Dockerize a simple PHP application and deploy it on an AWS EC2 instance. This application will interact with a MySQL database hosted on AWS RDS

Task:

Objective: Dockerize a simple PHP application and deploy it on an AWS EC2 instance. This application will interact with a MySQL database hosted on AWS RDS.

Task Details:

1. Create a PHP Web Application

- Create a simple PHP application that interacts with a MySQ[1]L database.
- Take any basic application from any github repository.
- The application can be a basic form that takes user input and stores it in a MySQL database.

2. Dockerize the PHP Application

- Create a Dockerfile for the application. This Dockerfile should use an official PHP Docker image and should copy your application files into the appropriate directory within the Docker image.

3. Set Up MySQL Database on AWS RDS

- Create a MySQL database instance on AWS RDS. Connect your PHP application to this database and ensure it works as expected.

4. Manual Deployment to EC2

- Create an AWS EC2 instance and install Docker on it.
- Manually transfer your PHP application files and Dockerfile to the EC2 instance.
- On the EC2 instance, build your Docker image using the docker build command and run it using the docker run command. Ensure the PHP application is able to connect and interact with the MySQL database hosted on AWS RDS.

5. Testing

- Test your application by accessing the public IP of your EC2 instance. Verify that your application can read and write to the database.

6. Set Up CI\CD (Use AWS Codepipeline or Jenkins):

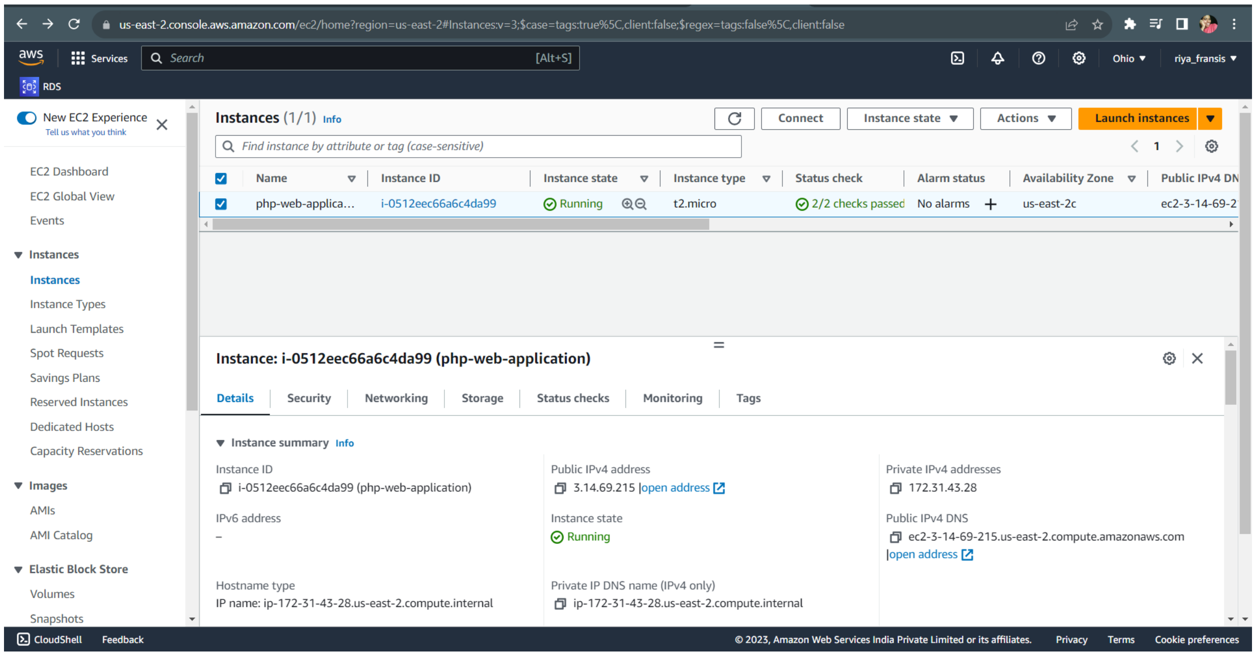
- Create a GitHub repository for your PHP application and push your code to it.
- Set up an AWS CodePipeline that uses your GitHub repository as a source.
- In the build spec file, specify commands to build your Docker image and push it to AWS Elastic Container Registry (ECR)
- Add a deployment stage
- This stage will be triggered once the build stage completes successfully.

[1] Saranya Sreedharan

- During this stage, AWS CodeDeploy will take the new Docker image, as specified in your AppSpec file, and run it on your EC2 instance.

You can use Jenkins as well.[2]

First create an ec2 instance (https,http,ssh,mysql/aurora)and mysql database in RDS. Note down the database name, username and password of the mysql for connection purposes.



Instances (1/1) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DN
php-web-applica...	i-0512eec66a6c4da99	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	ec2-3-14-69-2

Instance: i-0512eec66a6c4da99 (php-web-application)

Instance summary Info

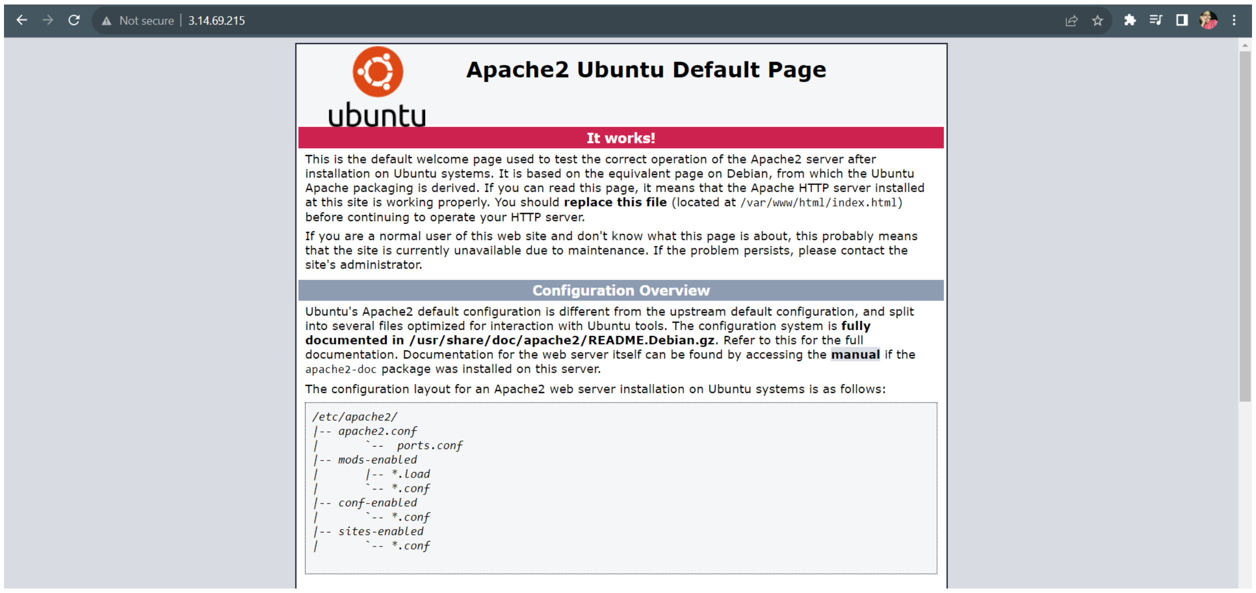
Instance ID: i-0512eec66a6c4da99 (php-web-application)
Public IPv4 address: 3.14.69.215 | open address
Private IPv4 addresses: 172.31.43.28
IPv6 address: –
Instance state: Running
Public IPv4 DNS: ec2-3-14-69-215.us-east-2.compute.amazonaws.com | open address
Hostname type: IP name: ip-172-31-43-28.us-east-2.compute.internal
Private IP DNS name (IPv4 only): ip-172-31-43-28.us-east-2.compute.internal

Then we want to install php and mysql-php dependency on our ec2 instance.

```
sudo su
sudo apt update
sudo apt install apache2
sudo apt install php libapache2-mod-php php-mysql
```

Then using the public ip of the instance, check if apache is installed or not. If installed it will show like this welcome page

[2] Saranya Sreedharan



This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

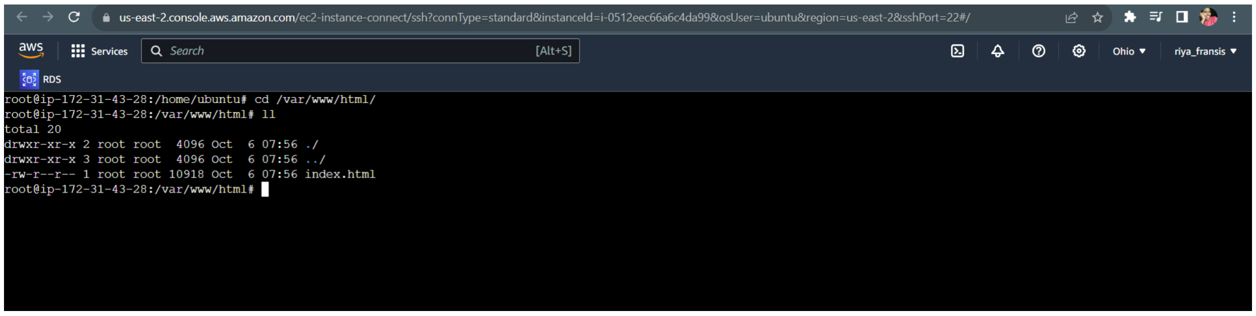
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

3

So we are going to create our simple-php application (Just demo application) in this ec2 instance . So go inside the /var/www/html/ page then see what are the files present there(do ll).


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

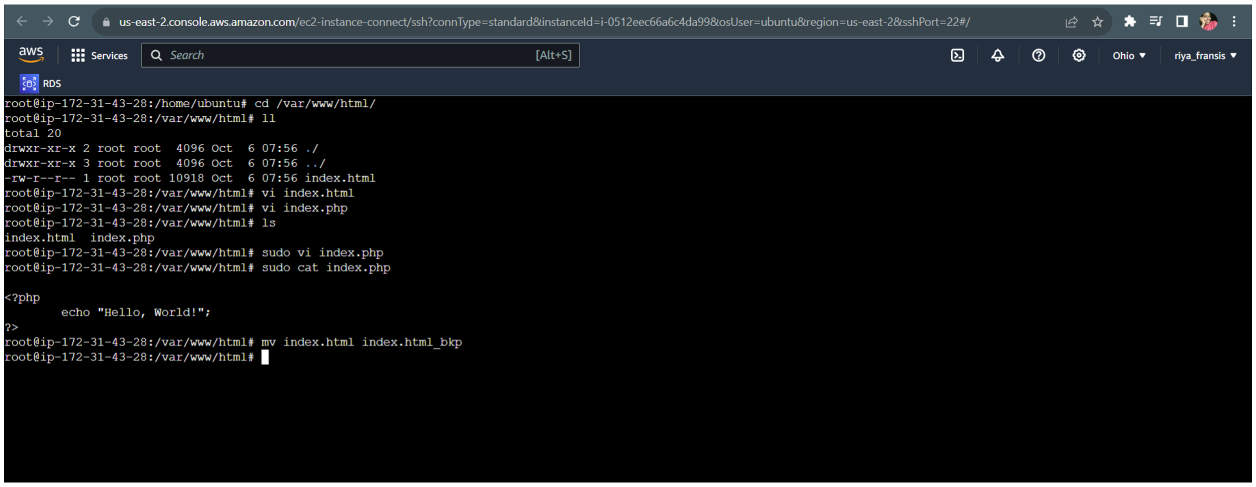
```
root@ip-172-31-43-28:/home/ubuntu# cd /var/www/html/
root@ip-172-31-43-28:/var/www/html# ll
total 20
drwxr-xr-x 2 root root  4096 Oct  6 07:56 ./
drwxr-xr-x 3 root root  4096 Oct  6 07:56 ../
-rw-r--r-- 1 root root 10918 Oct  6 07:56 index.html
root@ip-172-31-43-28:/var/www/html#
```

Here we want to create an index.php file which contains the basic php ‘hello world’ code.

```
sudo vi index.php
```

```
<?php
  echo "Hello, World!";
?>
```

3 Saranya Sreedharan

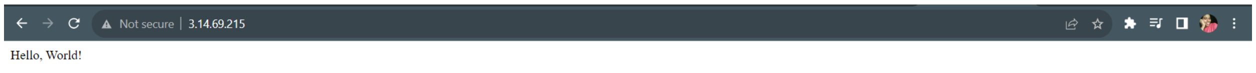

us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
root@ip-172-31-43-28:/home/ubuntu# cd /var/www/html/
root@ip-172-31-43-28:/var/www/html# ll
total 20
drwxr-xr-x 2 root root  4096 Oct  6 07:56 ./
drwxr-xr-x 3 root root  4096 Oct  6 07:56 ../
-rw-r--r-- 1 root root 10918 Oct  6 07:56 index.html
root@ip-172-31-43-28:/var/www/html# vi index.html
root@ip-172-31-43-28:/var/www/html# vi index.php
root@ip-172-31-43-28:/var/www/html# ls
index.html  index.php
root@ip-172-31-43-28:/var/www/html# sudo vi index.php
root@ip-172-31-43-28:/var/www/html# sudo cat index.php

<?php
        echo "Hello, World!";
?>
root@ip-172-31-43-28:/var/www/html# mv index.html index.html_bkp
root@ip-172-31-43-28:/var/www/html#
```

Then change the apache welcome page to our index.php code as the welcome page

mv index.html index.html_bkp

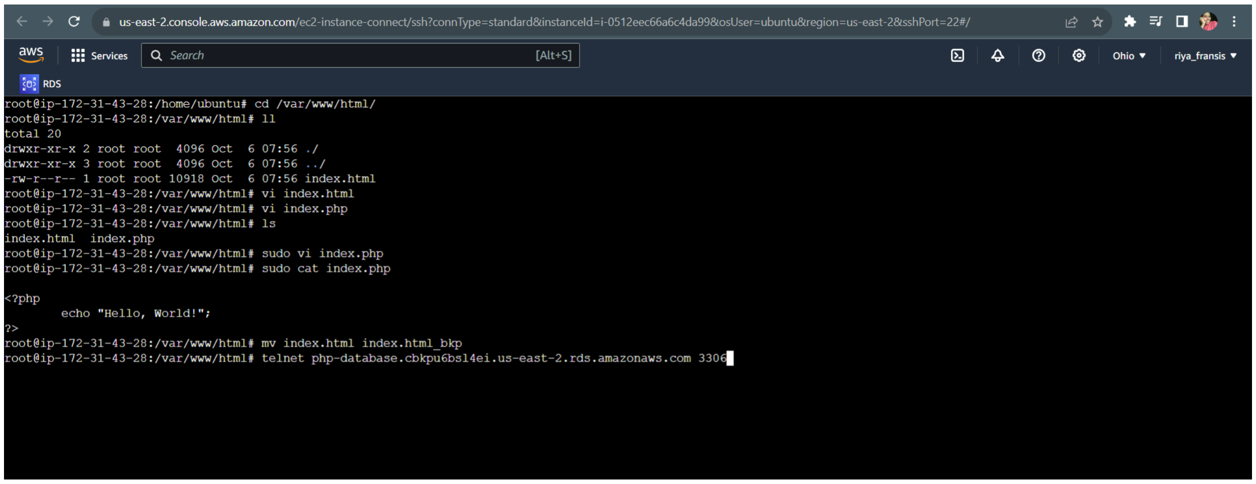
Then verify the application is accessible.



Hello, World!

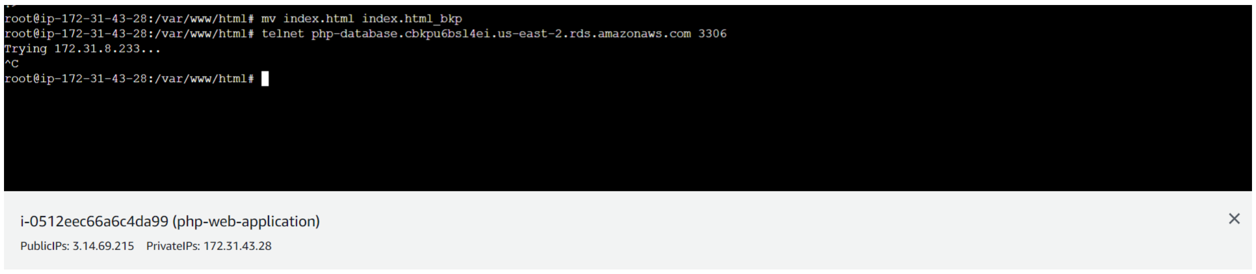
Then check that RDS is reachable or not using telnet command

telnet endpoint of RDS 3306


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
root@ip-172-31-43-28:/home/ubuntu# cd /var/www/html/
root@ip-172-31-43-28:/var/www/html# ll
total 20
drwxr-xr-x 2 root root  4096 Oct  6 07:56 ./
drwxr-xr-x 3 root root  4096 Oct  6 07:56 ../
-rw-r--r-- 1 root root 10918 Oct  6 07:56 index.html
root@ip-172-31-43-28:/var/www/html# vi index.html
root@ip-172-31-43-28:/var/www/html# vi index.php
root@ip-172-31-43-28:/var/www/html# ls
index.html  index.php
root@ip-172-31-43-28:/var/www/html# sudo vi index.php
root@ip-172-31-43-28:/var/www/html# sudo cat index.php

<?php
        echo "Hello, World!";
?>
root@ip-172-31-43-28:/var/www/html# mv index.html index.html_bkp
root@ip-172-31-43-28:/var/www/html# telnet php-database.cbkpu6bsl4ei.us-east-2.rds.amazonaws.com 3306
```

If it is showing and not connected then go to the RDS security group, add mysql/Aurora and source as the security group of the ec2 instance.

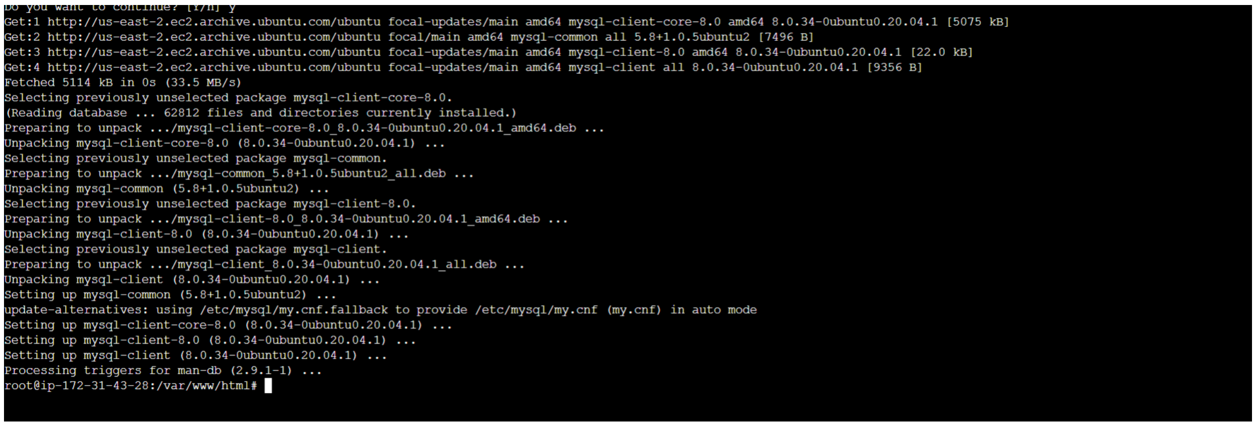


```
root@ip-172-31-43-28:/var/www/html# mv index.html index.html_bkp
root@ip-172-31-43-28:/var/www/html# telnet php-database.cbkpu6bsl4ei.us-east-2.rds.amazonaws.com 3306
Trying 172.31.8.233...
^C
root@ip-172-31-43-28:/var/www/html# 
```

i-0512eec66a6c4da99 (php-web-application)
PublicIPs: 3.14.69.215 PrivateIPs: 172.31.43.28

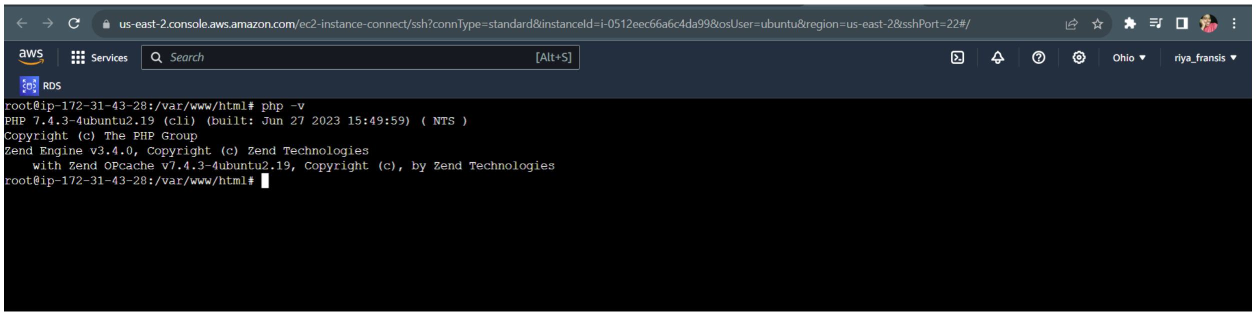
For the connection we need a mysql client. So install mysql client.

sudo apt install mysql-client



```
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-core-8.0 amd64 8.0.34-0ubuntu0.20.04.1 [5075 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/main amd64 mysql-common all 5.8+1.0.5ubuntu2 [7496 B]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-8.0 amd64 8.0.34-0ubuntu0.20.04.1 [22.0 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client all 8.0.34-0ubuntu0.20.04.1 [9356 B]
Fetched 5114 kB in 0s (33.5 MB/s)
Selecting previously unselected package mysql-client-core-8.0.
(Reading database ... 62812 files and directories currently installed.)
Preparing to unpack .../mysql-client-core-8.0_8.0.34-0ubuntu0.20.04.1_amd64.deb ...
Unpacking mysql-client-core-8.0 (8.0.34-0ubuntu0.20.04.1) ...
Selecting previously unselected package mysql-common.
Preparing to unpack .../mysql-common_5.8+1.0.5ubuntu2_all.deb ...
Unpacking mysql-common (5.8+1.0.5ubuntu2) ...
Selecting previously unselected package mysql-client-8.0.
Preparing to unpack .../mysql-client-8.0_8.0.34-0ubuntu0.20.04.1_amd64.deb ...
Unpacking mysql-client-8.0 (8.0.34-0ubuntu0.20.04.1) ...
Selecting previously unselected package mysql-client.
Preparing to unpack .../mysql-client_8.0.34-0ubuntu0.20.04.1_all.deb ...
Unpacking mysql-client (8.0.34-0ubuntu0.20.04.1) ...
Setting up mysql-common (5.8+1.0.5ubuntu2) ...
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Setting up mysql-client-core-8.0 (8.0.34-0ubuntu0.20.04.1) ...
Setting up mysql-client-8.0 (8.0.34-0ubuntu0.20.04.1) ...
Setting up mysql-client (8.0.34-0ubuntu0.20.04.1) ...
Processing triggers for man-db (2.9.1-1) ...
root@ip-172-31-43-28:/var/www/html# 
```

To Check the php version,
php -v


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
root@ip-172-31-43-28:/var/www/html# php -v
PHP 7.4.3-4ubuntu2.19 (cli) (built: Jun 27 2023 15:49:59) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.3-4ubuntu2.19, Copyright (c), by Zend Technologies
root@ip-172-31-43-28:/var/www/html# 
```

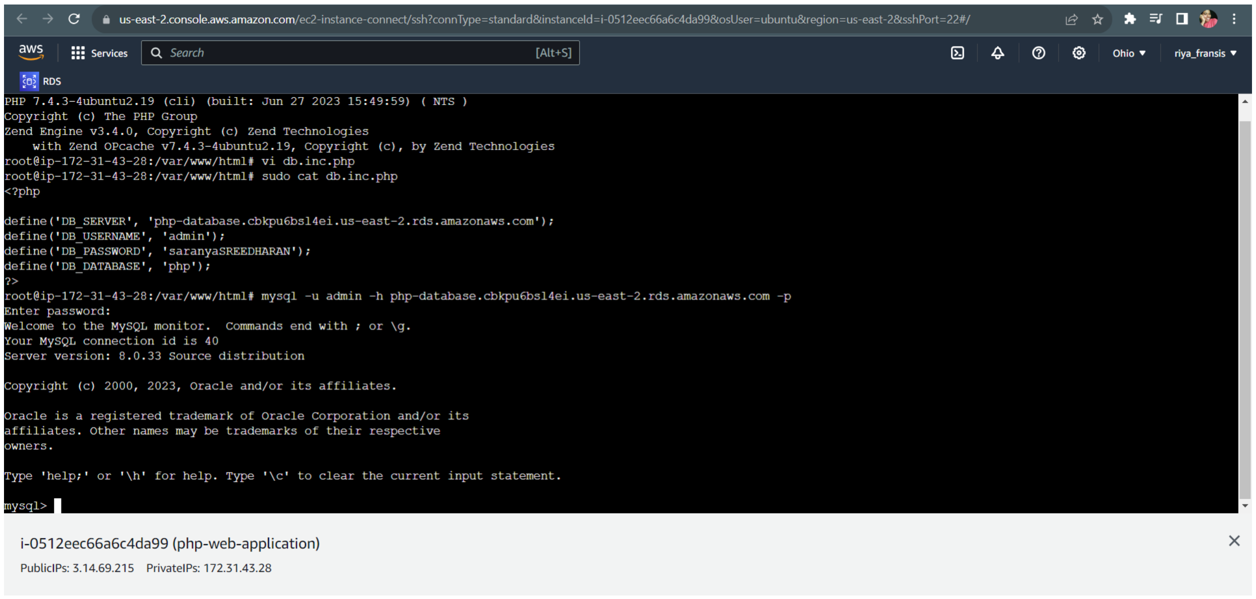
We need to pass the credentials to connect with database
So add credentials in the db.inc.php

[4] Saranya Sreedharan

sudo vi db.inc.php

```
<?php

define('DB_SERVER', 'db_instance_endpoint');
define('DB_USERNAME', 'tutorial_user');
define('DB_PASSWORD', 'master password');
define('DB_DATABASE', 'sample');
?>
```

Give the endpoint, username and password of mysql database. That ‘sample’ is representing any of the database name which we are going to create


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
PHP 7.4.3-4ubuntu2.19 (cli) (built: Jun 27 2023 15:49:59) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.3-4ubuntu2.19, Copyright (c), by Zend Technologies
root@ip-172-31-43-28:/var/www/html# vi db.inc.php
root@ip-172-31-43-28:/var/www/html# sudo cat db.inc.php
<?php

define('DB_SERVER', 'php-database.cbkpu6bsl4ei.us-east-2.rds.amazonaws.com');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'saranyaSREEDHARAN');
define('DB_DATABASE', 'php');
?>
root@ip-172-31-43-28:/var/www/html# mysql -u admin -h php-database.cbkpu6bsl4ei.us-east-2.rds.amazonaws.com -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 40
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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>
```

i-0512eec66a6c4da99 (php-web-application)

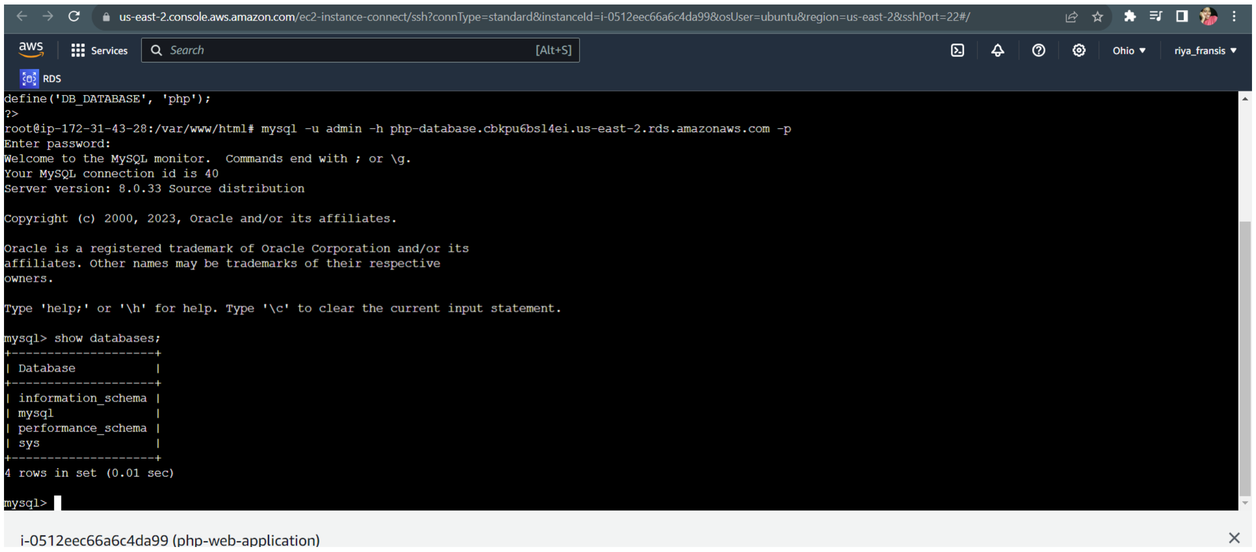
PublicIPs: 3.14.69.215 PrivateIPs: 172.31.43.28

5

Then connect with the mysql

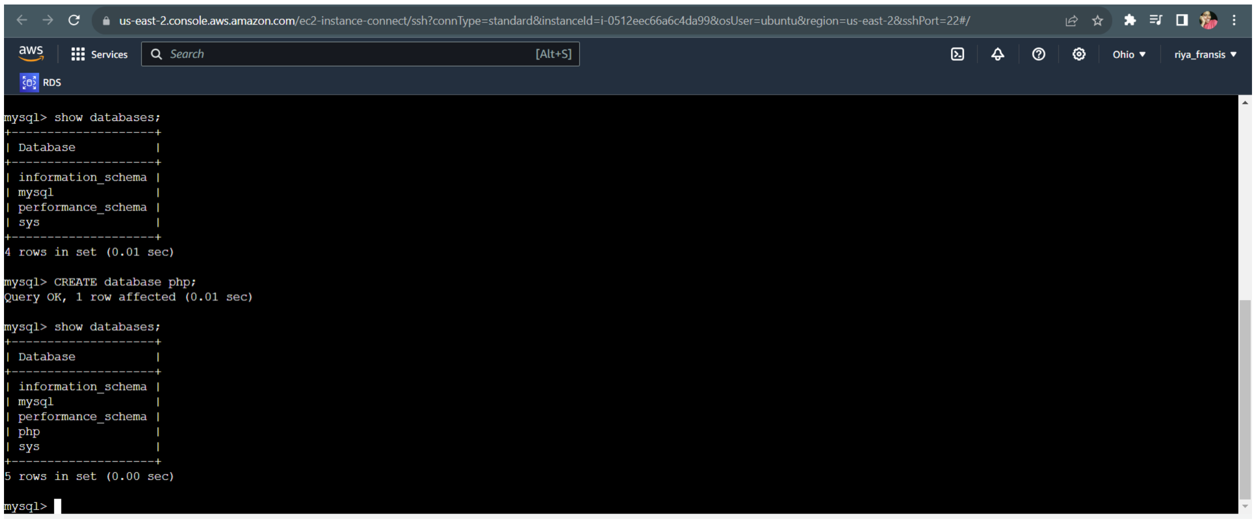
```
mysql -u username  -h endpoint -p password
```

[5] Saranya Sreedharan


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

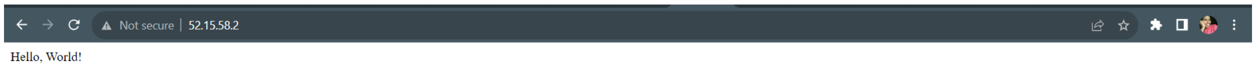
```
define('DB_DATABASE', 'php');
?>
root@ip-172-31-43-28:/var/www/html# mysql -u admin -h php-database.cbkpu6bs14ei.us-east-2.rds.amazonaws.com -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 40
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)

mysql> 
```

show databases;
CREATE database php


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)

mysql> CREATE database php;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| php                |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> 
```

Application is available.
[6]



Hello, World!

[6] Saranya Sreedharan

sudo vi samplepage.php

Then add the below content for the connection.
Refer amazon official page

```
<?php include "../inc/dbinfo.inc"; ?>
<html>
<body>
<h1>Sample page</h1>
<?php

  /* Connect to MySQL and select the database. */
  $connection = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD);

  if (mysqli_connect_errno()) echo "Failed to connect to MySQL: " . mysqli_connect_error();

  $database = mysqli_select_db($connection, DB_DATABASE);

  /* Ensure that the EMPLOYEES table exists. */
  VerifyEmployeesTable($connection, DB_DATABASE);

  /* If input fields are populated, add a row to the EMPLOYEES table. */
  $employee_name = htmlentities($_POST['NAME']);
  $employee_address = htmlentities($_POST['ADDRESS']);

  if (strlen($employee_name) || strlen($employee_address)) {
    AddEmployee($connection, $employee_name, $employee_address);
  }
?>

<!-- Input form -->
<form action="<?PHP echo $_SERVER['SCRIPT_NAME'] ?>" method="POST">
  <table border="0">
    <tr>
      <td>NAME</td>
      <td>ADDRESS</td>
    </tr>
    <tr>
      <td>
        <input type="text" name="NAME" maxlength="45" size="30" />
      </td>
[7]    <td>
        <input type="text" name="ADDRESS" maxlength="90" size="60" />
```

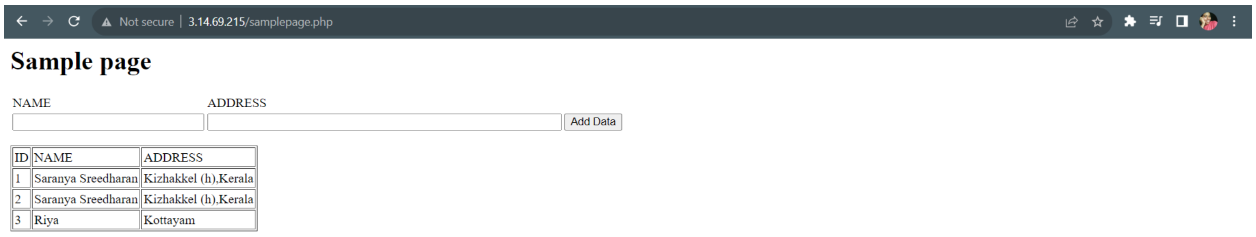
[7] Saranya Sreedharan

```
      </td>
      <td>
        <input type="submit" value="Add Data" />
      </td>
    </tr>
  </table>
</form>

<!-- Display table data. -->
<table border="1" cellpadding="2" cellspacing="2">
  <tr>
    <td>ID</td>
    <td>NAME</td>
    <td>ADDRESS</td>
  </tr>

<?php

$result = mysqli_query($connection, "SELECT * FROM EMPLOYEES");

while($query_data = mysqli_fetch_row($result)) {
  echo "<tr>";
  echo "<td>",$query_data[0], "</td>",
       "<td>",$query_data[1], "</td>",
       "<td>",$query_data[2], "</td>";
  echo "</tr>";
}
?>

</table>

<!-- Clean up. -->
<?php

  mysqli_free_result($result);
  mysqli_close($connection);

?>

</body>
```
[8]`</html>`

[8] Saranya Sreedharan

```
<?php

/* Add an employee to the table. */
function AddEmployee($connection, $name, $address) {
   $n = mysqli_real_escape_string($connection, $name);
   $a = mysqli_real_escape_string($connection, $address);

   $query = "INSERT INTO EMPLOYEES (NAME, ADDRESS) VALUES ('$n', '$a');";

   if(!mysqli_query($connection, $query)) echo("<p>Error adding employee data.</p>");
}

/* Check whether the table exists and, if not, create it. */
function VerifyEmployeesTable($connection, $dbName) {
  if(!TableExists("EMPLOYEES", $connection, $dbName))
  {
     $query = "CREATE TABLE EMPLOYEES (
         ID int(11) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
         NAME VARCHAR(45),
         ADDRESS VARCHAR(90)
       )";

     if(!mysqli_query($connection, $query)) echo("<p>Error creating table.</p>");
  }
}

/* Check for the existence of a table. */
function TableExists($tableName, $connection, $dbName) {
  $t = mysqli_real_escape_string($connection, $tableName);
  $d = mysqli_real_escape_string($connection, $dbName);

  $checktable = mysqli_query($connection,
      "SELECT TABLE_NAME FROM information_schema.TABLES WHERE TABLE_NAME = '$t'
AND TABLE_SCHEMA = '$d'");

  if(mysqli_num_rows($checktable) > 0) return true;

  return false;
}
?>
```

Now we are able communicate with the database as well.



Not secure | 3.14.69.215/samplepage.php

Sample page

NAME ADDRESS Add Data

ID	NAME	ADDRESS
1	Saranya Sreedharan	Kizhakkel (h),Kerala
2	Saranya Sreedharan	Kizhakkel (h),Kerala
3	Riya	Kottayam

So now the application is available in the public ip of the ec2 instance. Now we are going to dockerize the application. For that install docker in the ec2 instance.

Sudo apt install -y docker.io

Then create Dockerfile. (Make sure you are creating your Dockerfile inside /var/www/html/ folder because all your application code is available there.)

This is the Dockerfile.

```
# Use an official PHP runtime as the base image
FROM php:7.4-apache

# Set the working directory to /var/www/html
WORKDIR /var/www/html

# Install the PHP MySQL extension
RUN docker-php-ext-install mysqli

# Copy the application code into the container
COPY . /var/www/html

# Expose port 80 for Apache
EXPOSE 80

# Start the Apache web server
```

[9] Saranya Sreedharan

```
CMD ["apache2-foreground"]
```

Then build the image and run the container

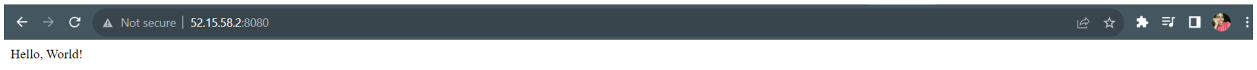
Build the docker image

```
docker build -t sarus23/my-php-app:1.0 .
```

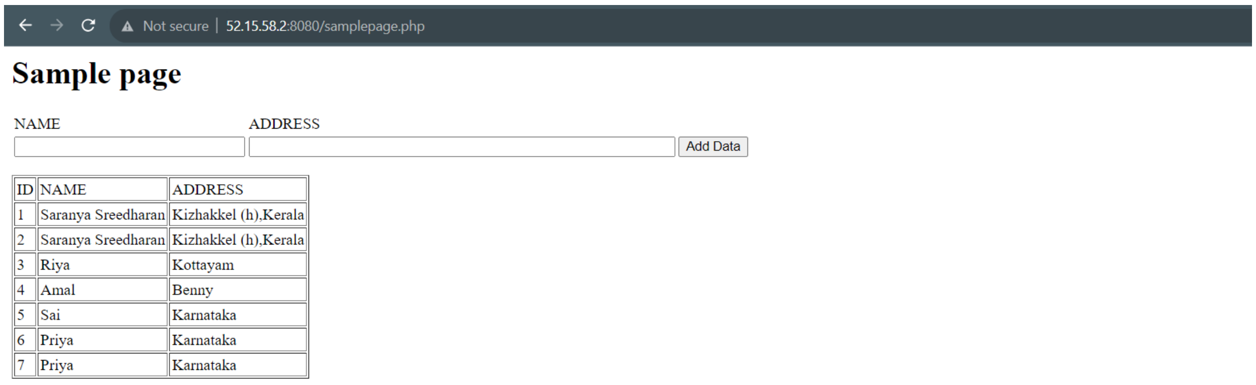
Then run the application

```
docker run -d -p 8080:80 sarus23/my-php-app:1.0
```

Our application is available in port 8080. So we containerized[10] our application and verified it in our local machine.



Hello, World!



Sample page

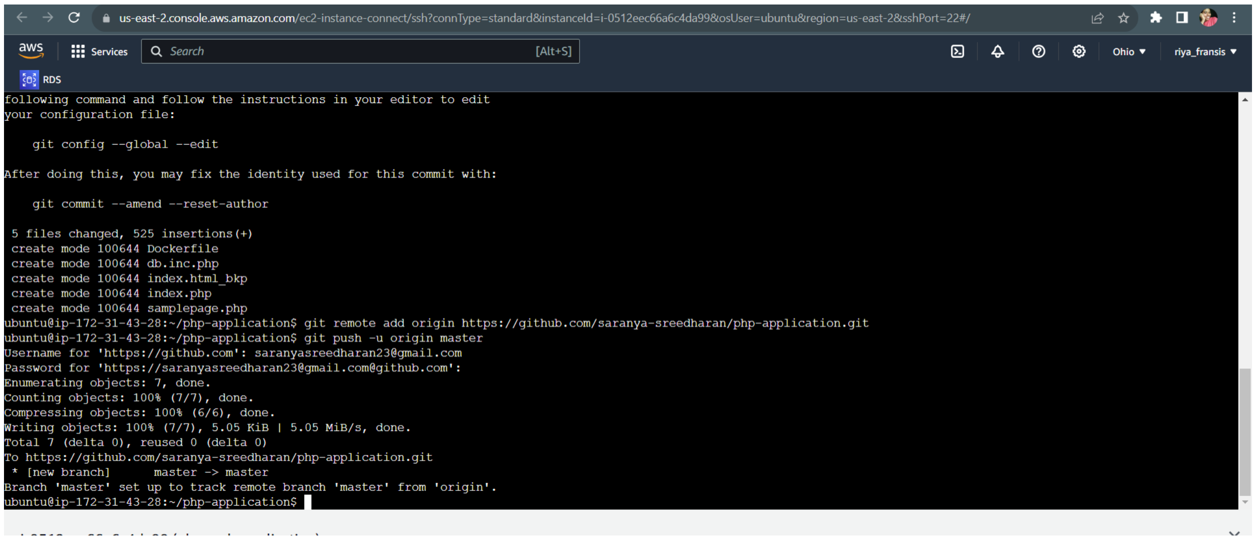
NAME ADDRESS

ID	NAME	ADDRESS
1	Saranya Sreedharan	Kizhakkel (h),Kerala
2	Saranya Sreedharan	Kizhakkel (h),Kerala
3	Riya	Kottayam
4	Amal	Benny
5	Sai	Karnataka
6	Priya	Karnataka
7	Priya	Karnataka

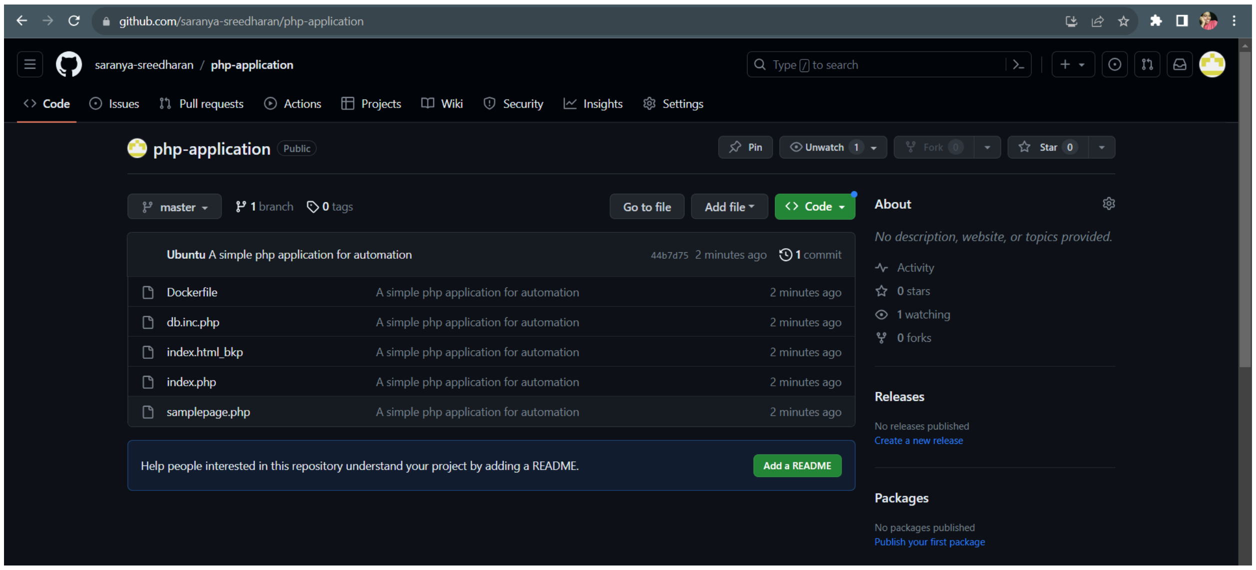
So next we will upload this code in a github repository. The code is available in

[10] Saranya Sreedharan

https://github.com/saranya-sreedharan/php-application


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0512eec66a6c4da99&osUser=ubuntu&region=us-east-2&sshPort=22#/
aws
Services
Search
[Alt+S]
Ohio
riya_fransis
RDS

```
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 5 files changed, 525 insertions(+)
 create mode 100644 Dockerfile
 create mode 100644 db.inc.php
 create mode 100644 index.html_bkp
 create mode 100644 index.php
 create mode 100644 samplepage.php
ubuntu@ip-172-31-43-28:~/php-application$ git remote add origin https://github.com/saranya-sreedharan/php-application.git
ubuntu@ip-172-31-43-28:~/php-application$ git push -u origin master
Username for 'https://github.com': saranyasreedharan23@gmail.com
Password for 'https://saranyasreedharan23@gmail.com@github.com':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 5.05 KiB | 5.05 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/saranya-sreedharan/php-application.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
ubuntu@ip-172-31-43-28:~/php-application$ 
```



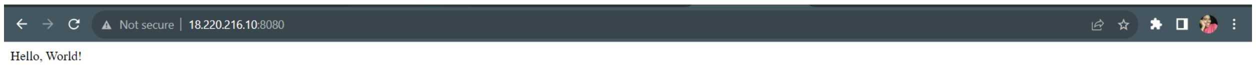
Ubuntu A simple php application for automation 44b7d75 2 minutes ago 1 commit

Dockerfile	A simple php application for automation	2 minutes ago
db.inc.php	A simple php application for automation	2 minutes ago
index.html_bkp	A simple php application for automation	2 minutes ago
index.php	A simple php application for automation	2 minutes ago
samplepage.php	A simple php application for automation	2 minutes ago

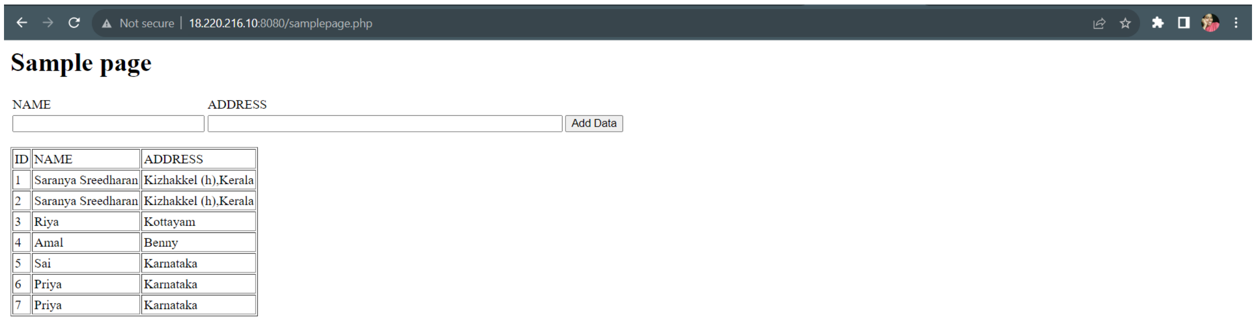
Help people interested in this repository understand your project by adding a README.

To verify that I created another ec2 instance and clone the repo then installed docker in the system then run the image the application is working fine now then to get the database connection we need to manually connect the instance with the database. Take the ec2 instance [11]security group then add that security group in the RDS instance security group. Then make sure that application is available.

[11] Saranya Sreedharan


Not secure | 18.220.216.10:8080

Hello, World!


Not secure | 18.220.216.10:8080/samplepage.php

Sample page

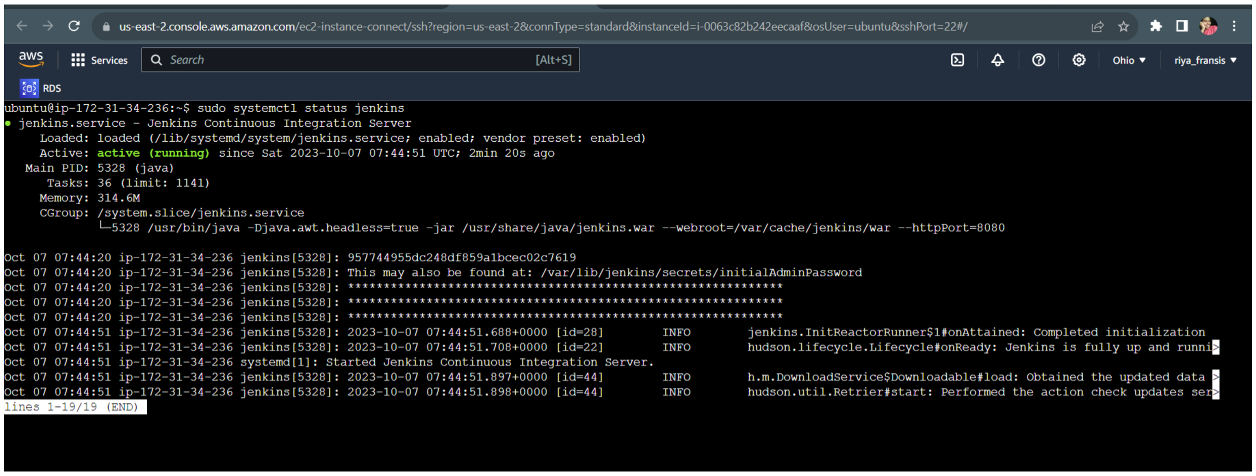
NAME ADDRESS

Add Data

ID	NAME	ADDRESS
1	Saranya Sreedharan	Kizhakkel (h),Kerala
2	Saranya Sreedharan	Kizhakkel (h),Kerala
3	Riya	Kottayam
4	Amal	Benny
5	Sai	Karnataka
6	Priya	Karnataka
7	Priya	Karnataka

Then we need to create a CICD pipeline using jenkins. So we are installing Jenkins using the below commands.

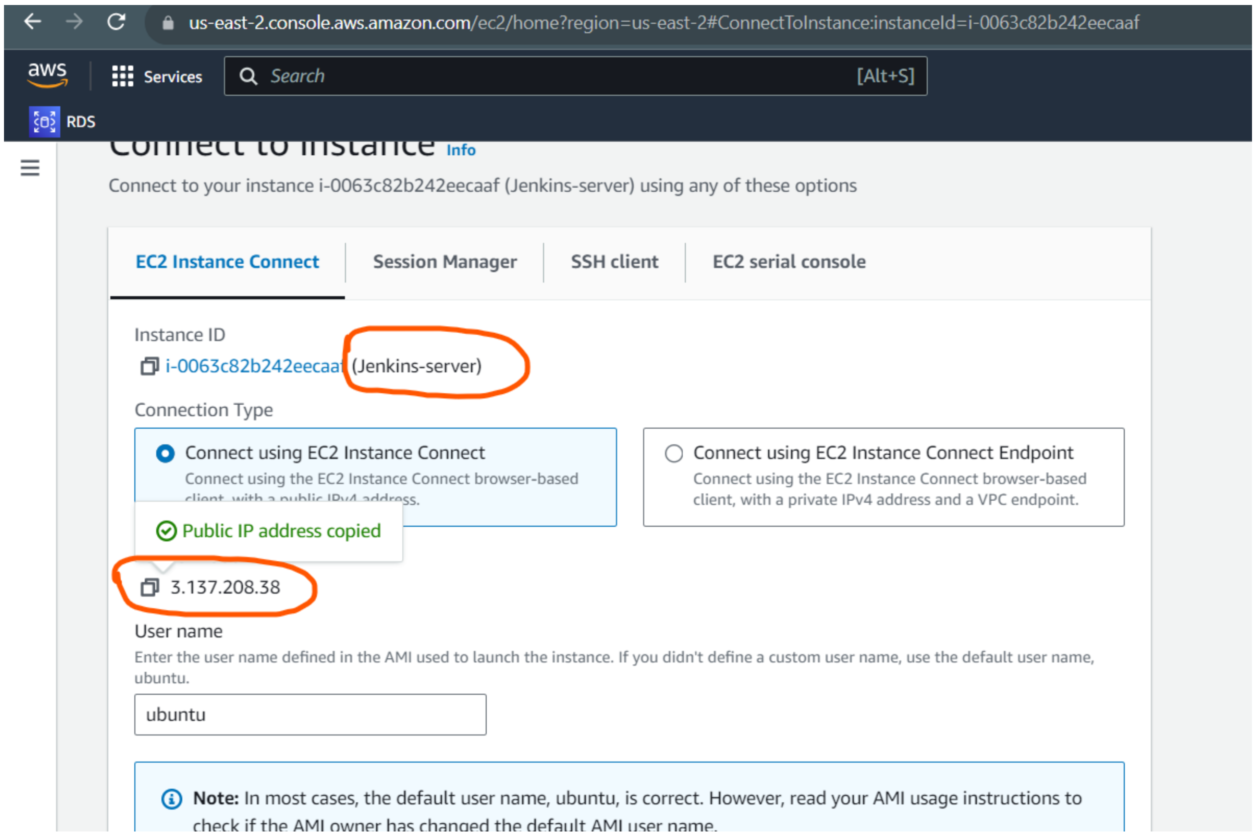
Jenkins installation document


us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-2&connType=standard&instanceId=i-0063c82b242eecaaf&osUser=ubuntu&sshPort=22#/

```
ubuntu@ip-172-31-34-236:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-10-07 07:44:51 UTC; 2min 20s ago
   Main PID: 5328 (java)
      Tasks: 36 (limit: 1141)
     Memory: 314.6M
     CGroup: /system.slice/jenkins.service
             └─5328 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Oct 07 07:44:20 ip-172-31-34-236 jenkins[5328]: 957744955dc248df859a1bcec02c7619
Oct 07 07:44:20 ip-172-31-34-236 jenkins[5328]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Oct 07 07:44:20 ip-172-31-34-236 jenkins[5328]: *************************************************************
Oct 07 07:44:20 ip-172-31-34-236 jenkins[5328]: *************************************************************
Oct 07 07:44:20 ip-172-31-34-236 jenkins[5328]: *************************************************************
Oct 07 07:44:51 ip-172-31-34-236 jenkins[5328]: 2023-10-07 07:44:51.688+0000 [id=28]        INFO        jenkins.InitReactorRunner$1#onAttained: Completed initialization
Oct 07 07:44:51 ip-172-31-34-236 jenkins[5328]: 2023-10-07 07:44:51.708+0000 [id=22]        INFO        hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and runni>
Oct 07 07:44:51 ip-172-31-34-236 systemd[1]: Started Jenkins Continuous Integration Server.
Oct 07 07:44:51 ip-172-31-34-236 jenkins[5328]: 2023-10-07 07:44:51.897+0000 [id=44]        INFO        h.m.DownloadService$Downloadable#load: Obtained the updated data >
Oct 07 07:44:51 ip-172-31-34-236 jenkins[5328]: 2023-10-07 07:44:51.898+0000 [id=44]        INFO        hudson.util.Retrier#start: Performed the action check updates ser>
lines 1-19/19 (END)
```

12

Jenkins installed and running successfully.



us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#ConnectToInstance:instanceId=i-0063c82b242eecaaf

Connect to instance Info

Connect to your instance i-0063c82b242eecaaf (Jenkins-server) using any of these options

Instance ID

i-0063c82b242eecaaf (Jenkins-server)

Connection Type

[✓ Connect using EC2 Instance Connect] Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

Connect using EC2 Instance Connect Endpoint: Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address copied

3.137.208.38

User name

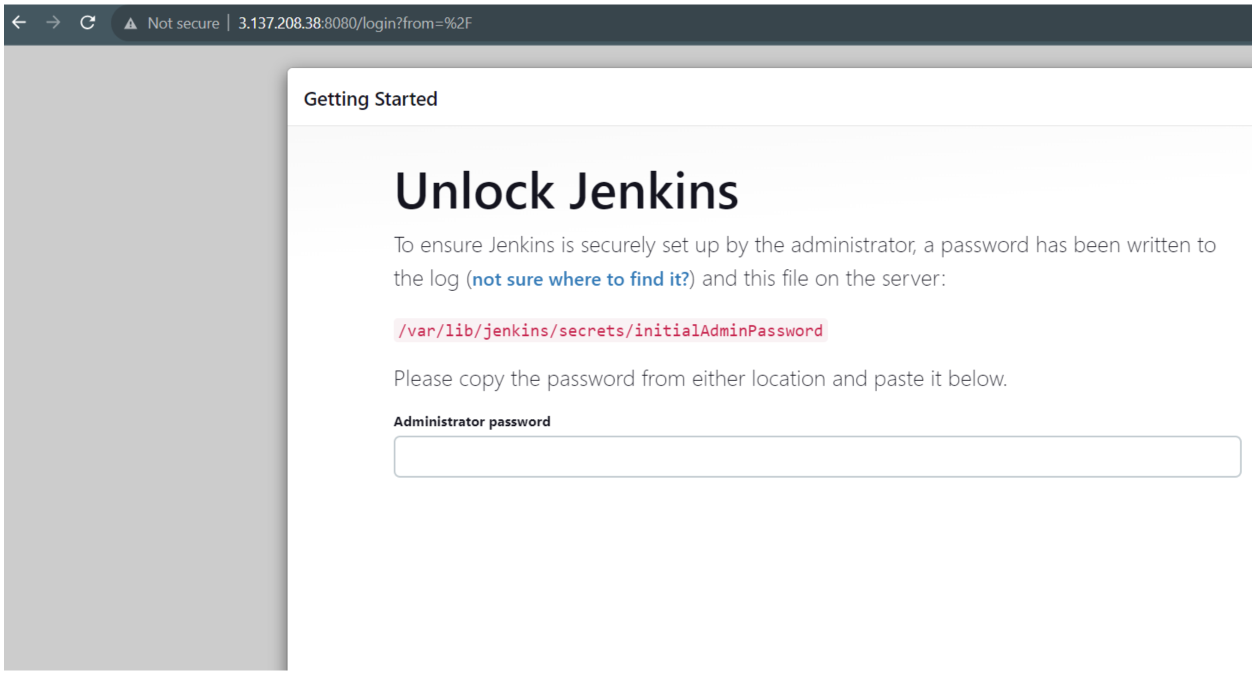
Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.

ubuntu

Note: In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Setup the jenkins. Install recommended plugins.

[12] Saranya Sreedharan



Unlock Jenkins

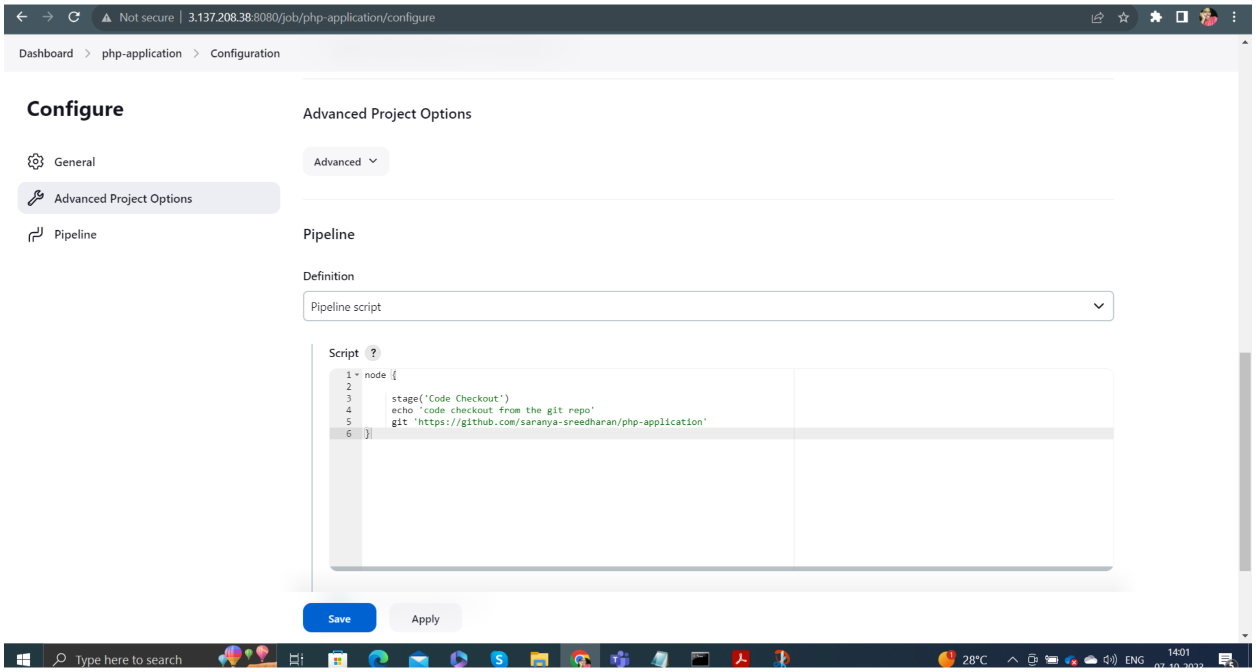
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Code checkout stage. Clone the project from the git repository.



Advanced Project Options

Pipeline

Definition: Pipeline script

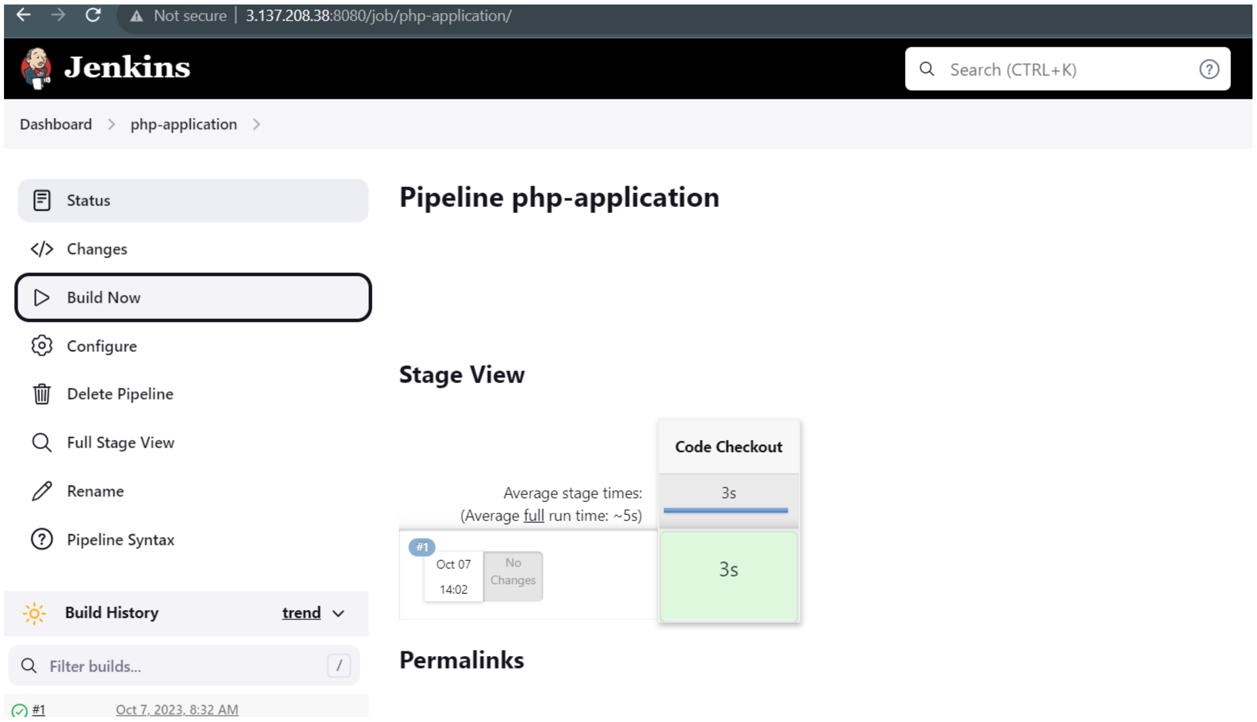
Script

```
node {

     stage('Code Checkout')
     echo 'code checkout from the git repo'
     git 'https://github.com/saranya-sreedharan/php-application'
}
```

13

Successfully clone the project.

13 Saranya Sreedharan



Pipeline php-application

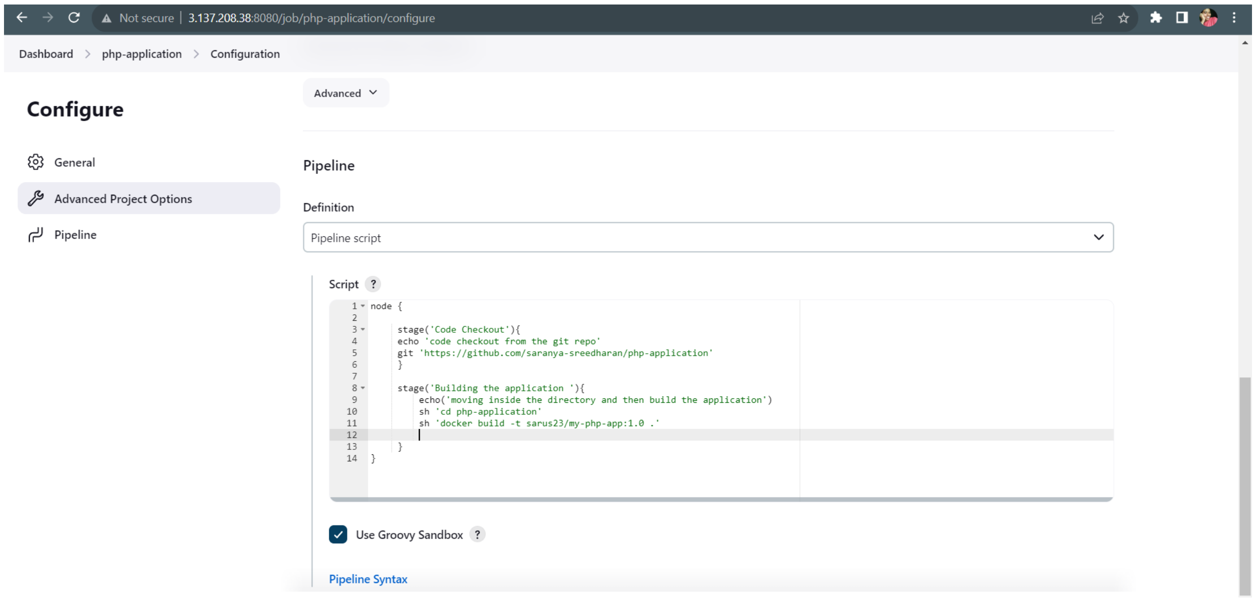
Stage View

	Code Checkout
Average stage times: (Average full run time: ~5s)	3s
#1 Oct 07 14:02 No Changes	3s

Permalinks

[14]

Install docker in jenkins(Manage jenkins- plugins-docker) then setup the docker in the tools.


Not secure | 3.137.208.38:8080/job/php-application/configure
Dashboard > php-application > Configuration
Configure
General
Advanced Project Options
Pipeline
Advanced

Pipeline

Definition: Pipeline script

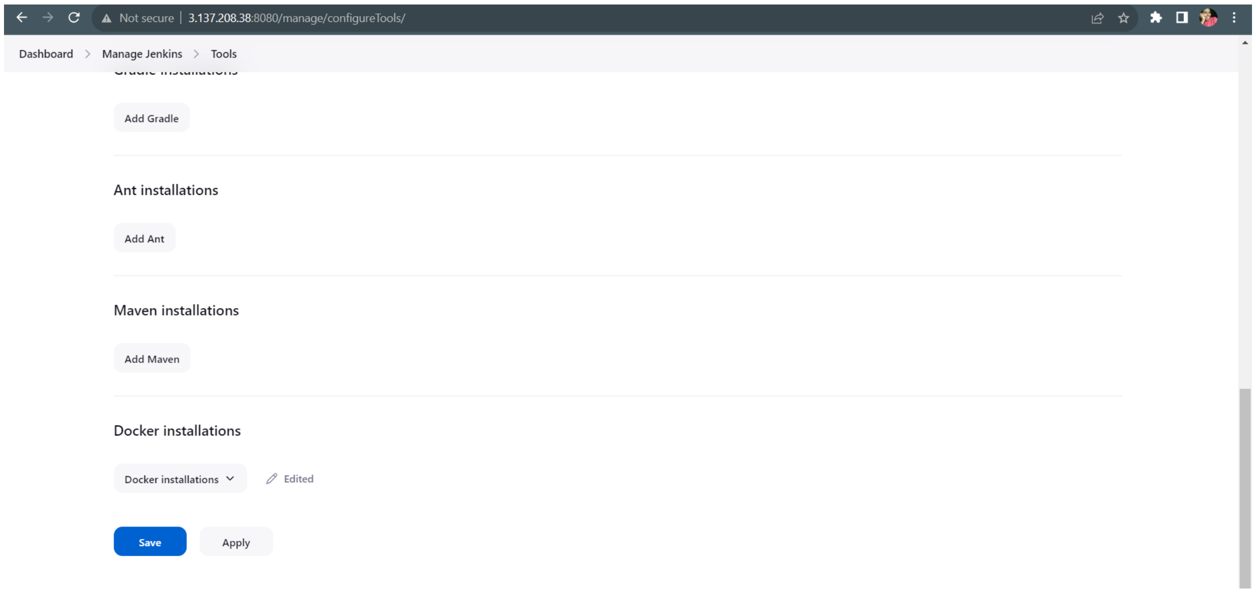
Script

```
node {

    stage('Code Checkout'){
    echo 'code checkout from the git repo'
    git 'https://github.com/saranya-sreedharan/php-application'
    }

    stage('Building the application '){
        echo('moving inside the directory and then build the application')
        sh 'cd php-application'
        sh 'docker build -t sarus23/my-php-app:1.0 .'

    }
}
```

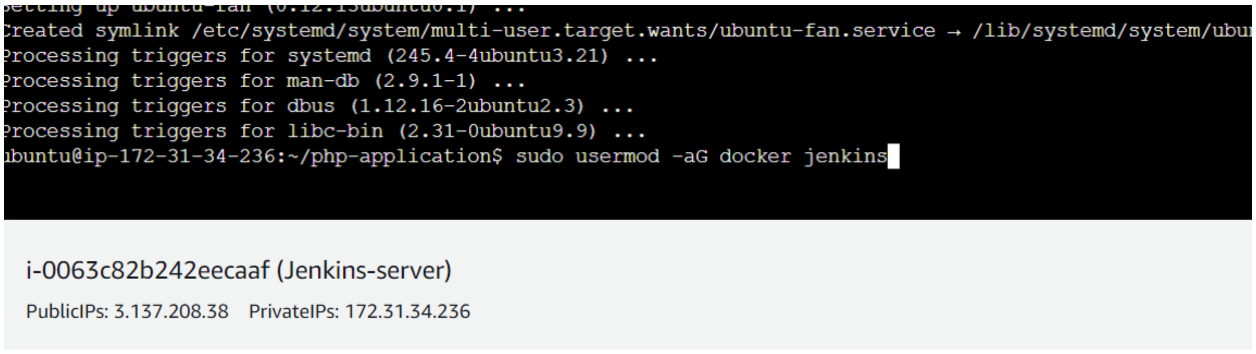
[✓] Use Groovy Sandbox

Pipeline Syntax

[14] Saranya Sreedharan



Give the permission to jenkins to run the docker commands
sudo usermod -aG docker jenkins

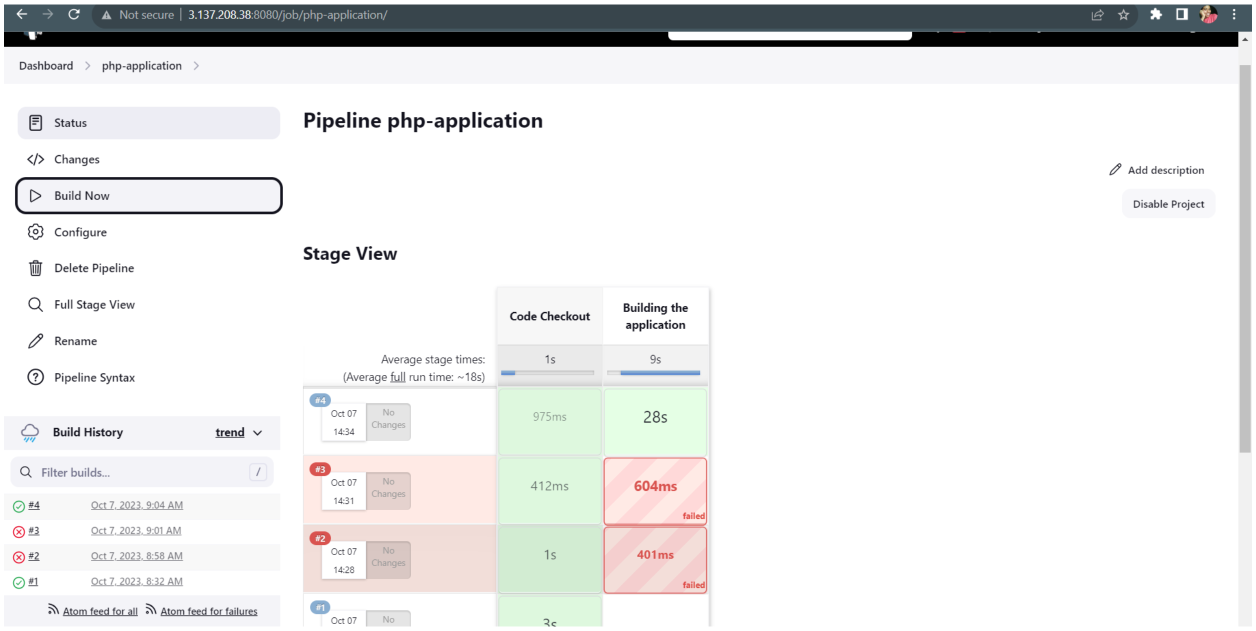


```
Setting up ubuntu-fan (0.12.13ubuntu0.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubu
Processing triggers for systemd (245.4-4ubuntu3.21) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.3) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
ubuntu@ip-172-31-34-236:~/php-application$ sudo usermod -aG docker jenkins
```

i-0063c82b242eecaaf (Jenkins-server)
PublicIPs: 3.137.208.38 PrivateIPs: 172.31.34.236

15

15 Saranya Sreedharan



Not secure | 3.137.208.38:8080/job/php-application/

Dashboard > php-application >

Pipeline php-application

Stage View

		Code Checkout	Building the application
Average stage times: (Average full run time: ~18s)		1s	9s
#4 Oct 07 14:34	No Changes	975ms	28s
#3 Oct 07 14:31	No Changes	412ms	604ms failed
#2 Oct 07 14:28	No Changes	1s	401ms failed
#1 Oct 07	No	3s	

Build History trend

Filter builds...

#4	Oct 7, 2023, 9:04 AM
#3	Oct 7, 2023, 9:01 AM
#2	Oct 7, 2023, 8:58 AM
#1	Oct 7, 2023, 8:32 AM

Atom feed for all Atom feed for failures

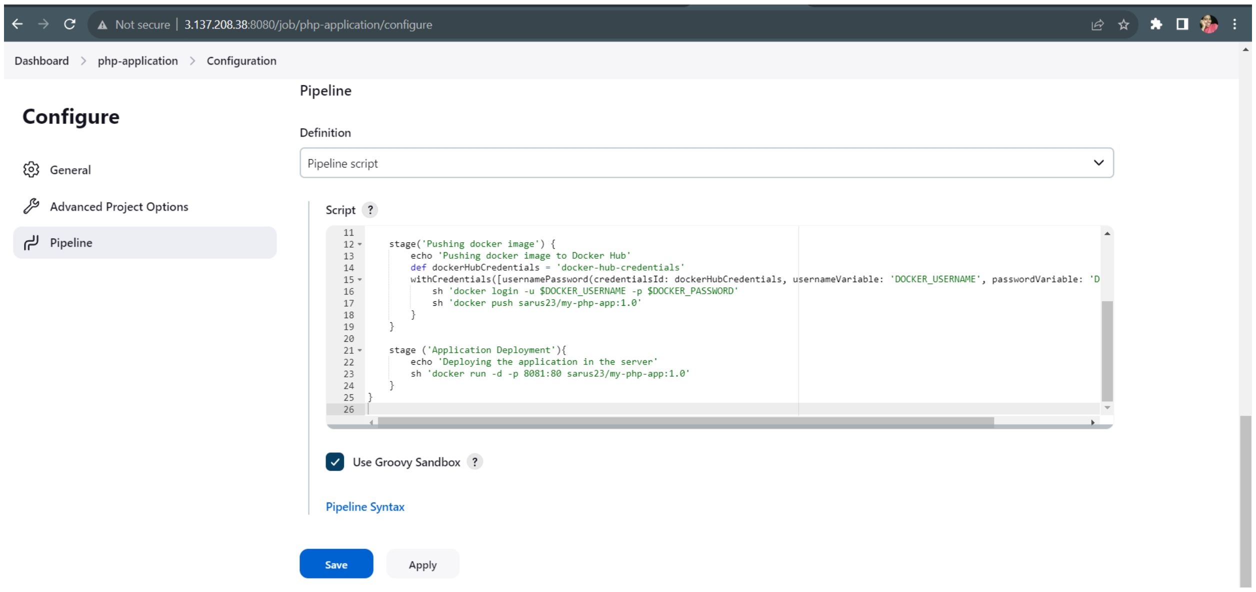
Then push the image to docker hub. Store the username and password in the global credentials then use this credentials in the pipeline for push the image in to docker hub.

```
stage('Pushing docker image') {
        echo 'Pushing docker image to Docker Hub'
        def dockerHubCredentials = 'docker-hub-credentials'
        withCredentials([usernamePassword(credentialsId: dockerHubCredentials,
usernameVariable: 'DOCKER_USERNAME', passwordVariable: 'DOCKER_PASSWORD')]) {
            sh 'docker login -u $DOCKER_USERNAME -p $DOCKER_PASSWORD'
            sh 'docker push sarus23/my-php-app:1.0'
        }
    }
```

[16]Then deploy the application. Run the docker image in the pipeline as the next stage to deploy our application.

```
stage ('Application Deployment'){
        echo 'Deploying the application in the server'
        sh 'docker run -d -p 8081:80 sarus23/my-php-app:1.0'
    }
```

[16] Saranya Sreedharan


Not secure | 3.137.208.38:8080/job/php-application/configure

Dashboard > php-application > Configuration

Configure

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

Script

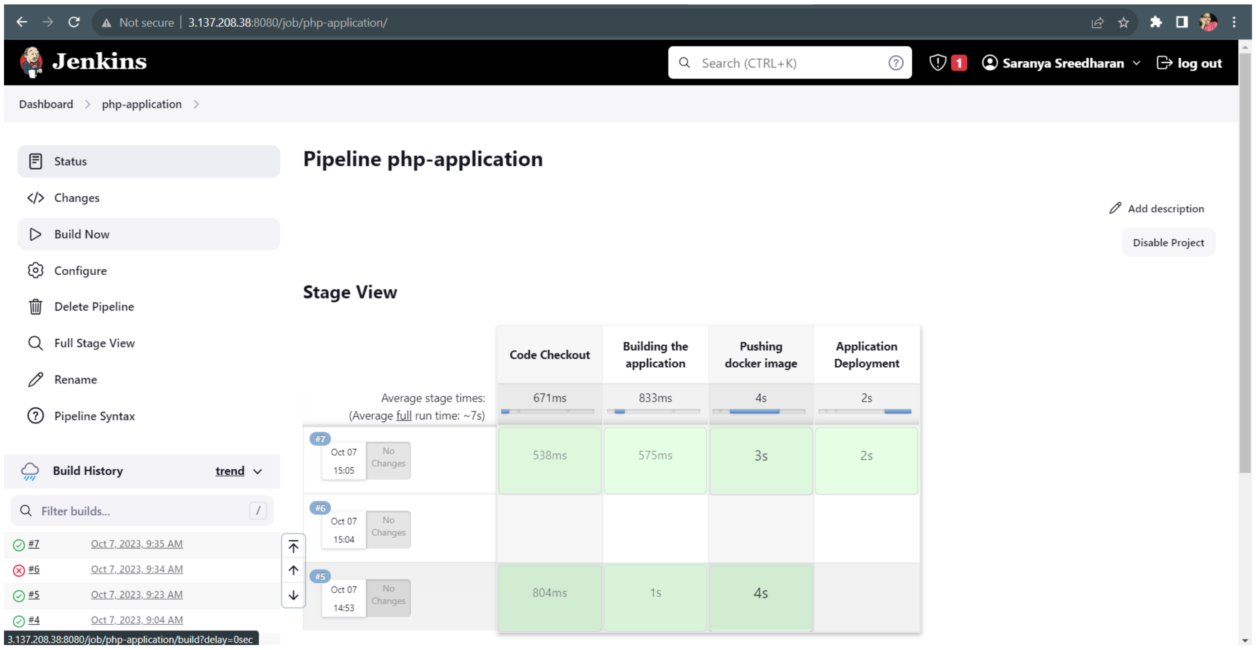
```
    stage('Pushing docker image') {
        echo 'Pushing docker image to Docker Hub'
        def dockerHubCredentials = 'docker-hub-credentials'
        withCredentials([usernamePassword(credentialsId: dockerHubCredentials, usernameVariable: 'DOCKER_USERNAME', passwordVariable: 'D
            sh 'docker login -u $DOCKER_USERNAME -p $DOCKER_PASSWORD'
            sh 'docker push sarus23/my-php-app:1.0'
        }
    }

    stage ('Application Deployment'){
        echo 'Deploying the application in the server'
        sh 'docker run -d -p 8081:80 sarus23/my-php-app:1.0'
    }
}

```

[✓] Use Groovy Sandbox

Pipeline Syntax

Save Apply

So successfully we deployed our application. Verify with the jenkins server ip address to access the application and user input page(samplepage.php)


Not secure | 3.137.208.38:8080/job/php-application/

Jenkins

Search (CTRL+K)

Saranya Sreedharan log out

Dashboard > php-application >

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Build History trend

Filter builds...

✓ #7	Oct 7, 2023, 9:35 AM
✗ #6	Oct 7, 2023, 9:34 AM
✓ #5	Oct 7, 2023, 9:23 AM
✓ #4	Oct 7, 2023, 9:04 AM

3.137.208.38:8080/job/php-application/build?delay=0sec

Pipeline php-application

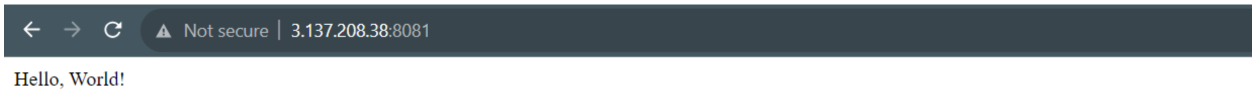
Add description

Disable Project

Stage View

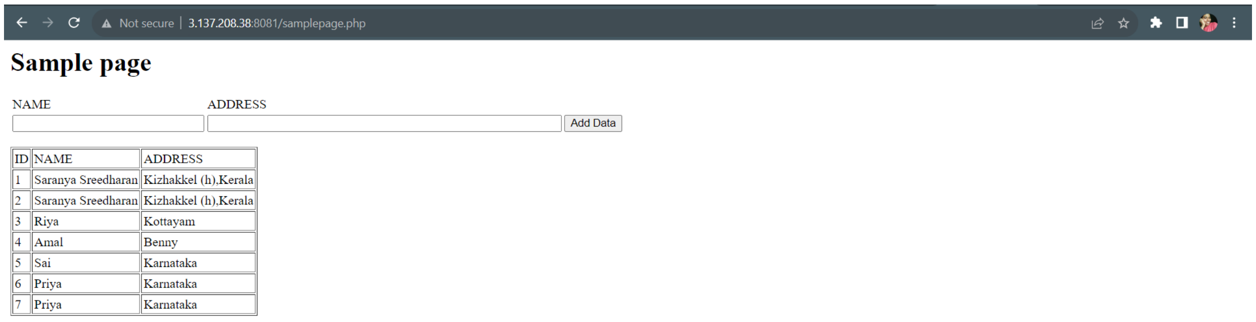
	Code Checkout	Building the application	Pushing docker image	Application Deployment
Average stage times: (Average full run time: ~7s)	671ms	833ms	4s	2s
#7 Oct 07 15:05 No Changes	538ms	575ms	3s	2s
#6 Oct 07 15:04 No Changes				
#5 Oct 07 14:53 No Changes	804ms	1s	4s	

17



Not secure | 3.137.208.38:8081

Hello, World!



Not secure | 3.137.208.38:8081/samplepage.php

Sample page

NAME ADDRESS

Add Data

ID	NAME	ADDRESS
1	Saranya Sreedharan	Kizhakkel (h),Kerala
2	Saranya Sreedharan	Kizhakkel (h),Kerala
3	Riya	Kottayam
4	Amal	Benny
5	Sai	Karnataka
6	Priya	Karnataka
7	Priya	Karnataka

Successfully Dockerized a PHP application and deployed it on an AWS EC2 instance conn[18]ected to an AWS RDS MySQL database!.

[18] Saranya Sreedharan