**Lab Steps**

Task 1: Sign in to AWS Management Console

1. Click on the **** button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,

* Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
* Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button

     3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**

Task 2: Creating an Elastic Beanstalk Application

In this task, you will learn how to navigate to Elastic Beanstalk and create a new environment with specific configurations.

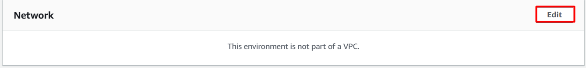
1. Navigate to Elastic Beanstalk by clicking on the menu in the top, then click on Elastic Beanstalk in the Compute section.
2. Once Elastic Beanstalk is opened, you’ll be presented with a getting started screen. Click on the 
3. On this page, we will enter simple details to get your environment up and running:
   * **Application Name**: Enter **Sample\_Application**
   * Note: This is usually the name of the product/project that you are building.
   * Leave the Application tags as default.
   * Under Platform: Choose **PHP**
4. Scroll down and click on the **Configure more options** button.

Graphical user interface, text, application

Description automatically generated

1. Scroll down and Click on **Edit** button besides the **Capacity** tab.  
   Graphical user interface, text, application

   Description automatically generated
2. Under **Instance Types**, **remove the existing instance types** and select **t2.micro** from the drop-down.  
   Text

   Description automatically generated with medium confidence
3. Click on **Save** button.
4. Click on the **Edit** button besides the **Network** tab.  
   
5. Under **Instance Settings**,
   * **Check** the **Public IP address** checkbox
   * **Check** the **us-east-1a** and **us-east-1b** subnets checkbox under **Instance subnets**.Graphical user interface, application

     Description automatically generated
   * Click on **Save** button.
6. Scroll down and click on 

Task 3: Creating Elastic Beanstalk Blue Environment

In this task, we will create a Blue environment with PHP using Elastic Beanstalk. We will begin by creating a new Elastic Beanstalk environment with PHP as the platform, followed by deploying our Sample code to the environment.

1. On the left side panel, click on the **Applications** option.
2. Click on the **Application name (Sample\_Application)** to see the environments present.  
   Graphical user interface, text, application, email

   Description automatically generated
3. Once you have created the Sample\_Application. **It has a default environment**.Table

   Description automatically generated
4. You need to create a **new environment** for the **Sample\_Application i.e. Blue environment**
5. To do so, click on **Create a new environment** button
6. On the **Select Environment Tier** page, select the **Web server environment** and then click on **Select** button.Graphical user interface, text, application, email

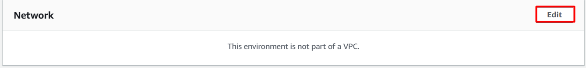
   Description automatically generated
7. On the **Create a web server environment** page provide the **Environment name** as **whizlabs-blue-environment**
8. **Note:** The environment name should be unique, if you face any errors while providing the name, then provide a different name.
9. In the platform, under **Managed platform** choose **PHP** from the dropdown.  
   Graphical user interface, text, application, email

   Description automatically generated
10. Scroll down and click on the **Configure more options** button.  
    Graphical user interface, text, application

    Description automatically generated
11. Scroll down and Click on **Edit** button besides the **Capacity** tab.  
    Graphical user interface, text, application

    Description automatically generated
12. Under **Instance Types**, remove the existing instance types and select **t2.micro** from the drop-down.  
    Text

    Description automatically generated with medium confidence
13. Click on **Save** button.
14. Click on the **Edit** button besides the **Network** tab.



1. Under **Instance Settings**,
   1. **Check** the **Public IP address** checkbox
   2. **Check** the **us-east-1a** and **us-east-1b** subnets checkbox underthe **Instance subnets**.Graphical user interface, application

      Description automatically generated
   3. Click on **Save** button.
2. Scroll down and click on 
3. The environment will take **5 to 10** minutes to provision the resources. Be patient until the environment setup is complete.
4. In the meantime, you can see the resources being provisioned one-by-one
5. Once the **whizlabs-blue-environment** is provisioned, you will get a page with **Health status** as **ok** and with a URL for the environment.
6. **Click** on the **URL** at the top-right corner to be navigated to the PHP application page, as shown below:  
   Graphical user interface, text

   Description automatically generated

Task 4: Creating Elastic Beanstalk Green Environment

In this task, we will configure the environment to use a Blue/Green deployment strategy by creating an identical Green environment and testing the deployment on it.

1. On the left side panel, click on the **Applications** option.
2. Click on the **Application name (Sample\_Application)** to see the environments present.  
   Graphical user interface, text, application, email

   Description automatically generated
3. Now there will be **two environments**.  
   Table

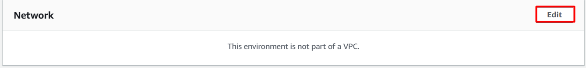
   Description automatically generated
4. You need to **create a new environment** for the **Sample\_Application i.e. Green environment**
5. To do so, click on **Create a new environment** button.
6. On the **Select Environment Tier** page, select the **Web server environment** and then click on **Select** button.Graphical user interface, text, application, email

   Description automatically generated
7. On the **Create a web server environment** page provide the **Environment name** as **whizlabs-green-environment**
8. **Note:** The environment name should be unique, if you face any errors while providing the name, then provide a different name.
9. In the platform, under **Managed platform** choose **Node.Js** from the dropdown.  
   Graphical user interface, text, application, email

   Description automatically generated
10. Scroll down and click on the **Configure more options** button.  
    Graphical user interface, text, application

    Description automatically generated
11. Scroll down and Click on **Edit** button besides the **Capacity** tab.  
    Graphical user interface, text, application

    Description automatically generated
12. Under **Instance Types**, **remove the existing instance types** and select **t2.micro** from the drop-down.  
    Text

    Description automatically generated with medium confidence
13. Click on **Save** button.
14. Click on the **Edit** button besides the **Network** tab.  
    
15. Under **Instance Settings**,
    1. **Check** the **Public IP address** checkbox
    2. **Check** the **us-east-1a and us-east-1b** subnets checkbox under the **Instance subnets**.Graphical user interface, application

       Description automatically generated
    3. Click on **Save** button.
16. Scroll down and click on 
17. The environment will take 5 to 10 minutes to provision the resources. Be patient until the environment is provisioned. In the meantime, you can see the resources being provisioned one by one.
18. Once the **whizlabs-green-env** is set up, you will get a page with **Health status** as **ok** and with a URL for the environment.Graphical user interface, application

    Description automatically generated
19. Click on the **URL** at the top-right corner to be navigated to the Node.js application page, as shown below:

Graphical user interface, text, website

Description automatically generated

Task 5: Swapping the URLs from Blue to Green

In this task, we will swap the URLs from the Blue environment to the Green environment on Elastic Beanstalk. 

1. Now we have two environments namely **whizlabs\_blue\_environment** with **PHP** and **whizlabs\_green\_environment** with **Node.js**
2. Next, we will need to **swap the URLs.**
3. In the **Elastic Beanstalk application dashboard** from **whizlabs\_green\_env,** click on **Actions** and**select Swap Environment URLs.**

Graphical user interface, text, application

Description automatically generated

1. Choose the **Environment name** as **whizlabs-blue-environment** from the dropdown in the **Select an Environment to Swap** section and then click on **Swap.**

Graphical user interface, text, application, email

Description automatically generated

1. The **swap will take a few seconds** to complete. You will see the **Successfully completed status** under **Recent Events.**

Graphical user interface, text, application, email

Description automatically generated

1. Once the **Swap is completed,** note that the **URL** of the **whizlabs\_green\_env** has been replaced with that of the **whizlabs\_blue\_environment.**

A picture containing text

Description automatically generated

1. Now **Click on the URL** (which was the URL for the whizlabs\_blue\_environment) and notice the **content of whizlabs\_green\_env instead.  
   Graphical user interface, text, website

   Description automatically generated**

**NOTE**: Wait for 2-5 minutes after swapping the URLs, if the new **page contents don't show up,** then clear the browser cache or try to access the URL from a different browser.

Task 6: Validation Test

1. Once the lab steps are completed, please click on the A picture containing text

   Description automatically generated button on the left side panel.
2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.
3. Sample output :

Graphical user interface, text, application, email

Description automatically generated

**Completion and Conclusion**

1. You have successfully created an **Elastic BeanStalk application.**
2. You have successfully created a **Blue environment with a PHP application.**
3. You have successfully **accessed the Blue environment’s URL** and verified that the content of the URL is a **PHP application page .**
4. You have successfully created a **Green environment with Node.js application.**
5. You have successfully **accessed the Green environment’s URL** and verified that the content of the URL is **Node.js application page.**
6. From the Green environment, you have successfully**initiated a swap of the environment URL.**
7. You have successfully verified that the **green environment’s URL** is **swapped** with that of **blue environment.**
8. You have successfully **accessed the new URL of the green environment** and verified that the **content of the URL** is **Node.js application page.**

**End Lab**