# Aws(cloud)-monolithic architecture

## Introduction:

Welcome to this report detailing the meticulous setup of MySQL and WordPress on an Ubuntu EC2 instance. This document chronicles the systematic installation and configuration process aimed at establishing a resilient database management system and a versatile content management system for hosting websites or web applications.

## The steps should be followed to create an instance

Log in to the AWS Management Console using your credentials. Ensure that you are logged in securely, and take necessary precautions to protect your login information.

Once logged in, navigate to the EC2 service dashboard. This is where you can manage your virtual servers, also known as instances.

Click on the "Launch Instance" button to begin the instance creation process. You'll be guided through a series of steps to configure your instance.

Select an AMI that suits your requirements. This is essentially the operating system and software stack that will be pre-installed on your instance.

Choose the instance type based on your workload. Instances come in various sizes with different CPU, memory, storage, and networking capacities.

Configure additional settings such as instance details, network settings, and storage options. You can customize parameters like instance name, network settings (VPC, subnet, IP addressing), and storage volumes.

( Optionally) you can add tags to your instance for better organization and management. Tags are key-value pairs that help you identify resources easily.

Create or select an existing security group. This acts as a virtual firewall for your instance, controlling inbound and outbound traffic.

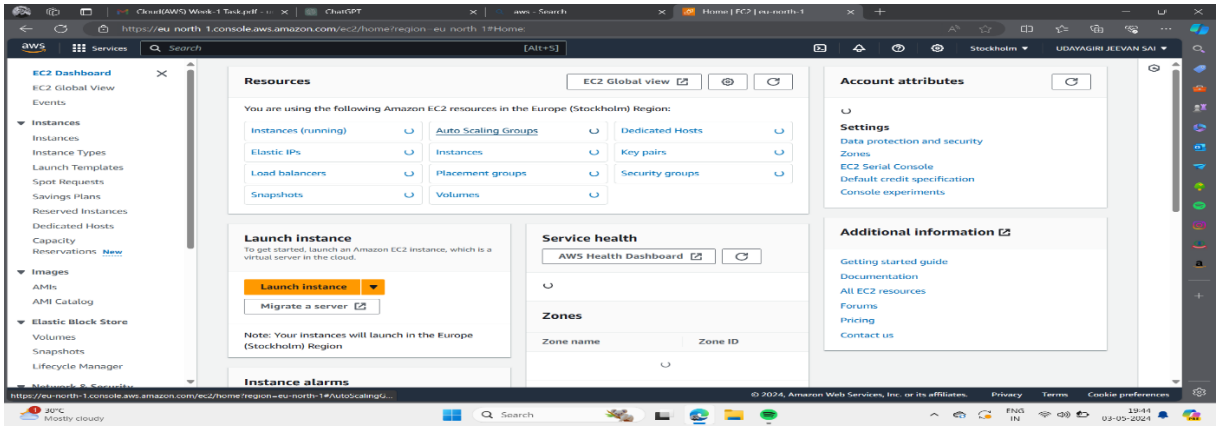
Review your instance configuration to ensure everything is set up as desired. Once confirmed, click "Launch" to initiate the instance creation process.

Select an existing key pair or create a new one. This key pair will be used to securely connect to your instance via SSH (for Linux instances) or RDP (for Windows instances).

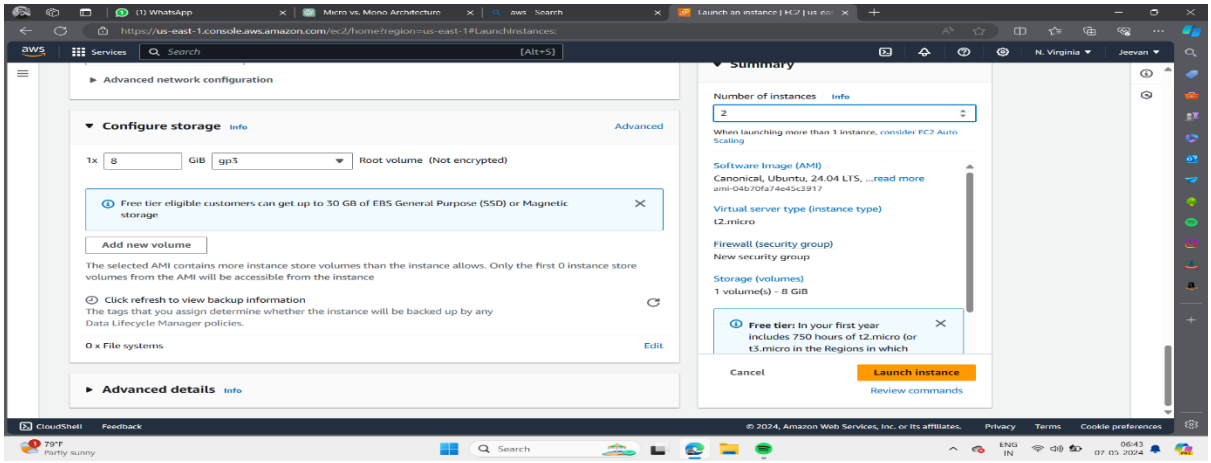
After selecting the key pair, click "Launch Instances" to finalize the process. Your instance will now be provisioned and launched.

Once the instance is running, you can access it via SSH (for Linux) or RDP (for Windows) using the appropriate credentials and the private key associated with your key pair eating an instance on a cloud platform like AWS can be done as follows:

**THE UPCOMING SHOTS SHOWS THE STEPS TO BE FOLLOWED**



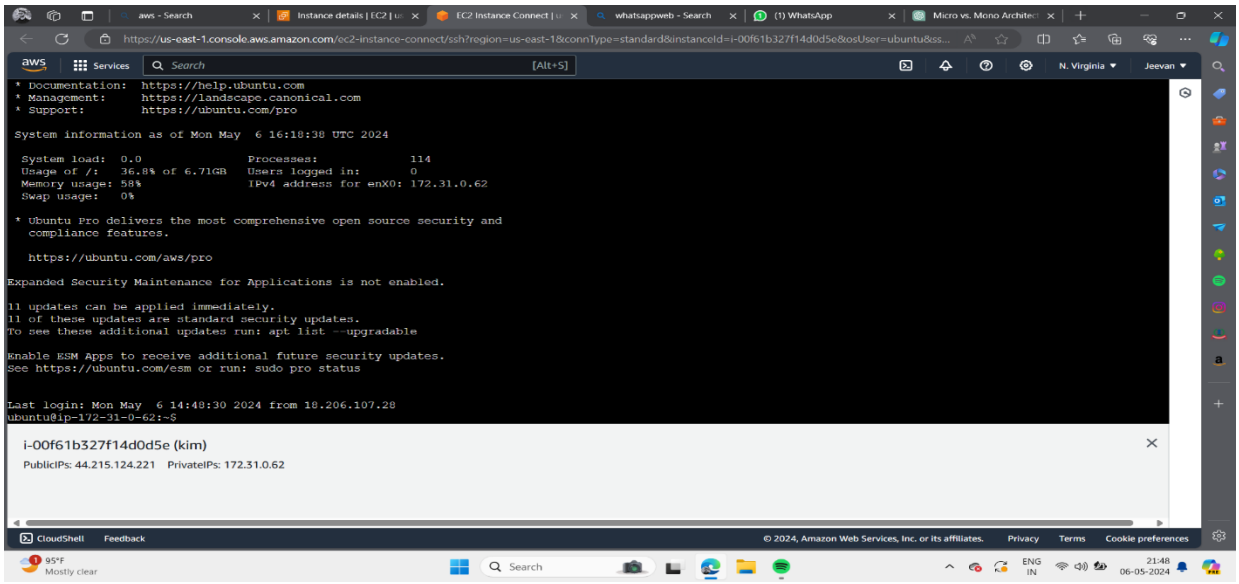

Select all requirements




Launch you instance as shown in above screenshot

Next

Later it will display like this which is connected to cmd prompt of AWS



# Installation and Configuration of MySQL:

Installation of MySQL Server and Client: Executed commands to update package index and install MySQL server and client, ensuring the latest version is obtained.

Run the mysql_ secure_installation script to enhance security by setting a root password, removing anonymous users, disallowing root login remotely, and removing the test database and access to it.

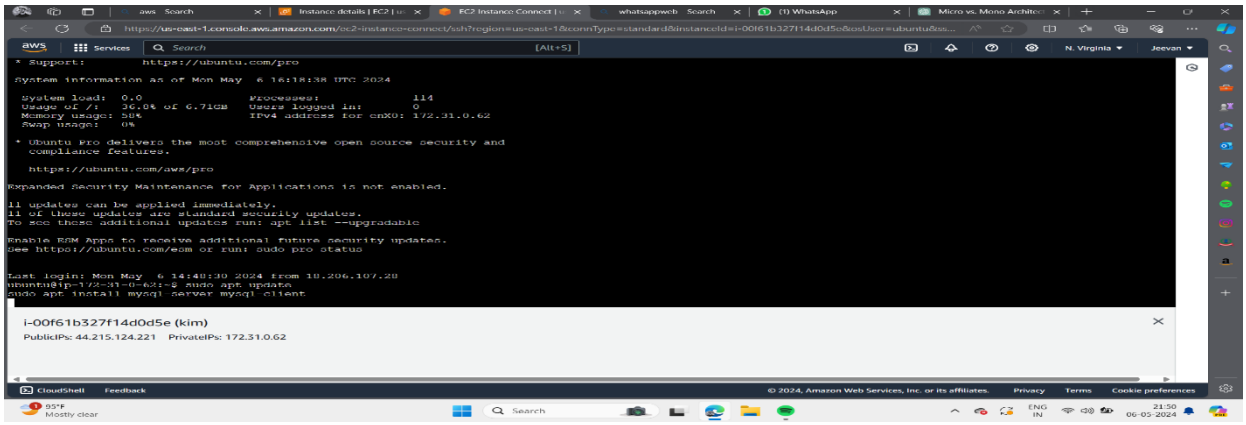
Logged into MySQL shell, created a database named wordpress', created a MySQL user with privileges on the 'wordpress' database, and verified the successful creation of the database and user.

The installation and configuration process were guided by prior knowledge of Linux server administration and MySQL management. Additionally, the official documentation of MySQL provided valuable insights and best practices during the setup process.

## Use the command

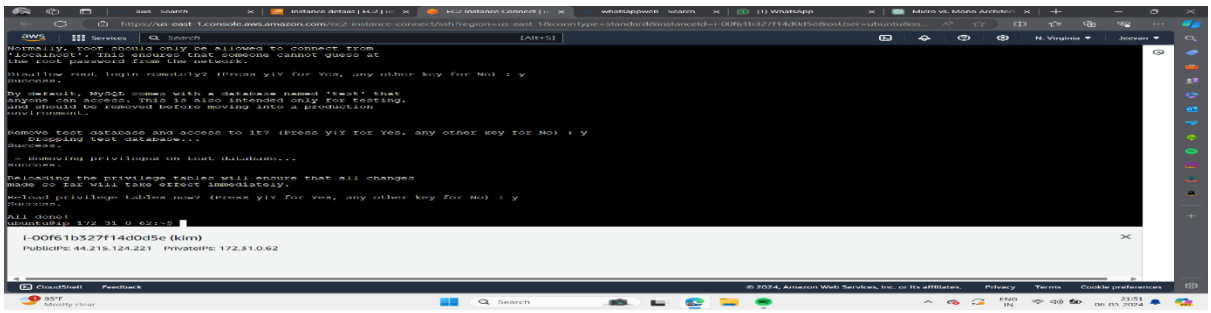
sudo apt update

sudo apt install mysql-server mysql-client  (to install)




## Secure mysql installation

sudo mysql_secure_installation



## Follow these steps for secure installation

Set root password

Remove anonymous users

Disallow root login remotely

Remove test database and access to it

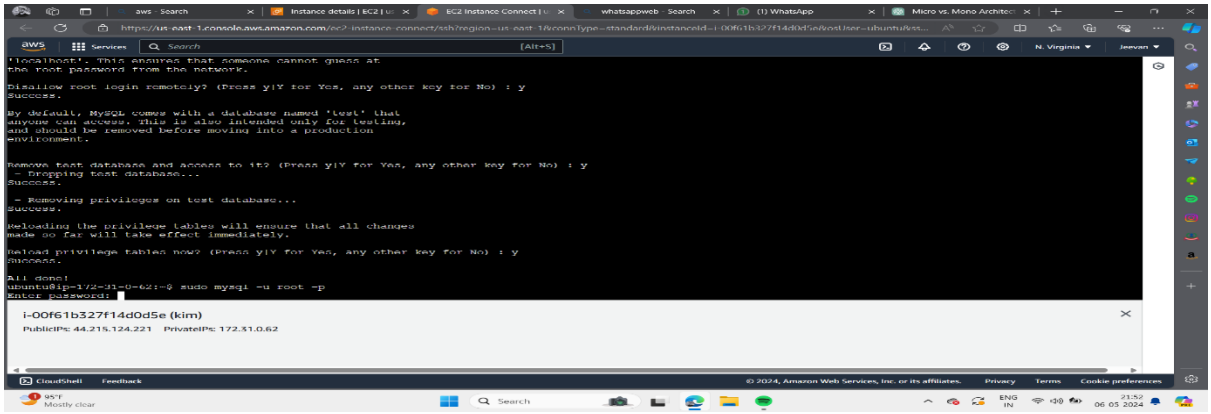
Reload privilege tables now

# Create a MySQL Database and User for WordPress

## Log into the MySQL shell as the root user

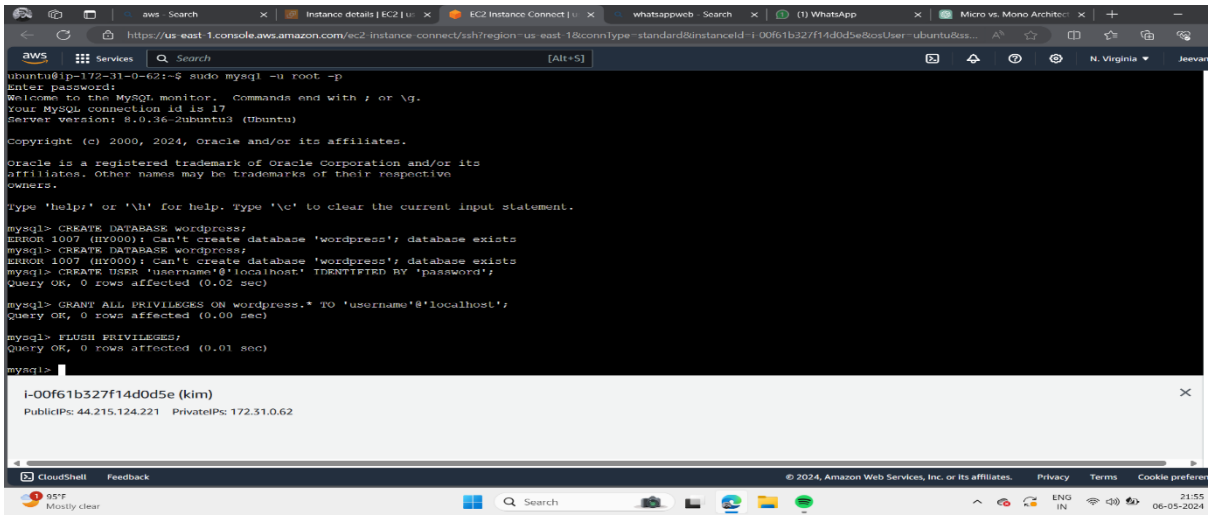
sudo mysql -u root -p

Enter the root password when prompted




Once logged in create a DB

CREATE DATABASE wordpress;



Next follow the steps

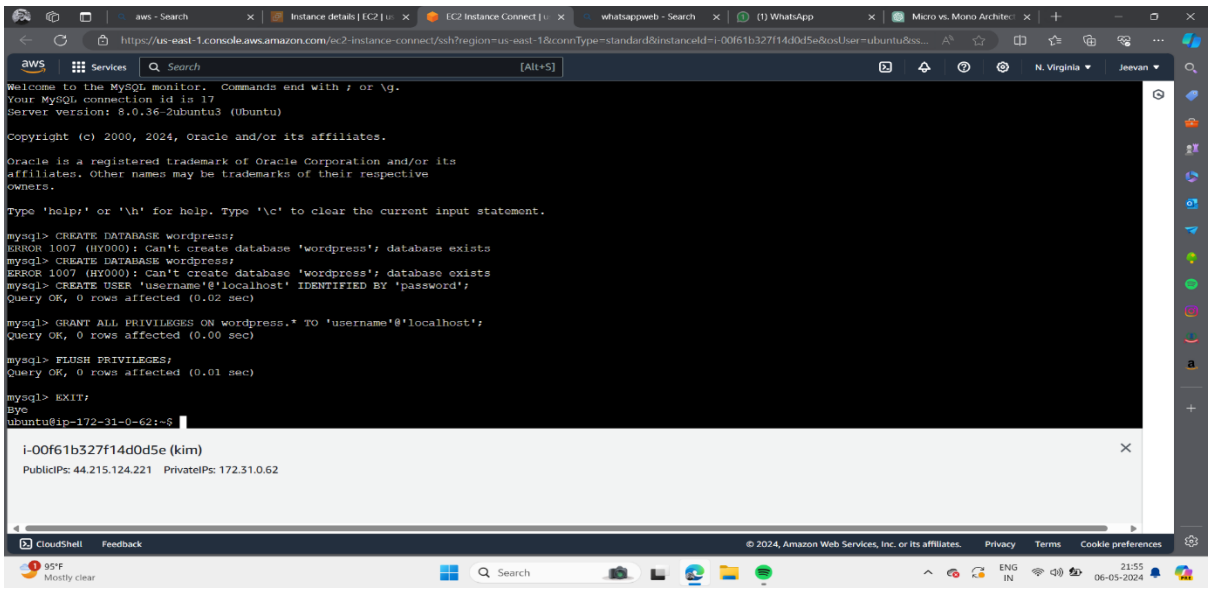
CREATE USER 'username'@'localhost' IDENTIFIED BY 'password';

GRANT ALL PRIVILEGES ON wordpress.* TO 'username'@'localhost';

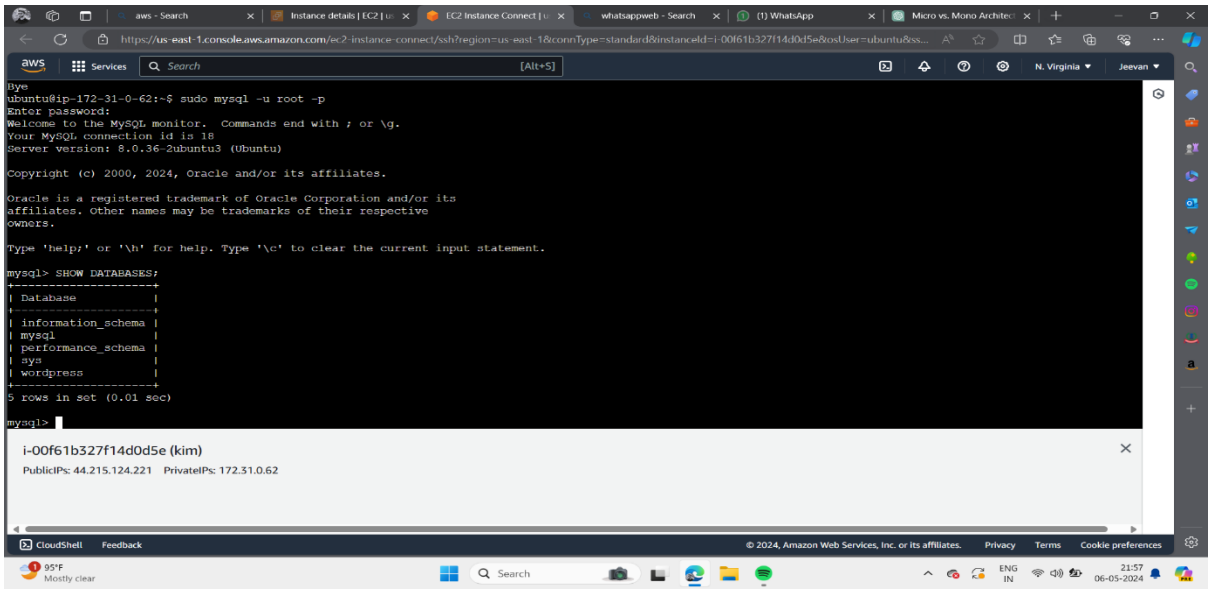
FLUSH PRIVILEGES;

After this exit command should be used

EXIT;



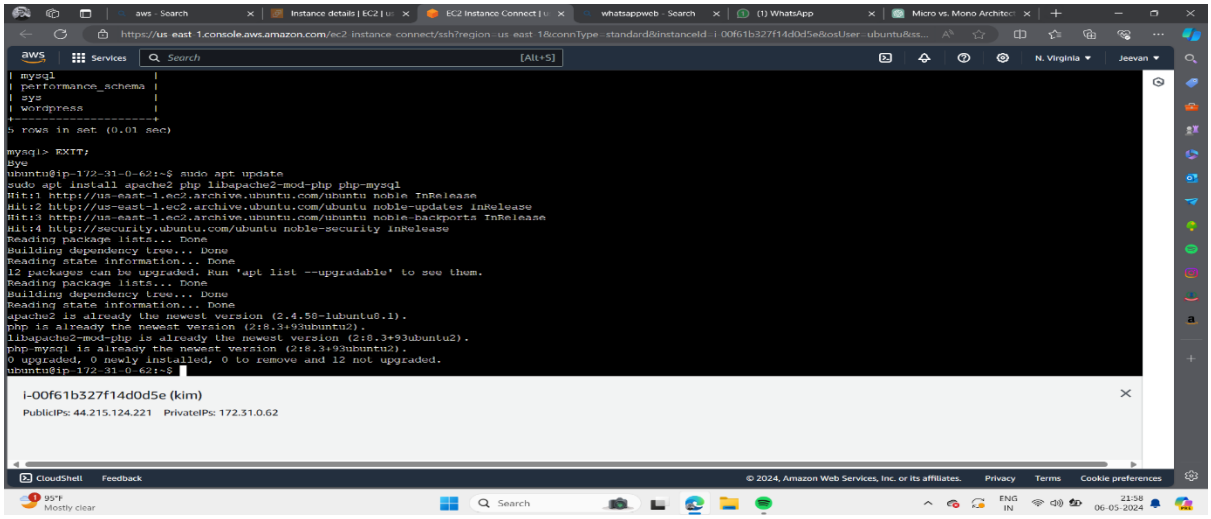

Show databases;



## Install Apache Web Server, PHP, and Required Extensions

sudo apt update

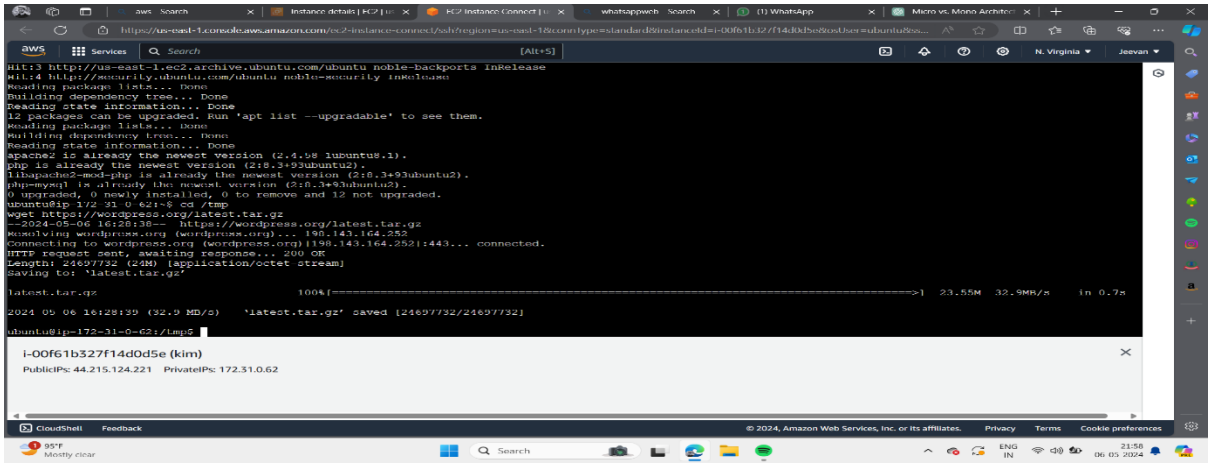
sudo apt install apache2 php libapache2-mod-php php-mysql




## Download and Configure WordPress

cd /tmp

wget https://wordpress.org/latest.tar.gz




## Extract the downloaded WordPress archive:

tar -xvzf latest.tar.gz

### Move the extracted WordPress files to the Apache document root directory (/var/www/html):

sudo mv wordpress/* /var/www/html/

### Set appropriate permissions on the WordPress directory:

sudo chown -R www-data:www-data /var/www/html/ sudo chmod -R 755 /var/www/html/

# Configure WordPress to Use MySQL Database

## Rename:

cd /var/www/html

sudo mv wp-config-sample.php wp-config.php

## Edit:

sudo nano wp-config.php

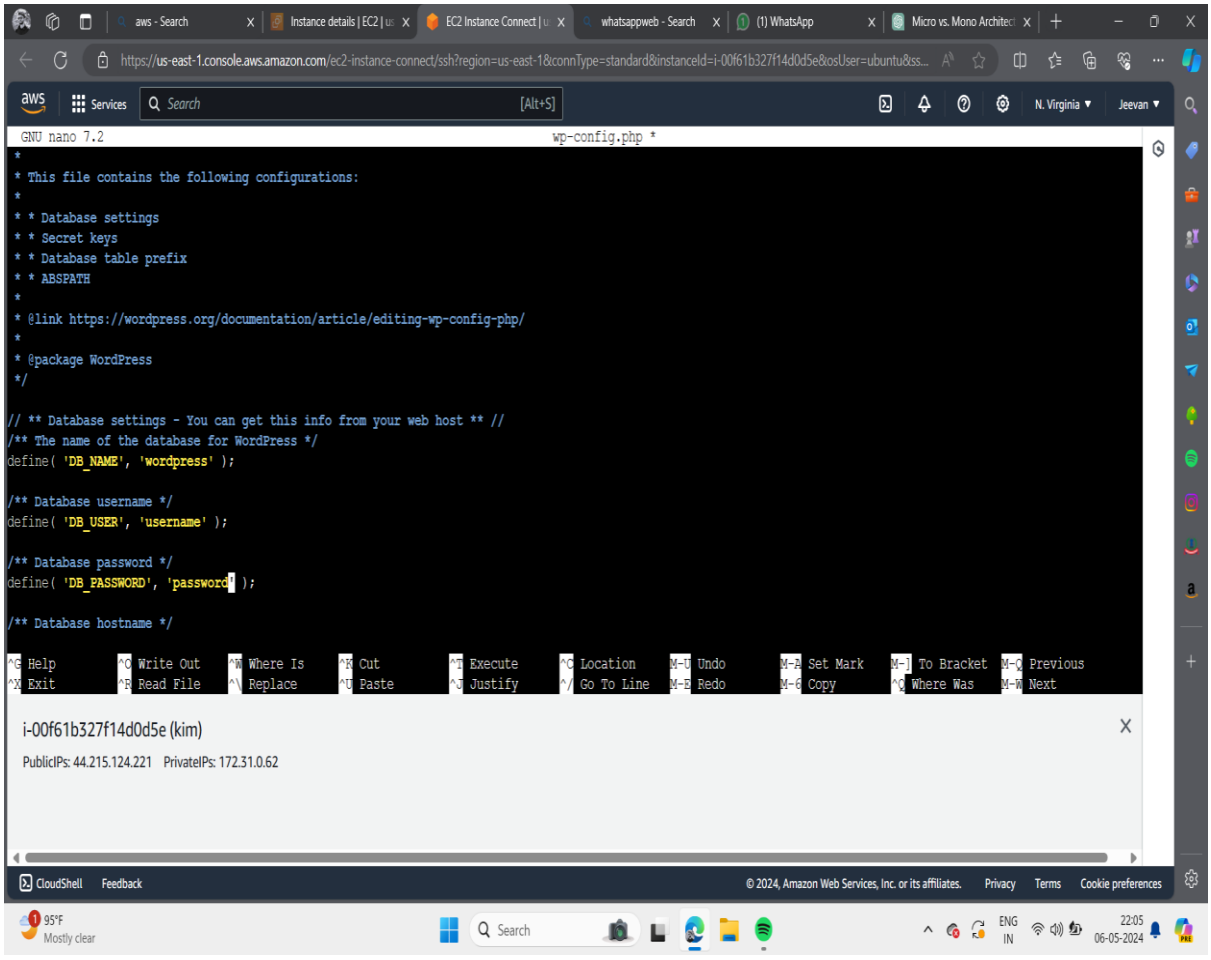
## Update the following lines with your MySQL database information:

define('DB_NAME', 'wordpress');

define('DB_USER', 'username');

define('DB_PASSWORD', 'password');

define('DB_HOST', 'localhost');



# Authentication Keys And Salts

define( 'AUTH_KEY',         'put your unique phrase here' );

define( 'SECURE_AUTH_KEY',  'put your unique phrase here' );

define( 'LOGGED_IN_KEY',    'put your unique phrase here' );

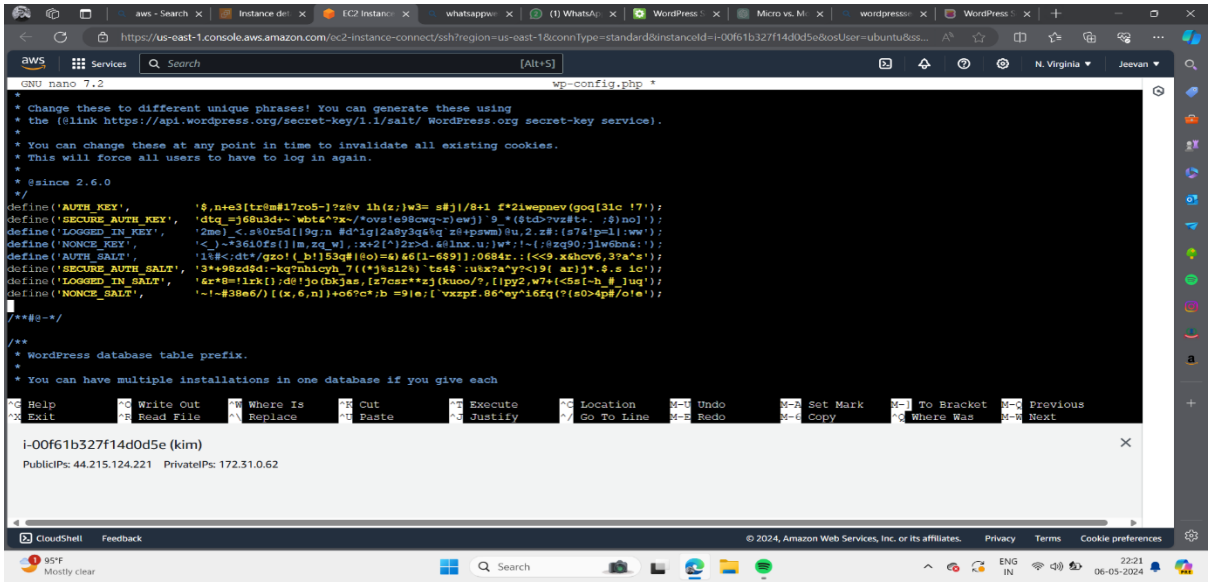
define( 'NONCE_KEY',        'put your unique phrase here' );

define( 'AUTH_SALT',        'put your unique phrase here' );

define( 'SECURE_AUTH_SALT', 'put your unique phrase here' );

define( 'LOGGED_IN_SALT',   'put your unique phrase here' );

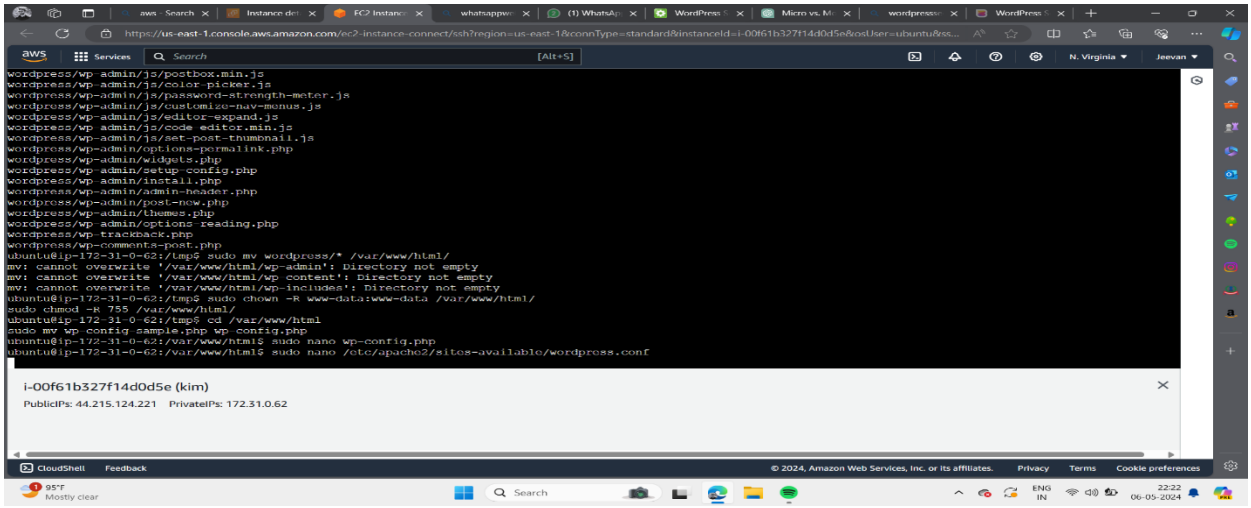
define( 'NONCE_SALT',       'put your unique phrase here' );




# Set Up Virtual Hosts to Serve WordPress

Create a new virtual host configuration file for WordPress:

sudo nano /etc/apache2/sites-available/wordpress.conf



# Paste the following configuration into the file:

<VirtualHost *:80>

ServerAdmin admin@example.com

DocumentRoot /var/www/html

ServerName your_domain.com (give your domain address here)

ServerAlias www.your_domain.com (give your domain address here)

<Directory /var/www/html>

Options FollowSymLinks

AllowOverride All

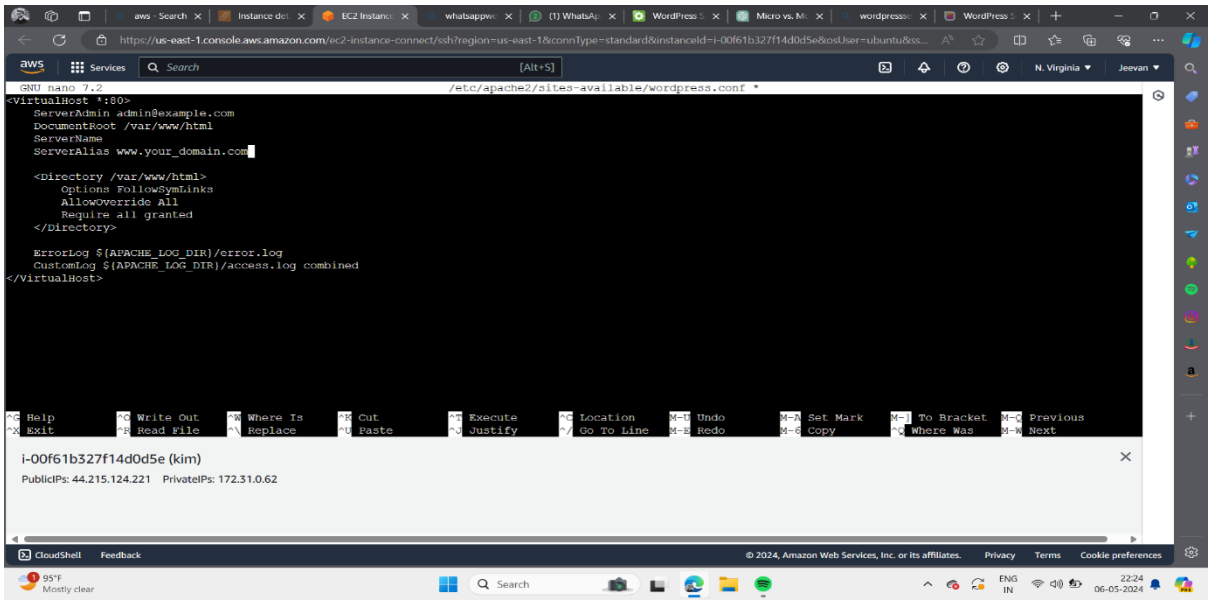
Require all granted

</Directory>

ErrorLog ${APACHE_LOG_DIR}/error.log

CustomLog ${APACHE_LOG_DIR}/access.log combined

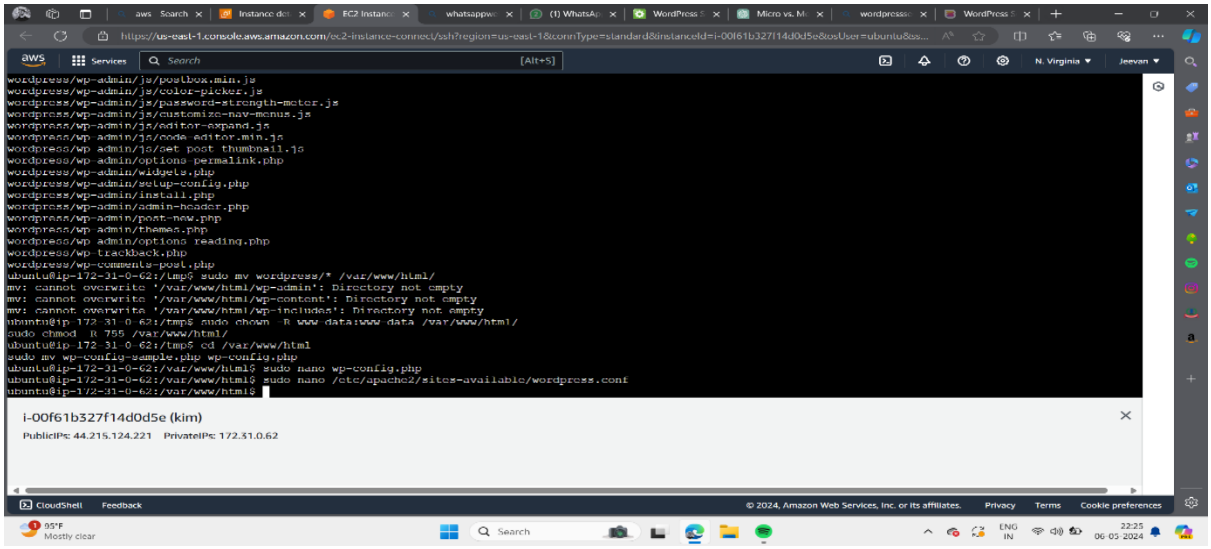
</VirtualHost>




# Enable the new virtual host configuration and disable the default one:

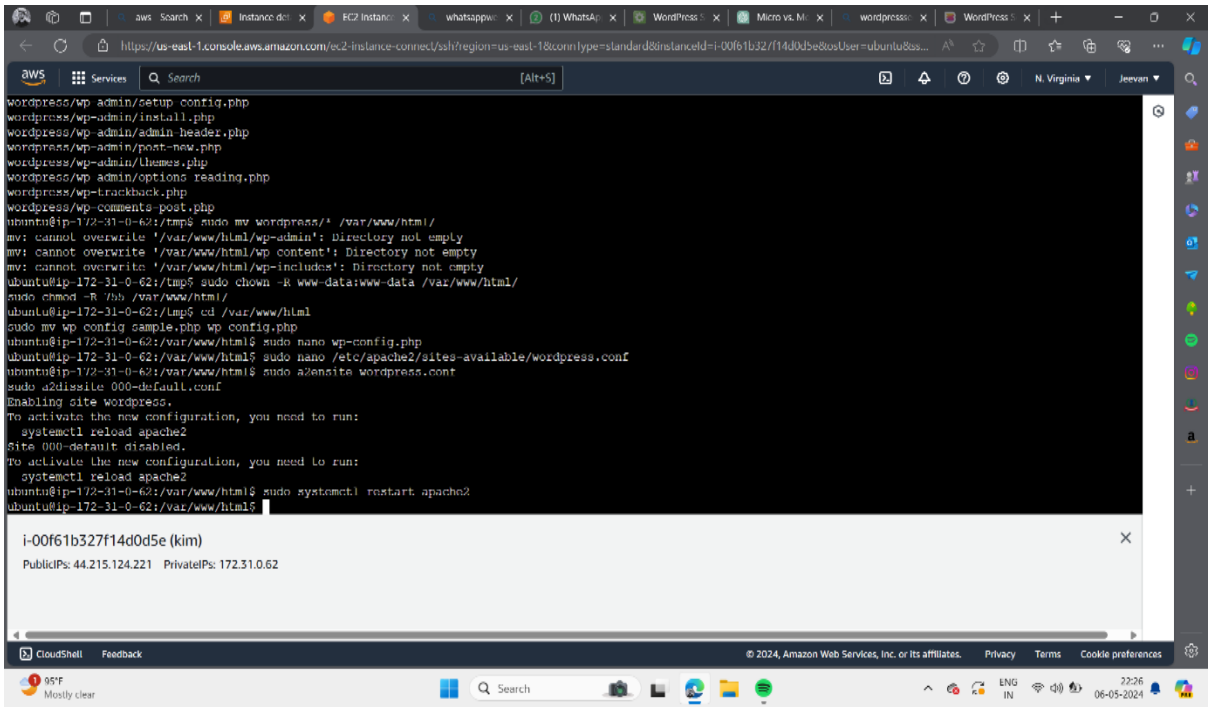
sudo a2ensite wordpress.conf

sudo a2dissite 000-default.conf




# Restart Apache for the changes to take effect:

sudo systemctl restart apache2



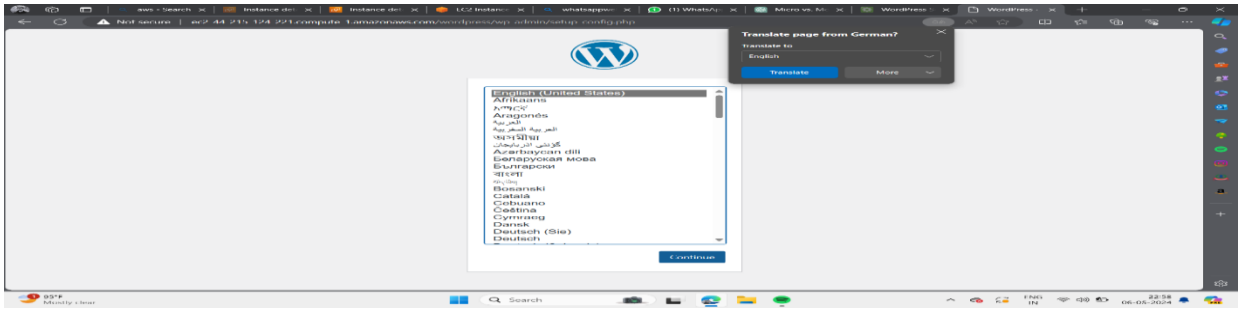

**Finally paste your Dns or instead paste IP address of your created instance**

**In my case ip address is : 44.215.124.221\wordpress**

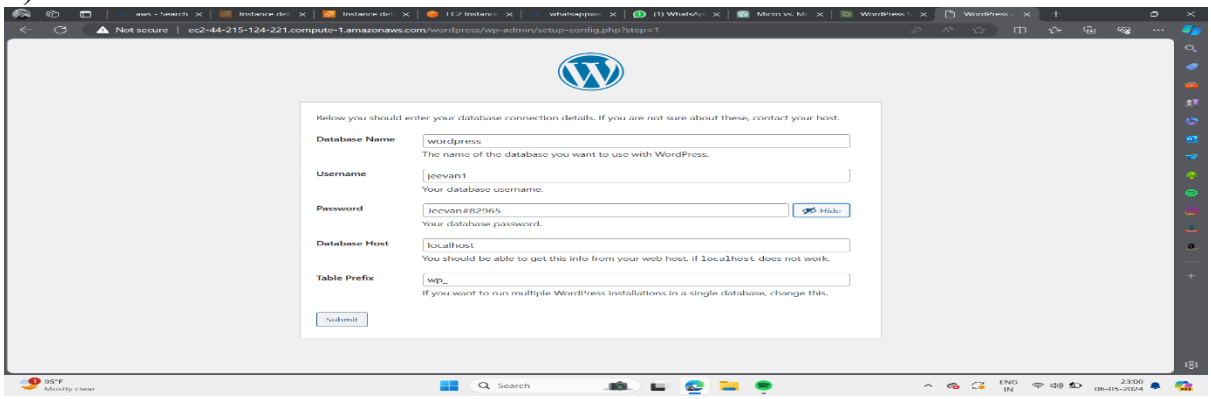
**In my case Dns is: ec2-44-215-124-221.compute-1.amazonaws.com\wordpress\wp-admin**

**Further the wordpress page will get displayed and user can allow to do further steps :** In my case I will be adding  images what I have done

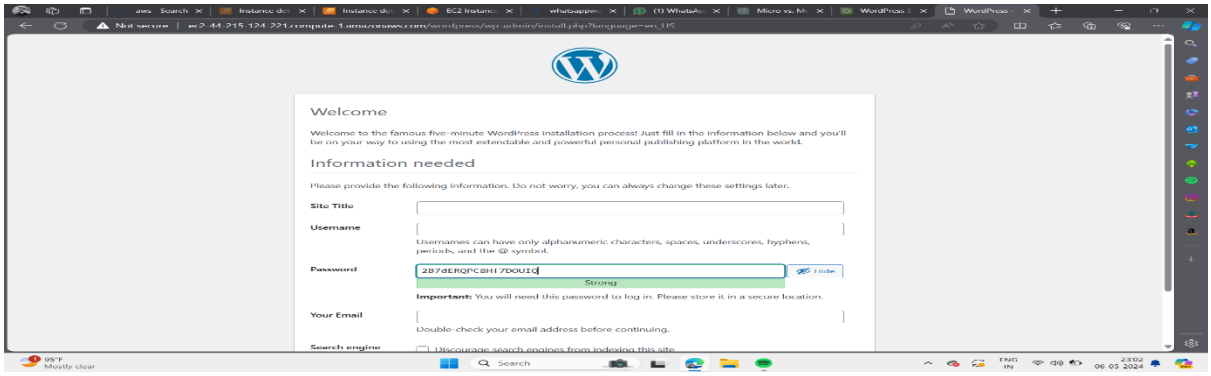
1)



2)



3)



4)finally the page saying welcome to wordpress will be displayed

Here is the deployed page as welcome page in wordpress

# Microservices

**Firstly we should create two instances that is 1 for wordpress and another for mysql**

## The steps should be followed to create an instance

Log in to the AWS Management Console using your credentials. Ensure that you are logged in securely, and take necessary precautions to protect your login information.

Once logged in, navigate to the EC2 service dashboard. This is where you can manage your virtual servers, also known as instances.

Click on the "Launch Instance" button to begin the instance creation process. You'll be guided through a series of steps to configure your instance.

Select an AMI that suits your requirements. This is essentially the operating system and software stack that will be pre-installed on your instance.

Choose the instance type based on your workload. Instances come in various sizes with different CPU, memory, storage, and networking capacities.

Configure additional settings such as instance details, network settings, and storage options. You can customize parameters like instance name, network settings (VPC, subnet, IP addressing), and storage volumes

Create or select an existing security group. This acts as a virtual firewall for your instance, controlling inbound and outbound traffic.

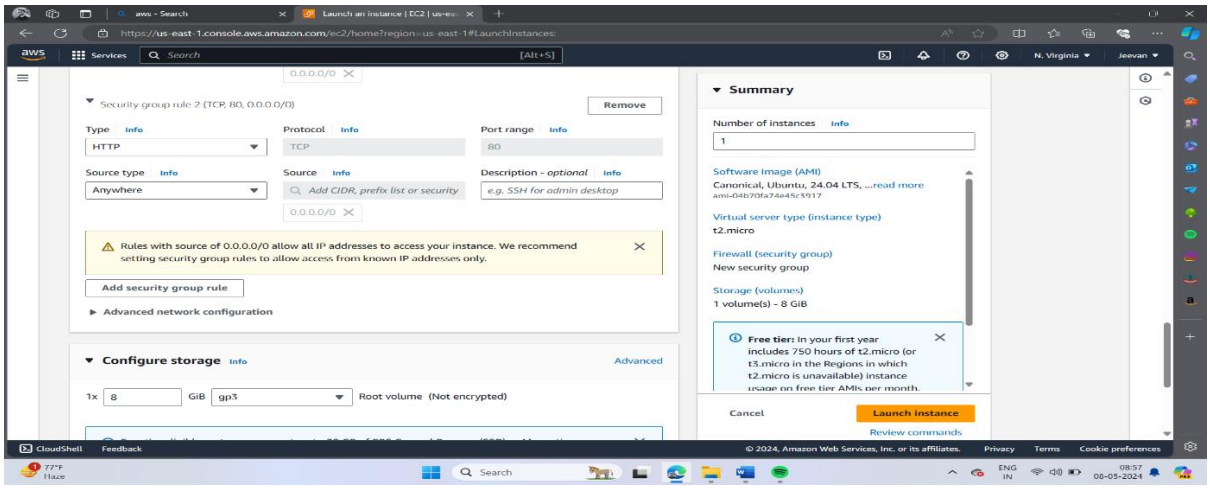
Review your instance configuration to ensure everything is set up as desired. Once confirmed, click "Launch" to initiate the instance creation process.

Select an existing key pair or create a new one. This key pair will be used to securely connect to your instance via SSH (for Linux instances) or RDP (for Windows instances).

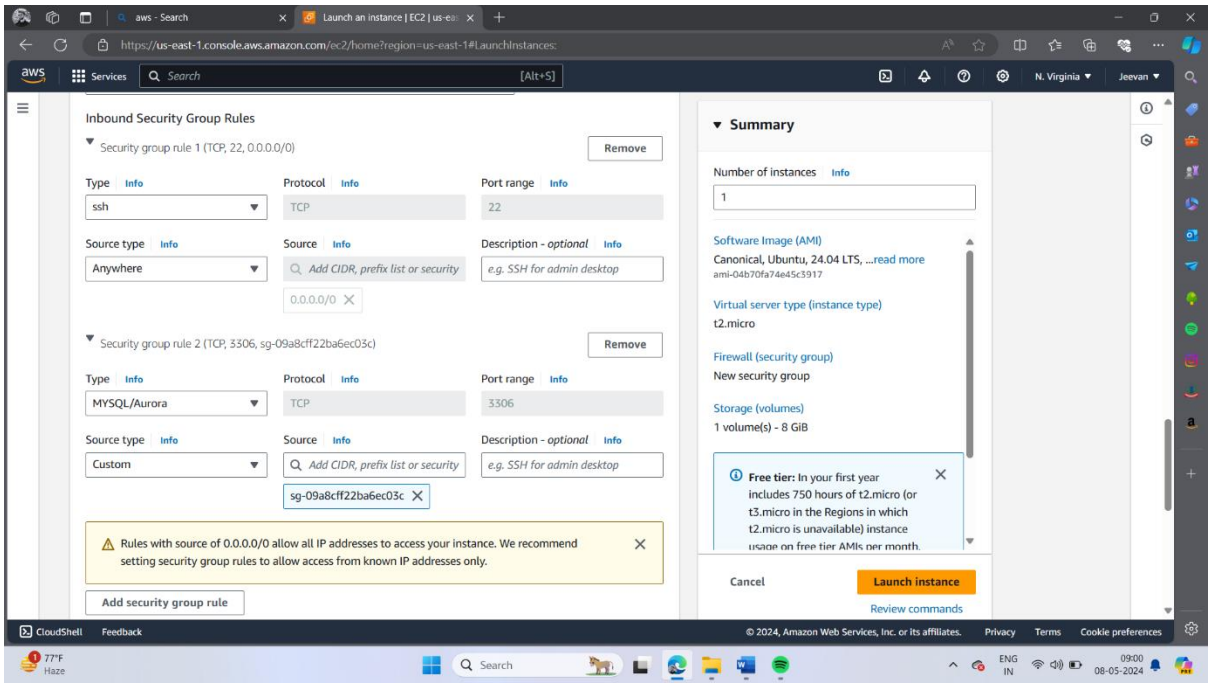
After selecting the key pair, click "Launch Instances" to finalize the process. Your instance will now be provisioned and launched.

Once the instance is running, you can access it via SSH (for Linux) or RDP (for Windows) using the appropriate credentials and the private key associated with your key pair eating an instance on a cloud platform like AWS can be done as follows:

### 1 instance for wordpress



**2 The procedure follows same for creating other instance Mysql**

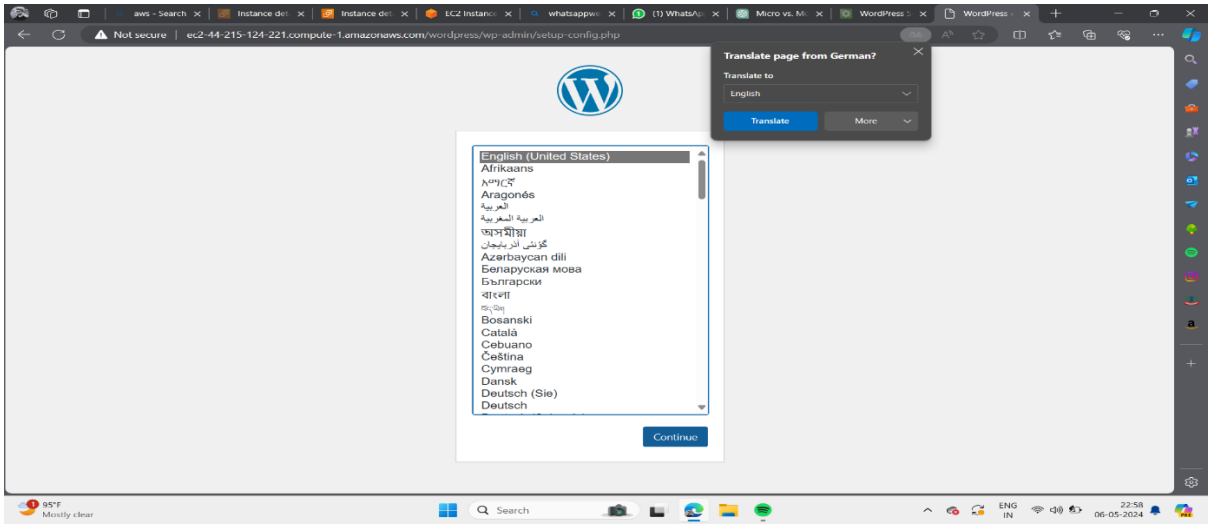



**After creating instances follow all the steps which are included in monolithic architecture and further provide connection's and connect servers with required configurations**

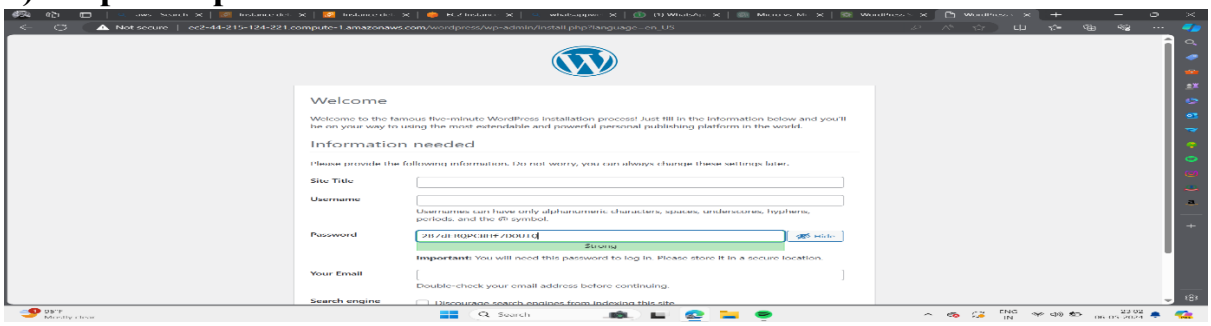
**Follow monolithic architecture for further activities after creating instances**

**Later paste your ip address in your web browser that will redirect to wordpress site and the wordpress site will open**

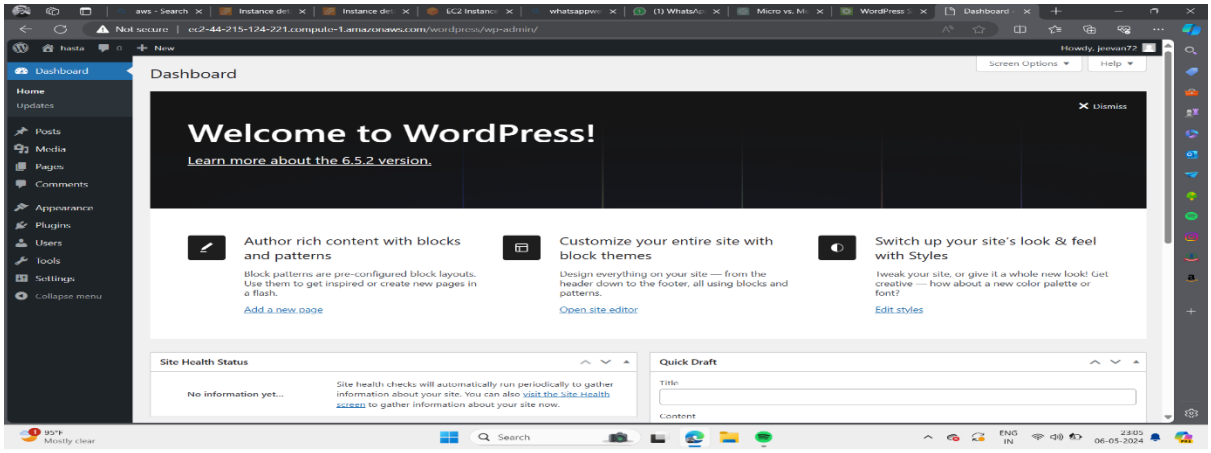
**1)**



## 2)setup wordpress




## 3)welcome to wordpress




**Thank you**