## 10/30/2022

# DevOps

Assignment 01

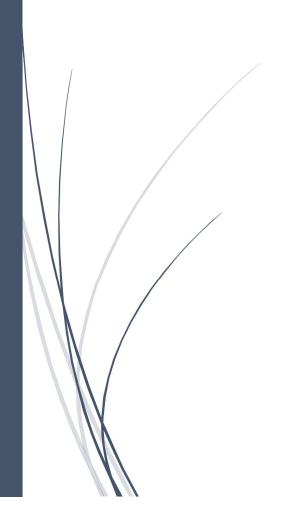
Submitted to:

Mr. Qasim Malik

Submitted by:

Khadija Tuz Zehra

FA19-BCS-033



### Contents

Create a new instance	2
Connect to VM with PuTTY	3
Installing Apache server	3
Install MySQL	4
Install phpMyAdmin database	5
Install PHP	6
Clone GitHub project	6
Deployment	8
URL	8
Part 2	9
Create Elastic Beanstalk instance	g
Connect to Elastic Beans	12
Create DB	13
Configure .env file	14
URL – part 02	15

### Create a new instance

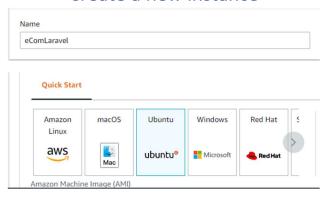


Figure 1: Name the instance, and specify the OS to be installed

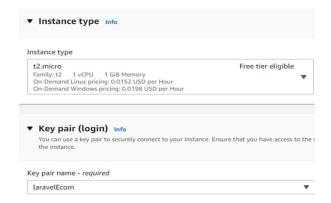


Figure 2: Generate a key, to connect to VM from local system

Define the following security inbound rules:

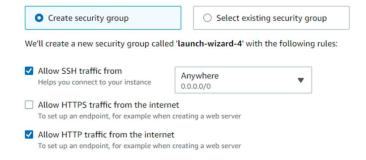


Figure 3: Allow SSH traffic (to connect to instance) and allow HTTP traffic

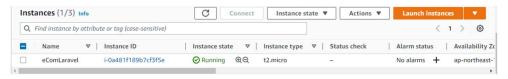
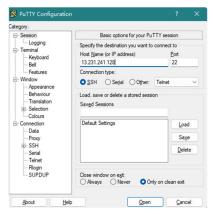


Figure 4: New instance created

### Connect to VM with PuTTY



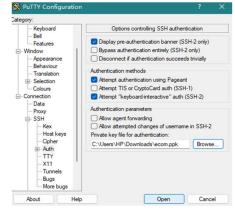


Figure 5: add Public IPv4 address

Figure 6: Add your .ppk file (key)

### Installing Apache server

Run the following commands beforehand,

- sudo apt-get update
- sudo apt-get upgrade

Installing zip unzip extension

```
ubuntu@ip-172-31-10-169:~$ sudo apt install zip unzip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
   libflashroml libftdil-2
Use 'sudo apt autoremove' to remove them.
```

Figure 7: Install zip unzip extension

### Install apache2

```
ubuntu@ip-172-31-10-169:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Figure 8: Install apache2 server

Check if apache2 is running:

```
ubuntu@ip-172-31-10-169:~$ sudo systemct1 status apache2
• apache2.service - The Apache HTTP Server
    Loaded: loaded (/lib/systemd/system/apache2.service; enabled; ve Active: active (running) since Sun 2022-10-30 l1:26:30 UTC; l6s
    Docs: https://httpd.apache.org/docs/2.4/
Main PID: 5196 (apache2)
```

Figure 9: apache2 server is installed and running

Access public IPv4 DNS address to confirm Apache server installation



Figure 10: The following page response confirms Apache server installation on ubuntu:

### Install MySQL

Run the following commands to install MySQL server:

```
ubuntu@ip-172-31-10-169:~$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Figure 11: install MySQL server

```
ubuntu@ip-172-31-2-179:~$ sudo mysql
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.31-0ubuntu0.22.04.1 (Ubuntu)
```

Figure 12: Switch to MySQL terminal to set password

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'm
ynewpassword';
Query OK, 0 rows affected (0.01 sec)
```

Figure 13: Create password for root user

```
ubuntu@ip-172-31-2-179:~$ sudo mysql_secure_installation
Securing the MySQL server deployment.
```

Figure 14: Install MySQL secure server

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y Success.

All done!
```

Figure 15: Configure the MySQL server

### Install phpMyAdmin database

ubuntu@ip-172-31-10-169:~\$ sudo apt install phpmyadmin

Figure 15: Install phpMyAdmin database

Open the following file and edit:



Figure 18: Restart your Apache server



Figure 19: access phpMyAdmin by URL: IP-address/phpMyAdmin/index.php

Login to phpMyAdmin:



Figure 20: Login to phpMyAdmin with valid credentials

### Install PHP

```
ubuntu@ip-172-31-10-169:-$ sudo apt install php libapache2-mod-php php-mbstring php-xmlrpc php-soap php-gd php-xml php-cli php-zip php-bcmath php-tokenizer php-json php-pear php-mysql php-culi php-cli php-zip php-bcmath php-tokenizer php-json php-pear php-mps php-mps php-mps php-mps php-mps php-soap php-soap php-xml php-cli php-zip php-bcmath php-tokenizer php-json php-gad php-xmlrpc php-cli php-cli php-cli php-cli is already tree... Done
php-cli is already the newest version (2:8.1+92ubuntul).
```

Figure 21: Run the following command to install PHP

```
ubuntu@ip-172-31-10-169:~$ php -v
PHP 8.1.2-lubuntu2.6 (cli) (built: Sep 15 2022 11:30:49) (NTS)
Copyright (c) The PHP Group
```

Figure 22: Check your php installed version

Figure 23: Add following lines to /var/www/html/info.php



Figure 24: Accessing the following URL Confirms PHP installation:

### Clone GitHub project

Add code from GitHub to local repository, clone your project from GitHub

```
ubuntu@ip-172-31-10-169:~$ mkdir eCommerce
ubuntu@ip-172-31-10-169:~$ cd eCommerce/
ubuntu@ip-172-31-10-169:~/eCommerce$ git clone https://github.com/khadijatuzzehr
a/eShopLarave18
Cloning into 'eShopLarave18'...
remote: Enumerating objects: 244, done.
remote: Counting objects: 100% (244/244), done.
```

Figure 25: Create a new directory and clone project from GitHub repo

Install composer:

```
ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8$ sudo apt install composer
```

```
ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8$ composer install
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Package operations: 113 installs, 0 updates, 0 removals
```

Figure 26 and 27: Install composer by following commands

ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8\$ cp .env.example .env ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8\$ php artisan key:generate Application key set successfully.

Figure 28: Copy .env.example to .env

```
ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8$ php artisan cache:clear
Application cache cleared!
ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8$ php artisan config:cache
Configuration cache cleared!
Configuration cached successfully!
ubuntu@ip-172-31-10-169:~/eCommerce/eShopLaravel8$ php artisan migrate
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014 10 12 000000 create users table (56.99ms)
Migrating: 2014 10 12 100000 create password resets table
Migrated: 2014 10 12 100000 create password resets table (51.03ms)
Migrating: 2019 08 19 000000 create failed jobs table
Migrated: 2019 08 19 000000 create failed jobs table (51.48ms)
Migrating: 2019 12 14 000001 create personal access tokens table
Migrated: 2019 12 14 000001 create personal access tokens table (79.69ms)
Migrating: 2022 01 02 192346 create categories table
Migrated: 2022 01 02 192346 create categories table (40.08ms)
Migrating: 2022_01_03_091431_create_products_table
Migrated: 2022 01 03 091431 create products table (34.17ms)
Migrating: 2022 01 09 092546 create carts table
Migrated: 2022 01 09 092546 create carts table (27.02ms)
Migrating: 2022 01 11 041105 create orders table
Migrated: 2022 01 11 041105 create orders table (36.22ms)
Migrating: 2022 01 11 041448 create order items table
Migrated: 2022 01 11 041448 create order items table (28.34ms)
```

Figure 29: Migrate all tables to database

### Deployment

To start Laravel Project, run the following commands

```
ubuntu@ip-172-31-38-120:/$ sudo cp -r /home/ubuntu/blog/Laravel_project-Content-Blog-Manageme nt-System /var/www/html

ubuntu@ip-172-31-38-120:/$ sudo chgrp -R www-data /var/www/html/Laravel_project-Content-Blog-Management-System/
ubuntu@ip-172-31-38-120:/$ sudo chmod -R 775 /var/www/html/Laravel_project-Content-Blog-Management-System/storage
ubuntu@ip-172-31-38-120:/$ cd /etc/apache2/sites-available
ubuntu@ip-172-31-38-120:/etc/apache2/sites-available$ sudo nano laravel_project.conf
ubuntu@ip-172-31-38-120:/etc/apache2/sites-available$ sudo a2dissite 000-default.conf
Site 000-default disabled.

To activate the new configuration, you need to run:
    systemctl reload apache2
ubuntu@ip-172-31-38-120:/etc/apache2/sites-available$ sudo a2ensite laravel_project
Enabling site laravel_project.

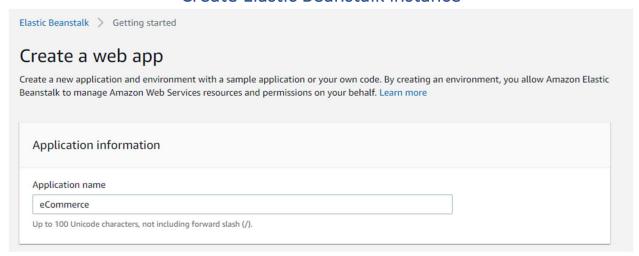
To activate the new configuration, you need to run:
    systemctl reload apache2
ubuntu@ip-172-31-38-120:/etc/apache2/sites-available$ sudo a2enmod rewrite
Module rewrite already enabled
ubuntu@ip-172-31-38-120:/etc/apache2/sites-available$ sudo systemctl restart apache2
```

### **URL**

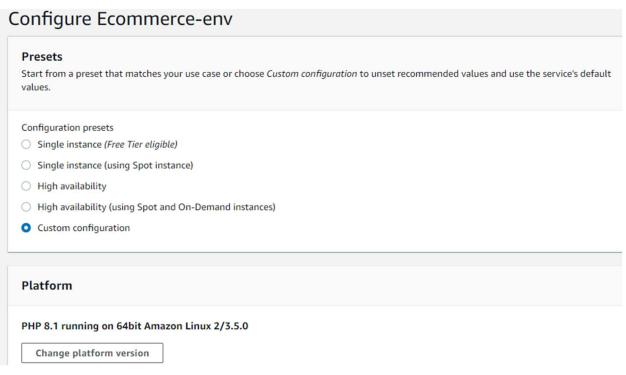
http://ec2-13-231-241-128.ap-northeast-1.compute.amazonaws.com/

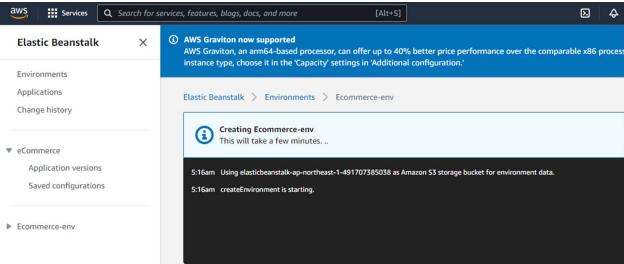
### Part 2

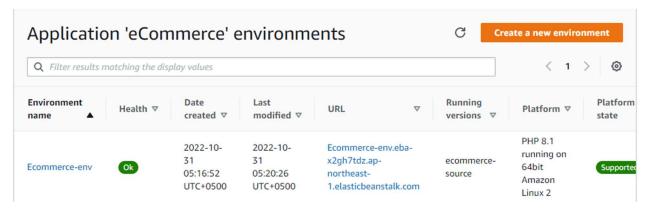
### Create Elastic Beanstalk instance

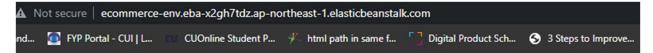






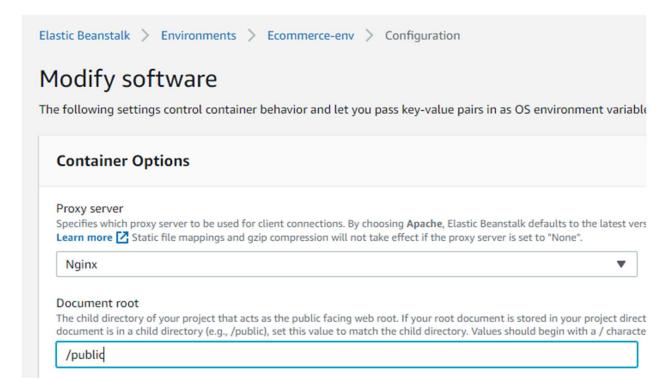


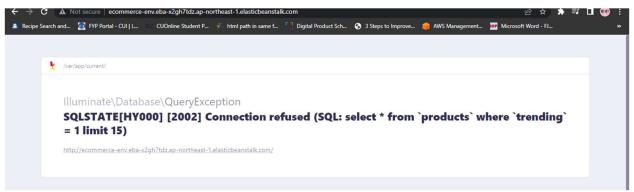




### 403 Forbidden

nginx/1.20.0

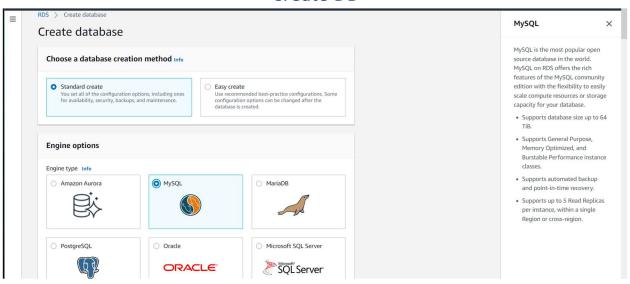


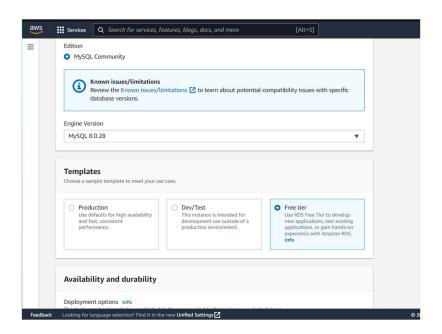


### Connect to Elastic Beans

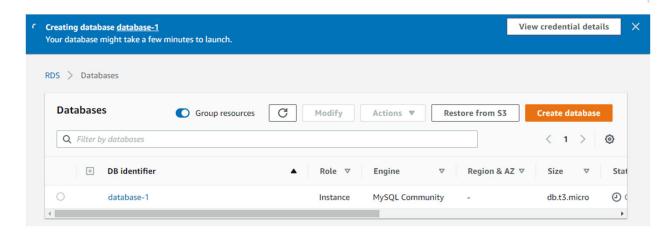
```
[ec2-user@ip-172-31-46-252 html]$ cd /etc/nginx
[ec2-user@ip-172-31-46-252 nginx]$ cd conf.f
-bash: cd: conf.f: No such file or directory
[ec2-user@ip-172-31-46-252 nginx]$ cd conf.d
[ec2-user@ip-172-31-46-252 conf.d]$ ls
elasticbeanstalk healthd_logformat.conf php-fpm.conf
[ec2-user@ip-172-31-46-252 conf.d]$ cd elasticbeanstalk/
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ ls
healthd.conf php.conf
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ sudo touch laravel.conf
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ ls
healthd.conf laravel.conf php.conf
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ sudo nano laravel.conf
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ sudo systemctl restart nginx
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$ sudo systemctl restart nginx
[ec2-user@ip-172-31-46-252 elasticbeanstalk]$
```

### Create DB





# Master username Info Type a login ID for the master user of your DB instance. phpmyadmin 1 to 16 alphanumeric characters. First character must be a letter. Auto generate a password Amazon RDS can generate a password for you, or you can specify your own password. Master password Info Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign). Confirm password Info



### Configure .env file

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:NSXrWWltsvRmNowMy+PzvO9juF9EtJ6L9MQxt7HziL4=
APP_DEUG=true
APP_URL=http://Ecommerce-env.eba-x2gh7tdz.ap-northeast-l.elasticbeanstalk.com/
LOG_CHANNEL=stack
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=database-l.cmzhdpyzlgno.ap-northeast-l.rds.amazonaws.com
DB_PORT=3306
DB_DATABASE="database-l"
DB_USERNAME=phpmyadmin
DB_PASSWORD=phpmyadmin
```

[ec2-user@ip-172-31-46-252 html]\$ sudo php artisan migrate

# URL – part 02

http://ecommerce-env.eba-x2gh7tdz.ap-northeast-1.elasticbeanstalk.com