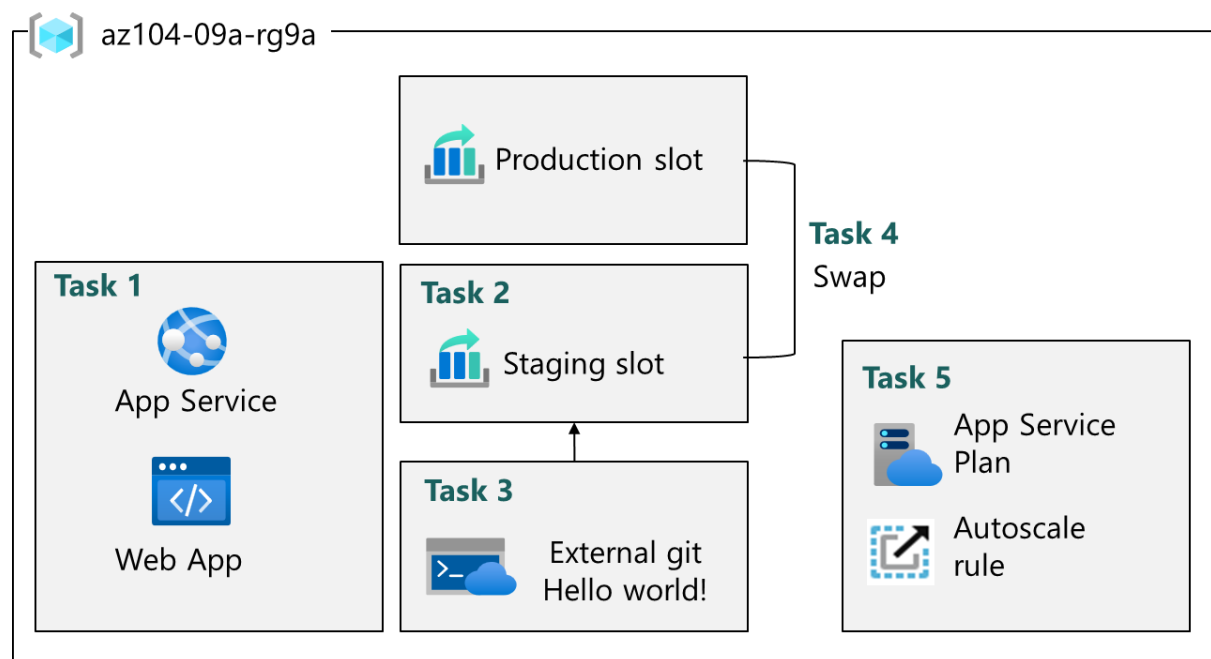


## Implement Azure Web Apps



### Task 1: Create and Configure Azure Web App

**Objective:** Deploy a PHP-based web app using Azure App Services.

#### Steps:

1. Sign in to [Azure Portal](#).
2. Search for **App Services** and select **+ Create > Web App**.
3. Fill in the following settings:
  - **Subscription:** Your Azure subscription
  - **Resource Group:** az104-rg9
  - **Web App Name:** Any globally unique name
  - **Publish:** Code
  - **Runtime Stack:** PHP 8.2
  - **Operating System:** Linux
  - **Region:** East US
  - **Pricing Plan:** Premium V3 P1V3
4. Click **Review + Create**, then **Create**.
5. After deployment, click **Go to resource**.

#### Screenshot Prompt:

## Create Web App ...

Basics Database Deployment Networking Monitor + secure Tags Review + create

### Summary



**Web App**  
by Microsoft

**Premium V3 (P1V3) sku**

Estimated price - 113.15 USD/Month



Basic authentication for this app is currently disabled and may impact deployments. Click to learn more.

### Details

Subscription	3b719365-011d-4d73-9835-1ac5f0fed5c4
Resource Group	az104-rg9-lod54493559
Name	WebApp54493559
Secure unique default hostname	Enabled
Publish	Code
Runtime stack	PHP 8.2

Create

< Previous

Next >

[Download a template for automation](#)

## Overview blade of webapp

Home > [Microsoft.Web-WebApp-Portal-59db4538-9442](#) | Overview >



**WebApp54493559**



Web App

[Browse](#) [Stop](#) [Swap](#) [Restart](#) [Delete](#) [Refresh](#) [Download publish profile](#) [Reset publish profile](#) ...

### Essentials

Resource group [\(move\)](#)  
[az104-rg9-lod54493559](#)

Status  
Running

Location [\(move\)](#)  
East US

Subscription [\(move\)](#)  
[AZ-104T00A CSR 2](#)

Subscription ID  
3b719365-011d-4d73-9835-1ac5f0fed5c4

Tags [\(edit\)](#)  
[Add tags](#)

Default domain  
[webapp54493559-eye3a2dgdttetbah5.eastus-01.azurewebsites.n](#)

App Service Plan  
[ASP-az104rg9lod54493559-9ba9 \(P1v3: 1\)](#)

Operating System  
Linux

Health Check  
[Not Configured](#)

Properties Monitoring Logs Capabilities Notifications (0) Recommendations

## Task 2: Create and Configure Deployment Slot


**Objective:** Set up a staging slot for pre-production testing.

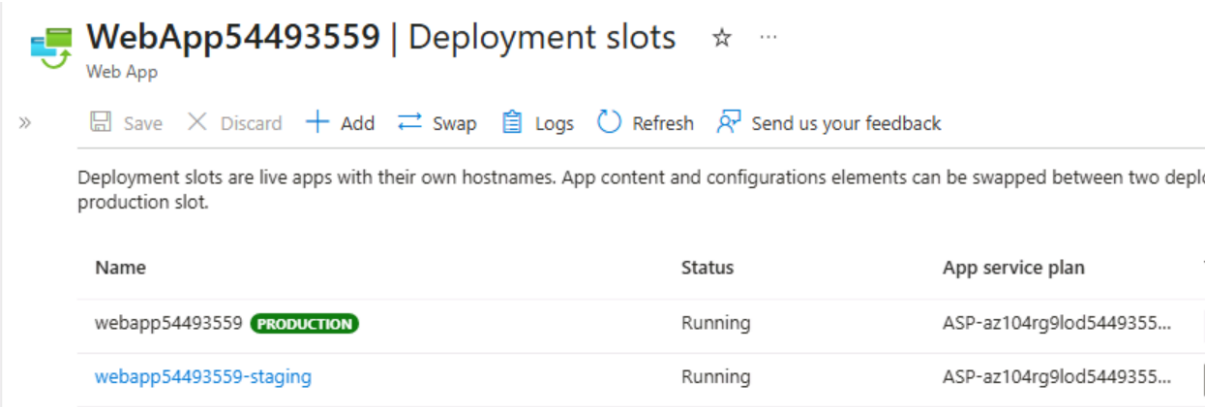
### Steps:

1. On the Web App blade, click the **Default domain** link to view the default page.

2. Close the tab and go to **Deployment > Deployment slots**.
3. Click **Add Slot**:
  - **Name**: staging
  - **Clone Settings From**: Do not clone
4. Click **Add** and refresh to see both slots.
5. Click the **staging slot** to open its blade.

#### Screenshot Prompt:

 Capture the Deployment Slots blade showing both Production and Staging.



**WebApp54493559 | Deployment slots** ☆ ...

Web App

» Save Discard Add Swap Logs Refresh Send us your feedback

Deployment slots are live apps with their own hostnames. App content and configurations elements can be swapped between two deployment slots.

Name	Status	App service plan
webapp54493559 <b>PRODUCTION</b>	Running	ASP-az104rg9lod5449355...
webapp54493559-staging	Running	ASP-az104rg9lod5449355...

### Task 3: Configure Web App Deployment Settings

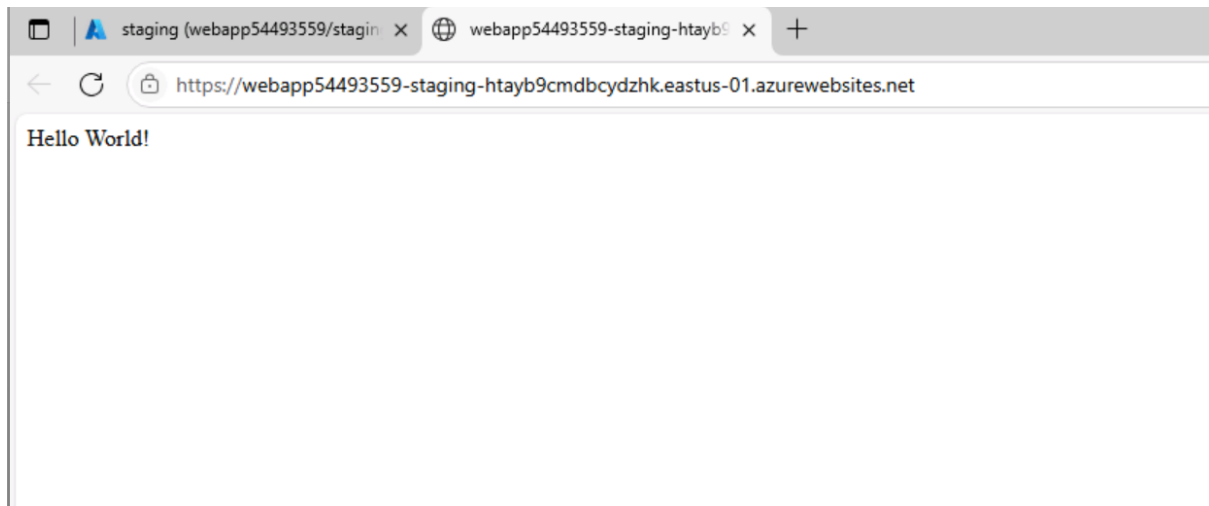
**Objective:** Connect staging slot to GitHub for continuous deployment.

#### Steps:

1. In the staging slot, go to **Deployment Center > Settings**.
2. Set:
  - **Source**: External Git
  - **Repository**: <https://github.com/Azure-Samples/php-docs-hello-world>
  - **Branch**: master
3. Click **Save**.
4. Go to **Overview > Default domain** and open the URL.

#### Screenshot Prompt:

 Take a screenshot of the Hello World page in the browser.




#### Task 4: Swap Deployment Slots

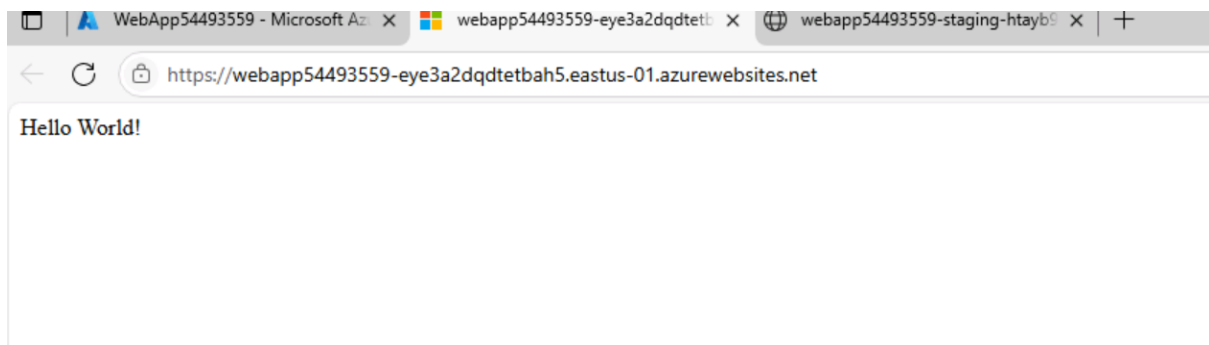
**Objective:** Move tested code from staging to production.

**Steps:**

1. Go to **Deployment slots** and click **Swap**.
2. Review settings and click **Start Swap**.
3. Return to the Web App Overview and open the **Default domain**.

**Screenshot Prompt:**

 Capture the production slot showing the Hello World page.



#### Task 5: Configure and Test Autoscaling

**Objective:** Enable autoscaling to handle traffic spikes.

**Steps:**

1. In the production slot, go to **Settings > Scale out (App Service plan)**.
2. Set:
  - **Scaling Mode:** Automatic
  - **Maximum Burst:** 2
3. Click **Save**.

4. Go to **Diagnose and solve problems > Load Test your App > Create Load Test**.
5. Name the test, click **Review + Create**, then **Create**.
6. In the test plan:
  - Click **Add Request**
  - Paste your **Default domain URL**
  - Click **Add**, then **Create**
7. Monitor live metrics: Virtual users, Response time, Requests/sec.
8. Click **Stop** to end the test.

### Screenshot Prompts:

 *Capture the autoscale settings page.*

Scale out method

☐ Manual  
Maintain a constant instance count for your application

☒ Automatic  
Platform managed scale out and in based on traffic

☐ Rules Based  
User defined rules to scale on a schedule or based on any app metric

Maximum burst ⓘ

2

Always ready instances ⓘ

1

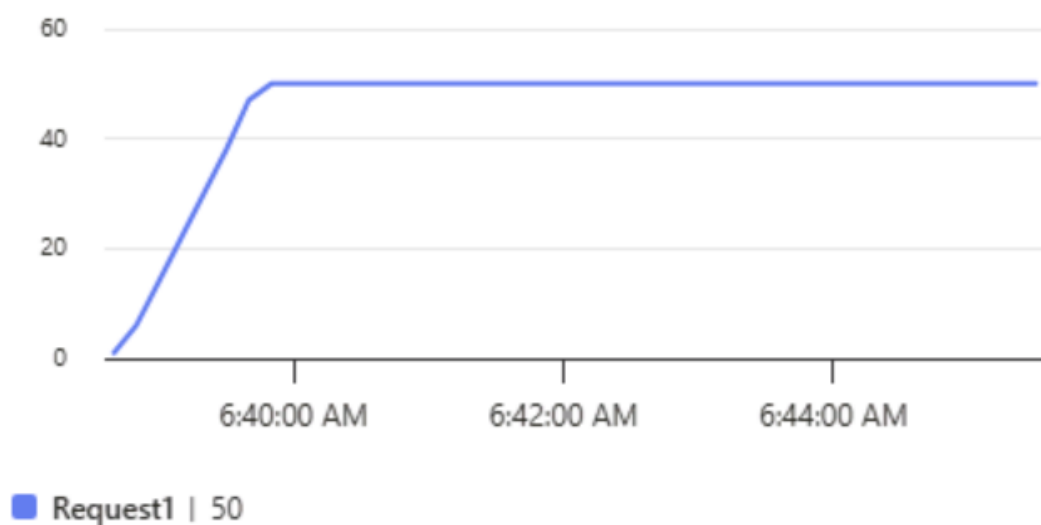
Enforce scale out limit ⓘ

Save

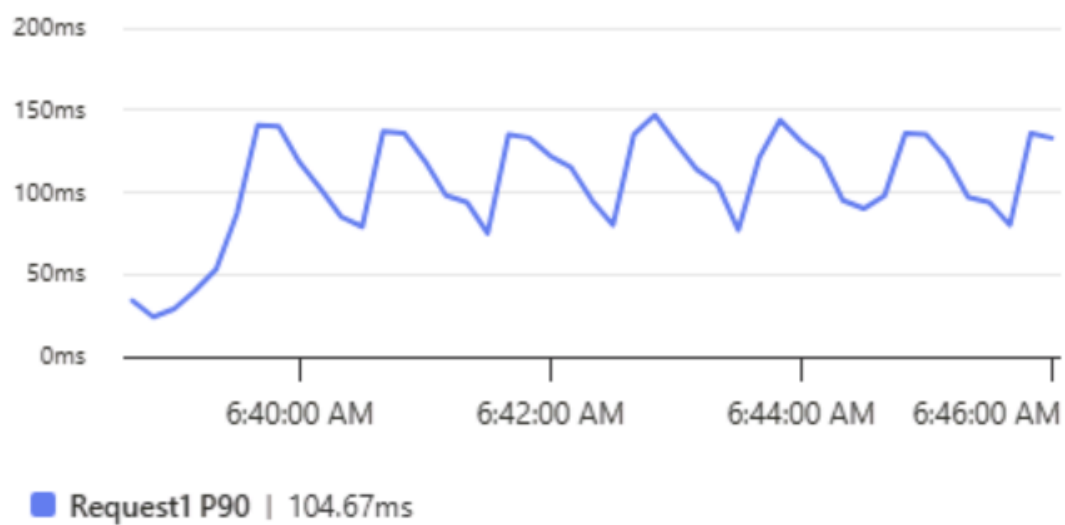
Discard

 *Capture the live metrics dashboard during load testing*

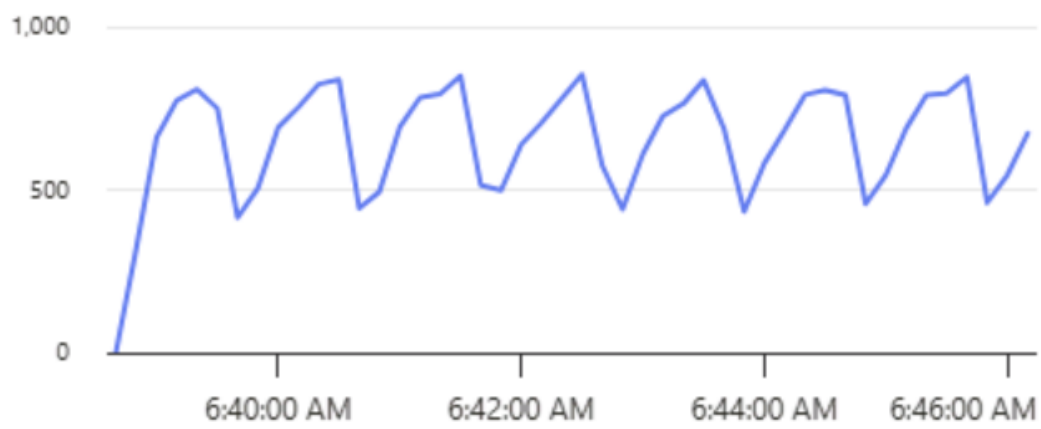
### Virtual Users (Max)



### Response time (successful responses)



## Requests/sec (Avg)



■ Request1 | 649.98