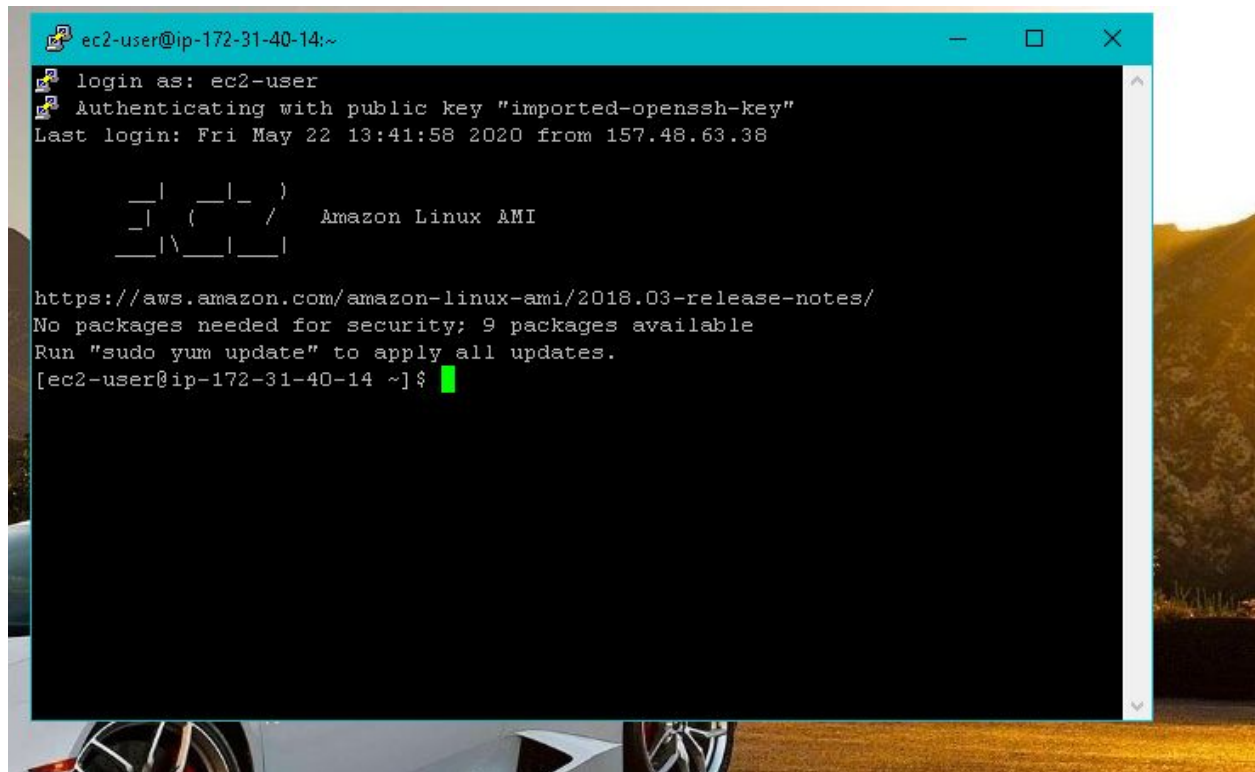
d. Choose Open.



e.

4. If this is the first time you have connected to this instance, PuTTY displays a security alert dialog box that asks whether you trust the host to which you are connecting.

5. And the login as :ec2-user



3.Steps to create PhpMyAdmin,Mysql using Putty:

i.Install PHP in PhpMyAdmin Folder:

Use below commands

1. [ec2-user ~]\$ sudo yum update
2. [ec2-user ~]\$ sudo yum --enablerepo=epel install phpmyadmin
3. [ec2-user ~]\$ cd /var/www/html
4. [ec2-user html]\$ wget
<https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz>

5. [ec2-user html]\$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1

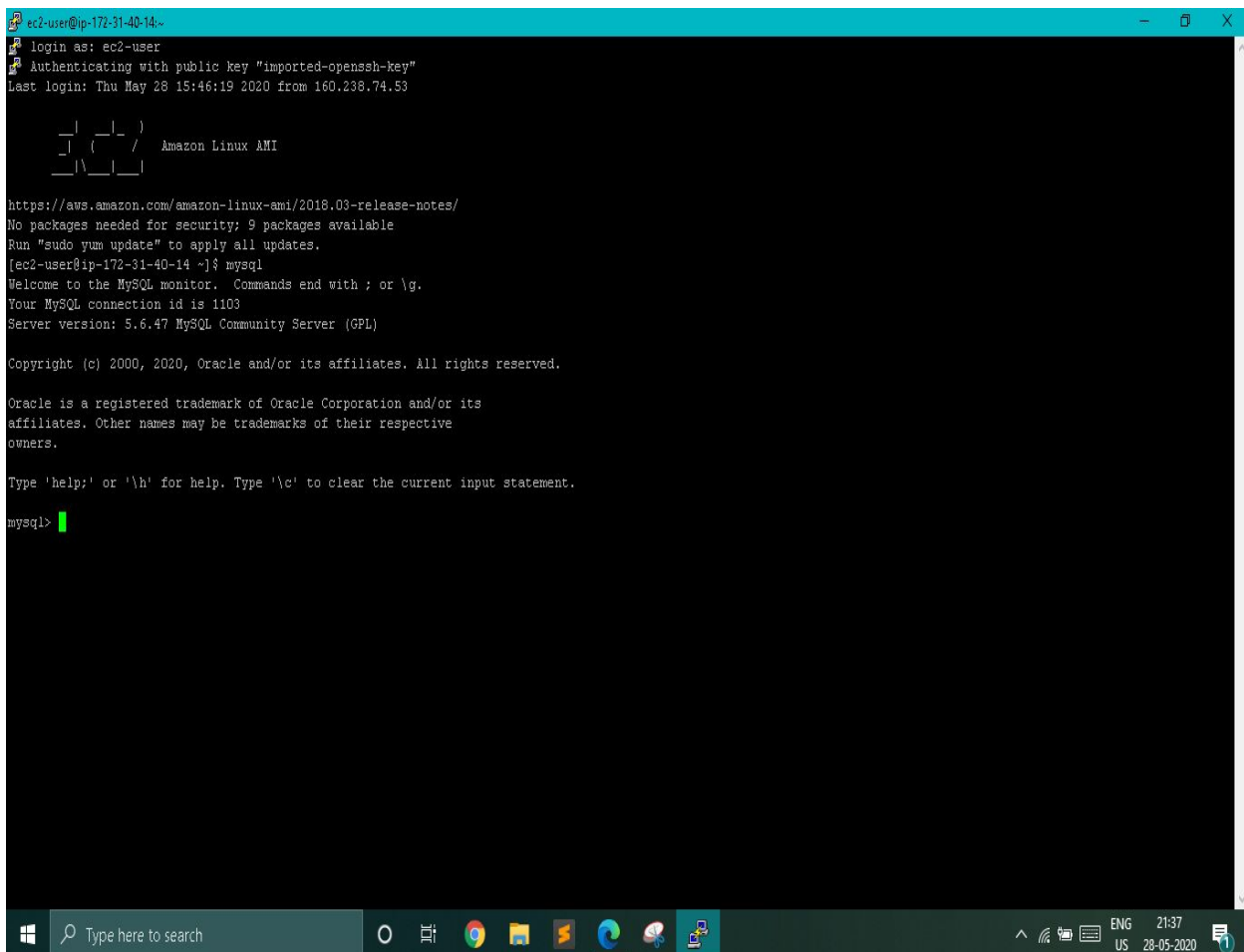
i.Install Mysql:

To install a MySQL Server

1. First step is to of course ssh into the EC2 instance

Then, at a command prompt, use the following command to install MySQL Server:

`sudo yum install mysql-server`



```
ec2-user@ip-172-31-40-14:~$  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
Last login: Thu May 28 15:46:19 2020 from 160.238.74.53  
  
 _ _ _ _ _  
 _ _ _ _ _ / Amazon Linux AMI  
 _ _ _ _ _  
  
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/  
No packages needed for security; 9 packages available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-40-14 ~]$ sudo yum install mysql-server  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 1103  
Server version: 5.6.47 MySQL Community Server (GPL)  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql>
```

To start the installed MySQL Server

Start mysql, and configure it to start up automatically on reboot.

`sudo chkconfig mysqld on`

`sudo service mysqld start`

1. You would see a response like the following.
Starting mysqld

```
[ec2-user@ip-172-31-40-14 ~]$ sudo service mysqld start
Starting mysqld: [ OK ]
[ec2-user@ip-172-31-40-14 ~]$
```

Configuring newly installed MySQL Server

We need to access MySQL from another server, then we need to execute these following additional steps.

First off, create a MySQL user who can connect from any type of host using the following SQL:

```
GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost'
```

```
GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost'
```

```
CREATE USER 'admin'@'localhost' IDENTIFIED BY '[123456]'
```

1. `GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost'`

2. Flush PRIVILEGES.

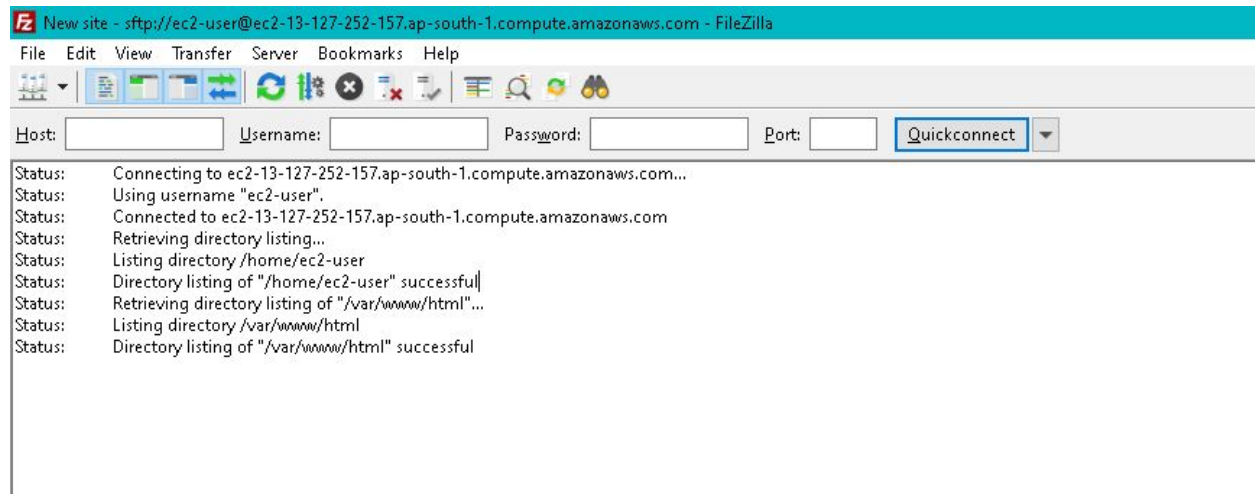
Now we can check if its working or not by

<http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/phpMyAdmin>

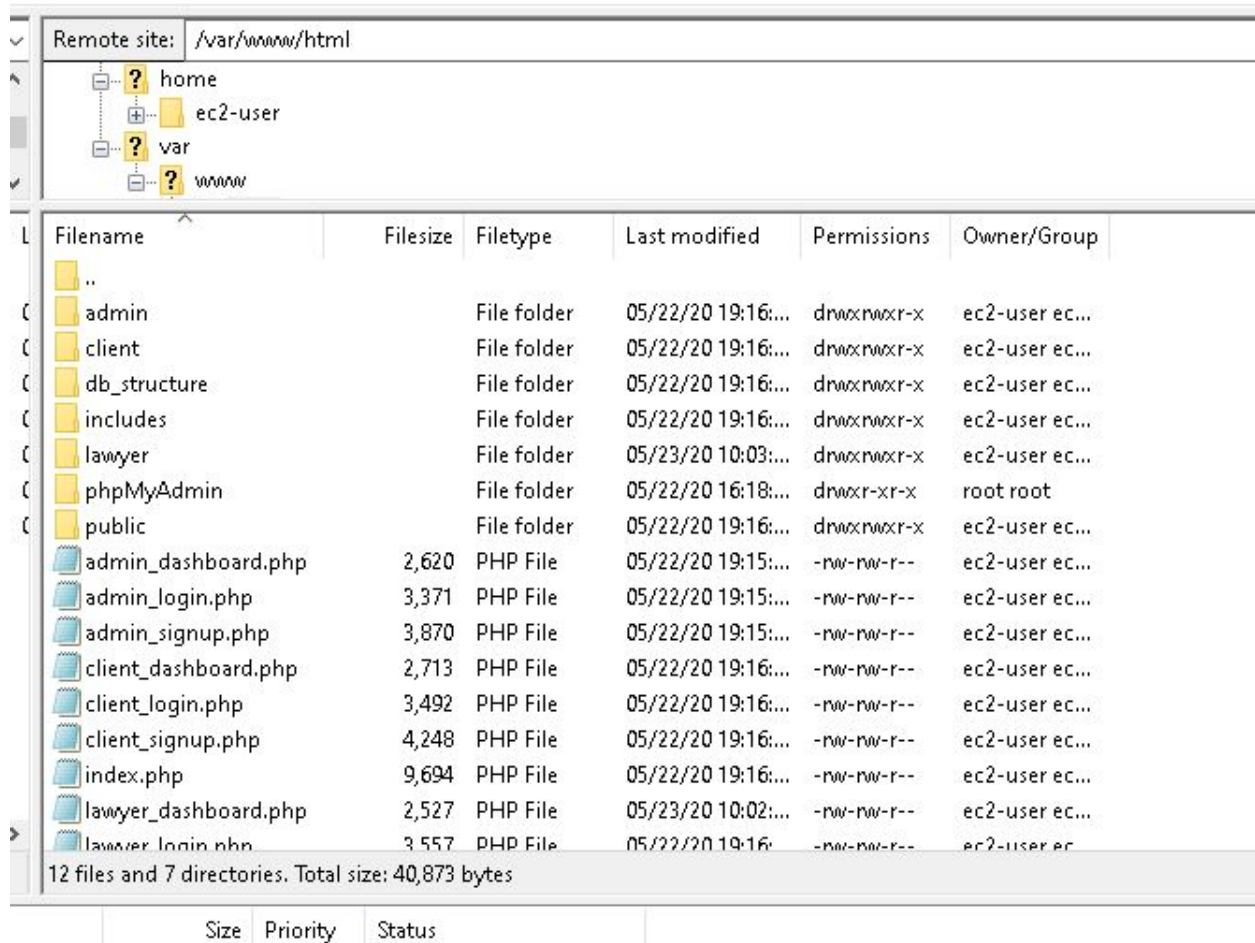
4.Steps to upload files to our PhpMyAdmin:

i.How to upload/download files to ec2 Instance using FileZilla and SFTP

1. Convert (.pem) file to (.ppk) which was downloaded during Instance creation using putty key generator file.
2. Add public IP/Elastic IP(ec2-13-127-252-157.ap-south-1.compute.amazonaws.com) in host address, add port 22, username “ec2-user” of your ec2 instance.
3. Then click on Edit--Settings--Sftp add your .ppk file
4. Then click on quick connect
5. Now just drag and drop to upload and download files

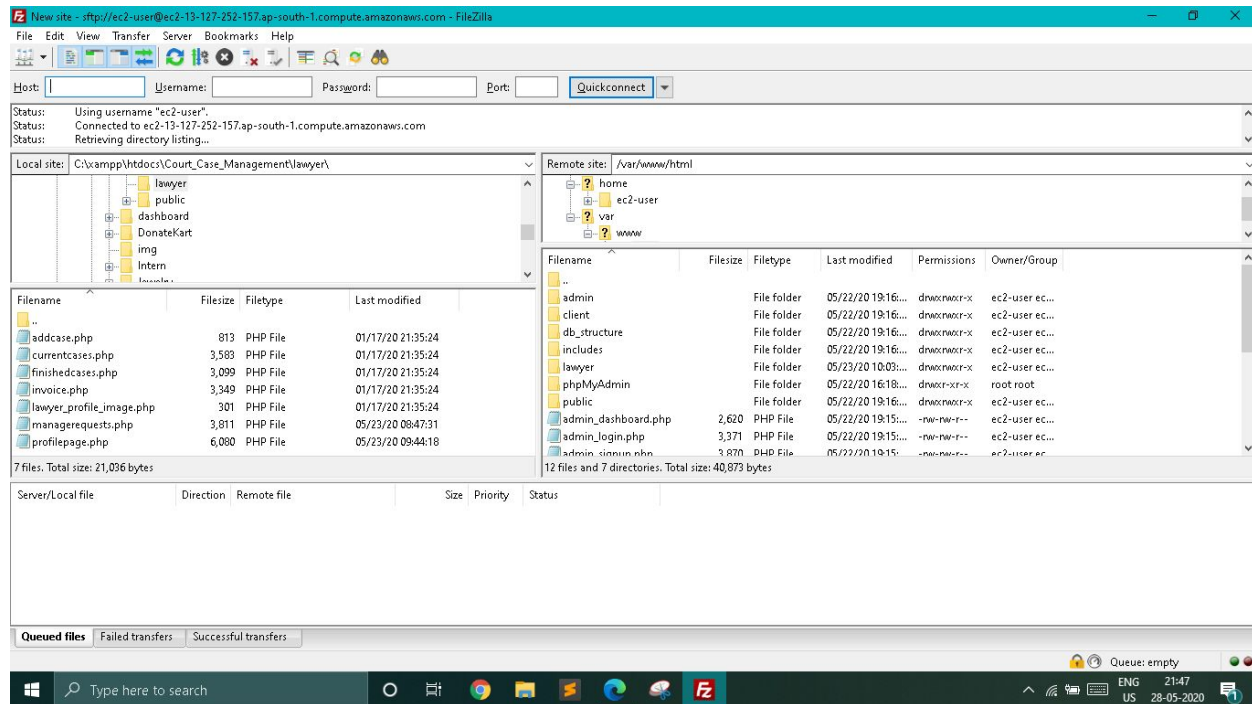


Above one is connection.



This one is our files in

<http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com>



From above we can upload our files from our local directory to our

<http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com> in `/var/www/html` folder.

Results:

Finally, it launches our website at <http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/index.php> or <http://13.127.252.157/index.php>

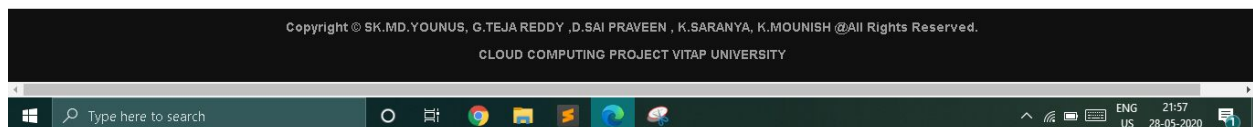
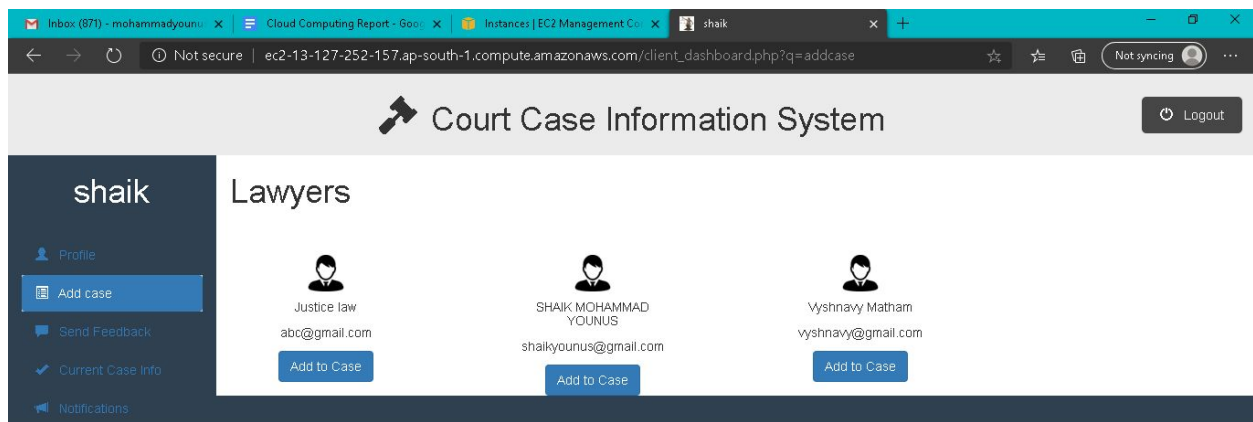
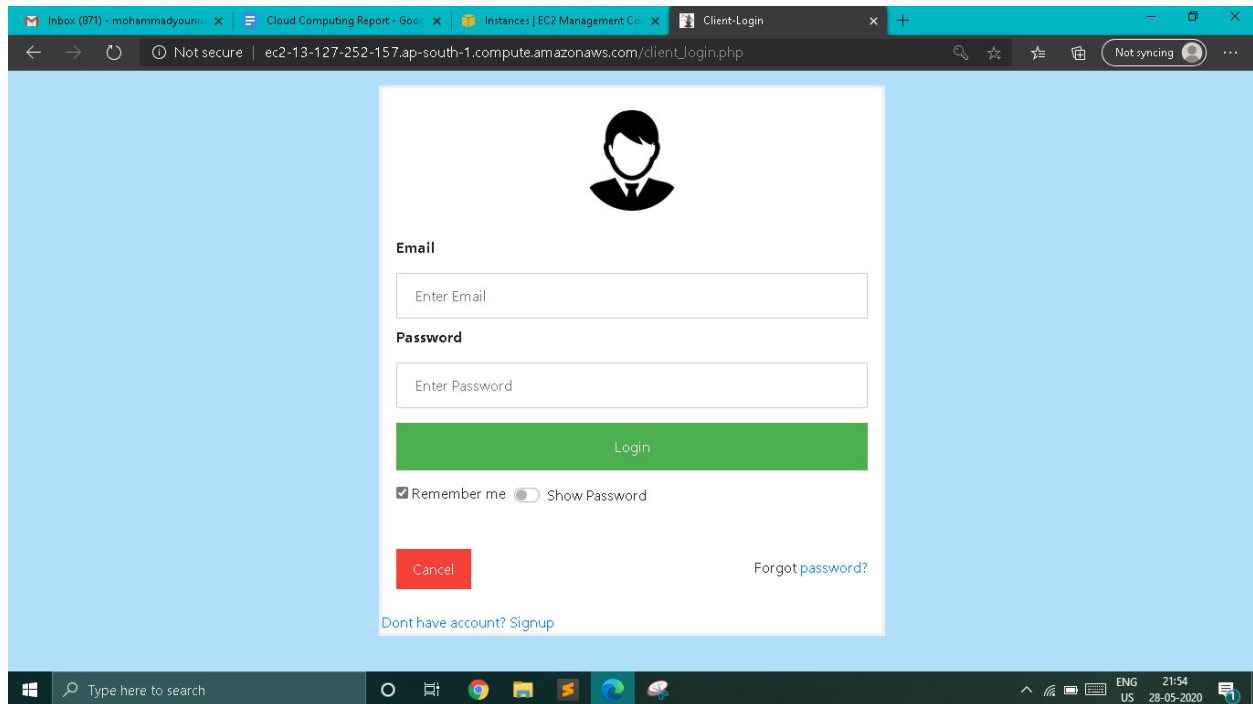
Here we have 3 modules that have been created:

- Client login
- Lawyer login
- Admin login

Client Module:

http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/client_login.php

The client login section allows the user to enter the case details and register a complaint which will be monitored by the lawyer. It also displays their case details and the status of the case.

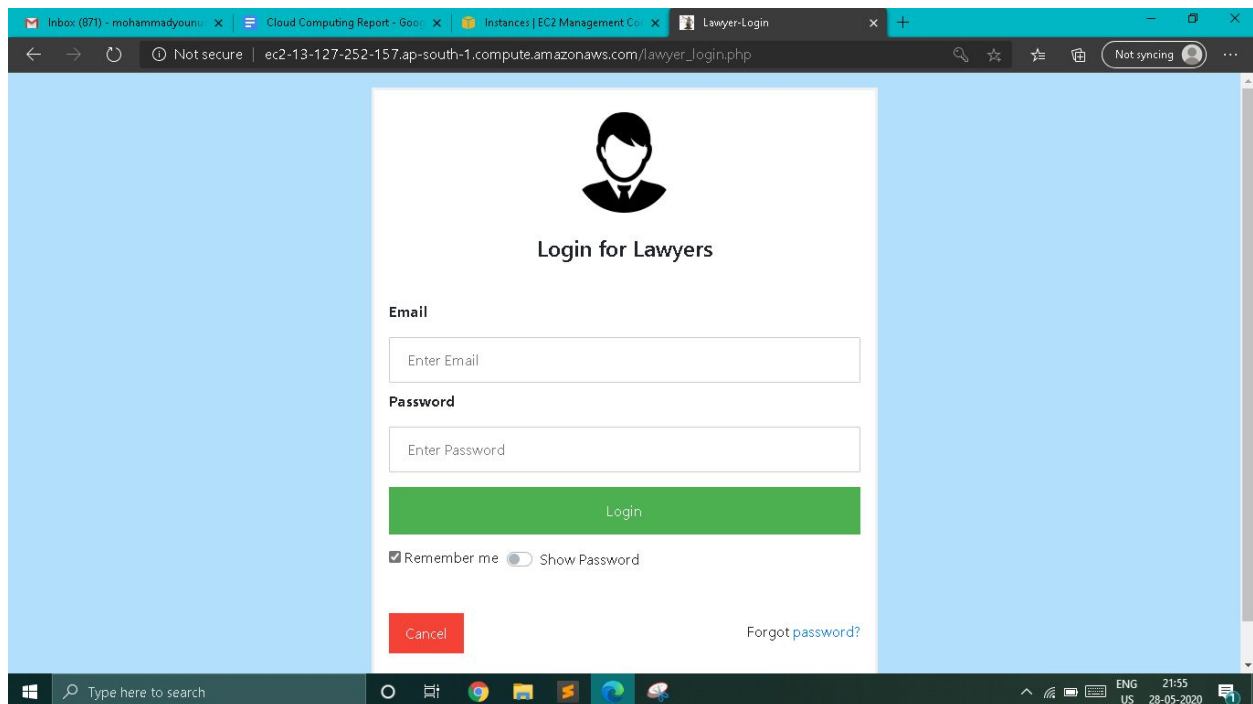


Lawyer Module:

http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/lawyer_login.php

Whereas, the lawyer login section allows the lawyer assigned for the respective case to see the case details and start investigation and he can change the status of the case if the case is closed/completed.

It also displays the number of pending cases and cases that have been completed successfully.



SHAIK MOHAMMAD

Current Cases

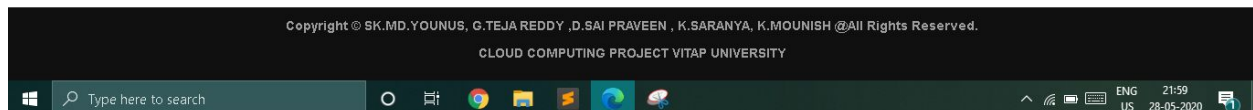
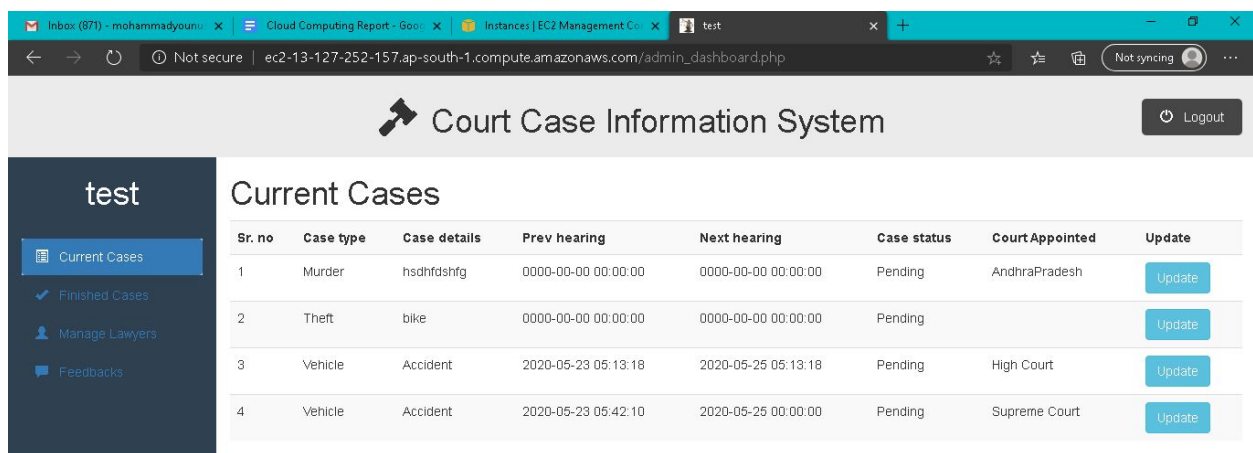
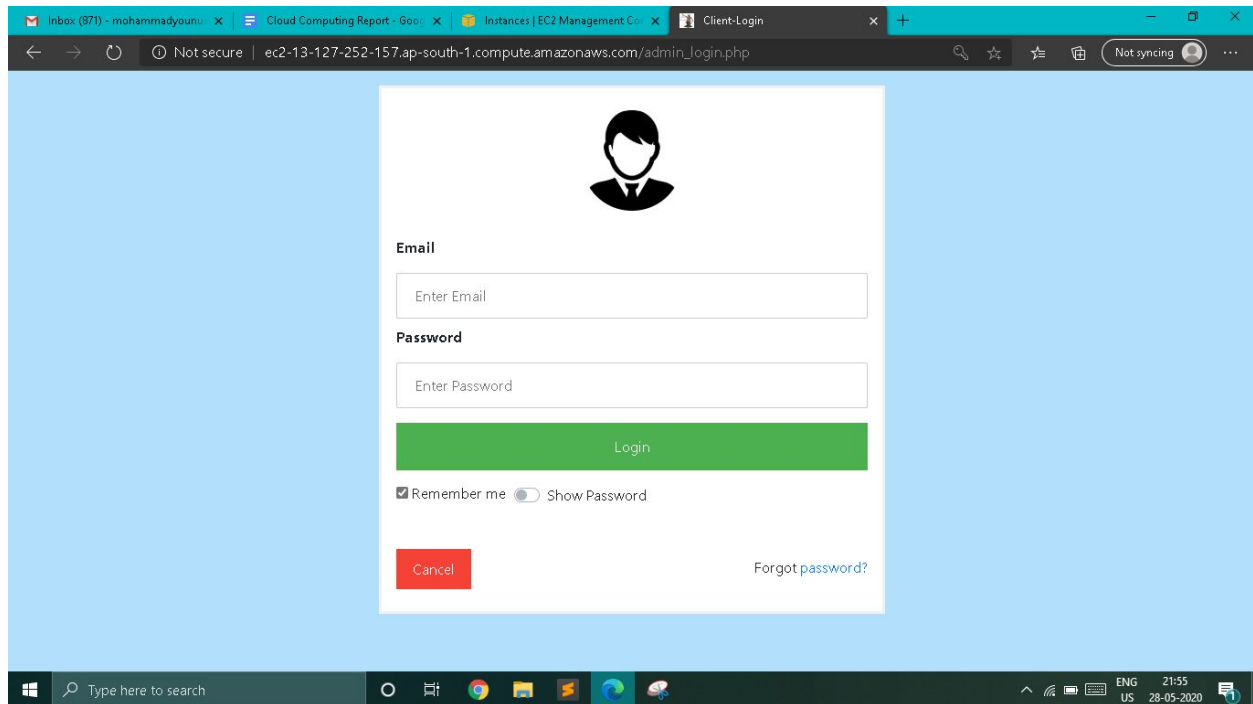
Sr. no	Case type	Case details	Prev hearing	Next hearing	Case status	Court Appointed	Update
1	Vehicle	Accident	2020-05-23 05:42:10	2020-05-25 00:00:00	Pending	Supreme Court	<button>Update</button>

Copyright © SK.MD.YOUNUS, G.TEJA REDDY ,D.SAI PRAVEEN , K.SARANYA, K.MOUNISH @All Rights Reserved.
CLOUD COMPUTING PROJECT VITAP UNIVERSITY

Admin Module:

http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/lawyer_login.php

The admin login Section can monitor the lawyer section as he can see which lawyer has been assigned to which case and he can alter the changes if he wants to.



Mysql Database:

To edit and view our database we need to go to the

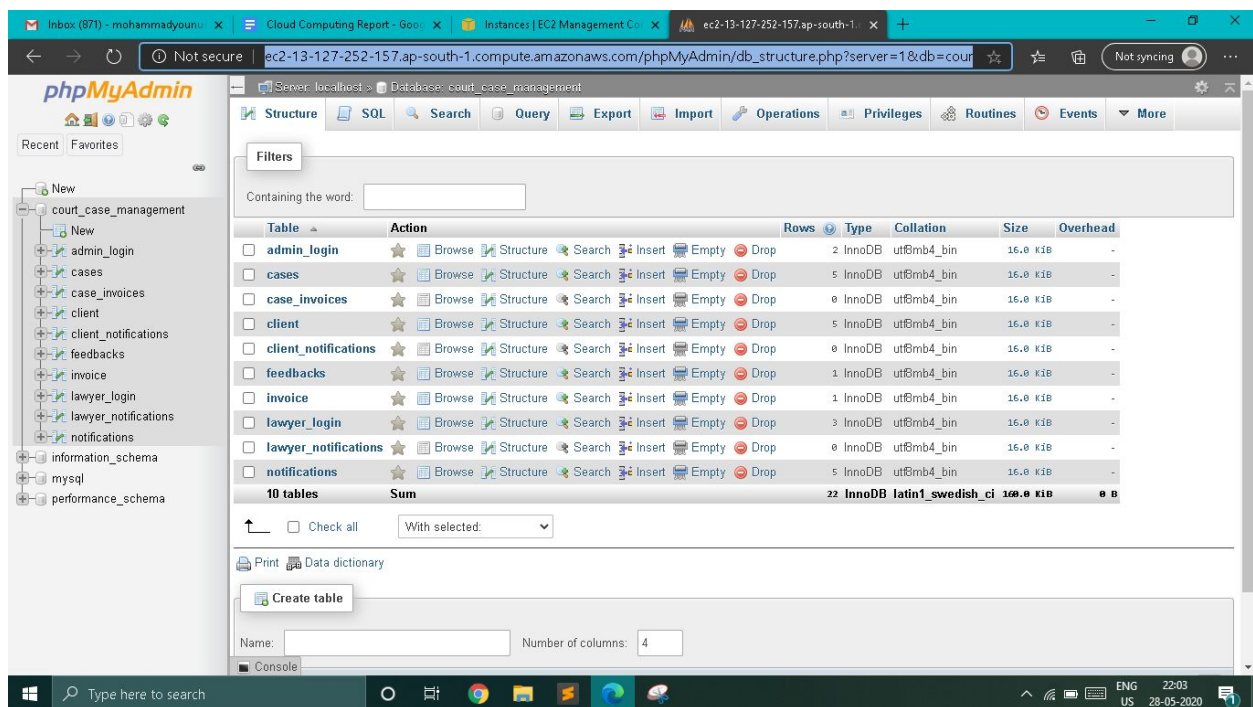
<http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/phpMyAdmin/>

Username:admin

Password: 123456

And then go to

http://ec2-13-127-252-157.ap-south-1.compute.amazonaws.com/phpMyAdmin/db_structure.php?server=1&db=court_case_management



Conclusion:

In this project, We can register our complaints as a client and can monitor our case developments. Whereas we can also login as a lawyer and can solve the pending cases assigned to the lawyer in order. Finally, these are the modules implemented in this project Court Case Management System.

This is how I used AWS Elastic Compute Service or EC2 (**Infrastructure as a Service**). Amazon Web Services takes the responsibility of networking, storage, server and virtualization and we are responsible for managing the Operating System, data and application.

References:

1. <https://docs.aws.amazon.com/quickstarts/latest/vmlaunch/step-1-launch-instance.html>
2. <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html>
3. <https://www.bing.com/search?q=Filezilla+to+upload+files+in+EC2+instance&cvid=7aece781b1c0402dacc8d70339708461&FORM=ANNTA1&PC=U531>