Bootcamp-Project 1

BUILD A CI/CD PIPELINE FOR A .NET WEB APPLICATION USING AZURE DEVOPS

Mohanramrajan Erran Bothalraj

Table of Contents

Introduction

In modern software development, **Continuous Integration (CI) and Continuous Deployment (CD)** play a crucial role in ensuring efficient, automated, and reliable application delivery. This project focuses on setting up a **CI/CD pipeline** in **Azure DevOps** for a .NET application, enabling seamless code integration, automated testing, and controlled deployments to Azure App Service.

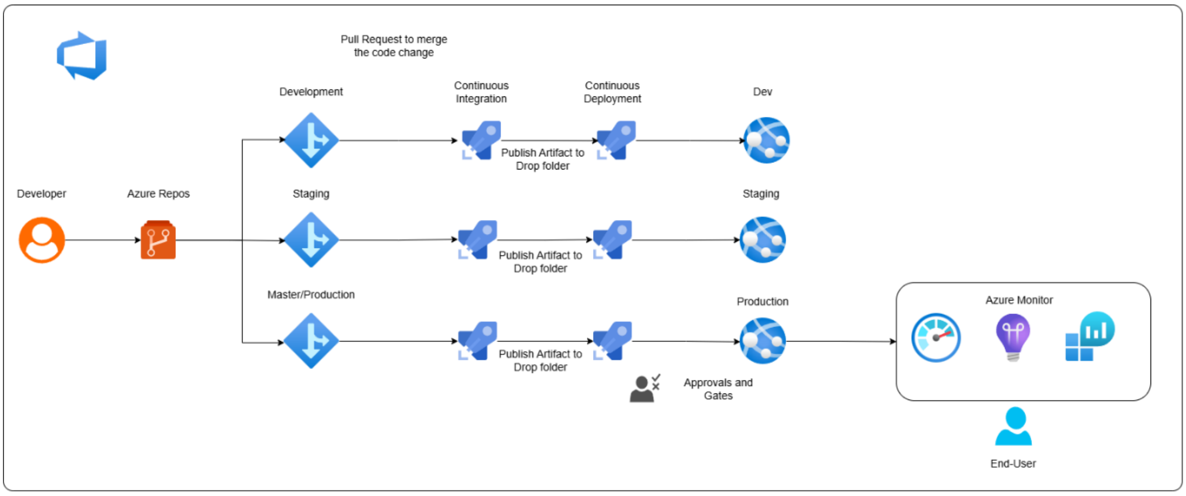
Project Objectives

- **Source Code Management:** Store and manage the application code in **Azure Repos (Git)**.
- **Automated Build Process:** Set up a **CI pipeline** to restore dependencies, build the application, and run unit tests.
- **Deployment Automation:** Configure a **CD pipeline** to deploy the application to different environments (Dev, Staging, Production).
- **Approval Workflows:** Implement **manual approval gates** for production deployments.
- **Application Monitoring:** Use **Azure Monitor and Application Insights** for real-time tracking of performance and errors.

Expected Outcome

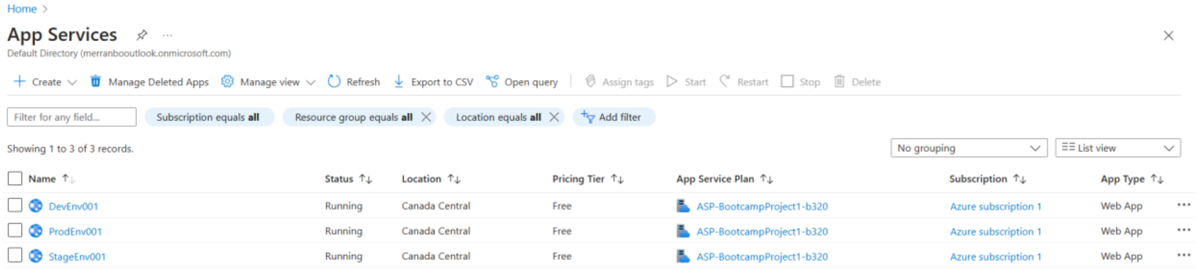
By implementing this CI/CD pipeline, the project ensures **faster development cycles, high-quality releases, and efficient application monitoring**, enabling a **DevOps-driven** approach for smooth application deployment and maintenance.

Architecture Diagram



Pre-Requisite

1. GitHub repository with a .Net application with Unit tests: https://github.com/merranbo1989/newlocalrepo.git
2. Create Web Apps for 3 environments (Dev, Staging, and Production) using Azure App Services



Home >

App Services

Default Directory (merranbooutlook.onmicrosoft.com)

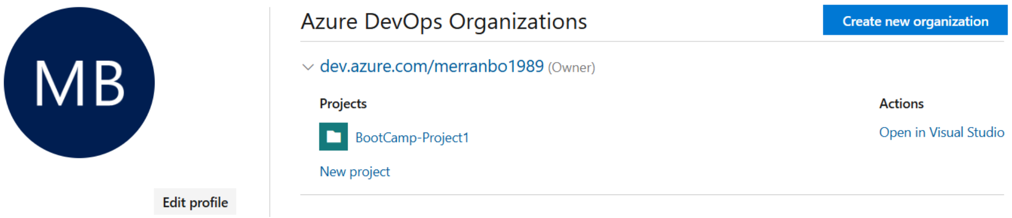
Filter for any field... | Subscription equals **all** | Resource group equals **all** | Location equals **all** | Add filter

Showing 1 to 3 of 3 records.

No grouping | List view

Name	Status	Location	Pricing Tier	App Service Plan	Subscription	App Type
DevEnv001	Running	Canada Central	Free	ASP-BootcampProject1-b320	Azure subscription 1	Web App
ProdEnv001	Running	Canada Central	Free	ASP-BootcampProject1-b320	Azure subscription 1	Web App
StageEnv001	Running	Canada Central	Free	ASP-BootcampProject1-b320	Azure subscription 1	Web App

3. Create a new Organization and Project in Azure DevOps environment.



Azure DevOps Organizations

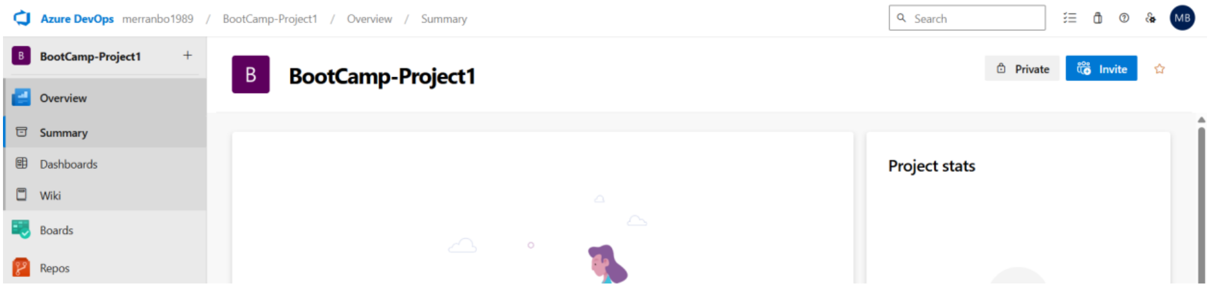
dev.azure.com/merranbo1989 (Owner)

Projects	Actions
BootCamp-Project1	Open in Visual Studio
New project	

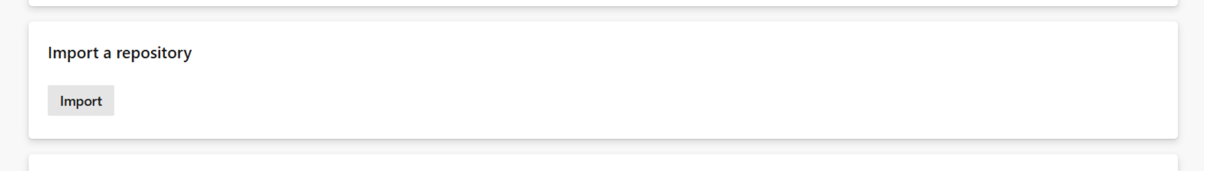
Solution Steps

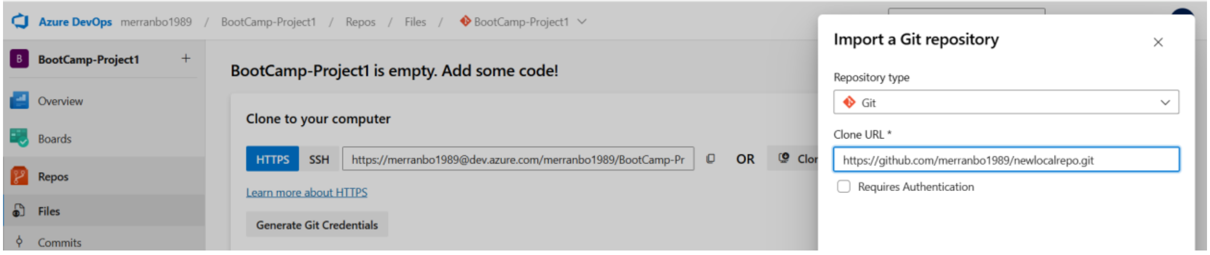
Part 1: Source Code Management (Azure Repos)

A. Create a New DevOps Project under organization



B. Import the .Net application code from the GitHub repository (Pre-Requisite step 1) into the Azure Repo (BootCamp-Project1)





Azure DevOps
merranbo1989 / BootCamp-Project1 / Repos / Files / BootCamp-Project1
BootCamp-Project1
Overview
Boards
Repos
Files
Commits

BootCamp-Project1 is empty. Add some code!

Clone to your computer

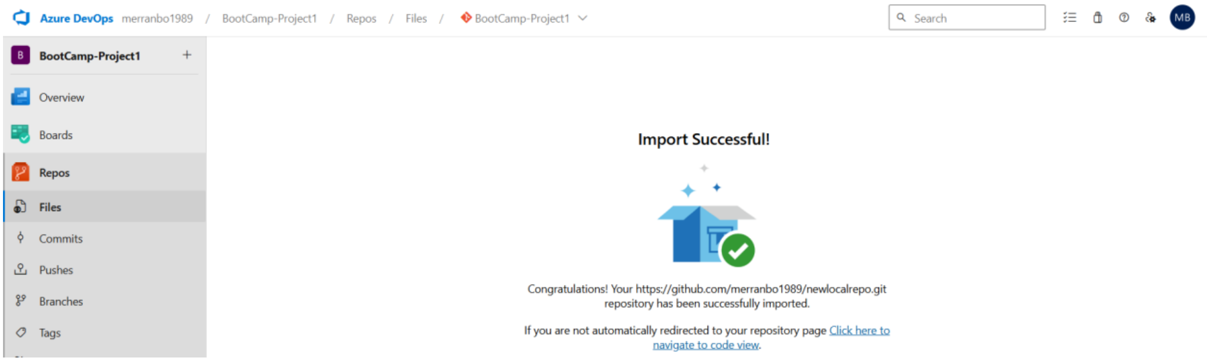
HTTPS | SSH | https://merranbo1989@dev.azure.com/merranbo1989/BootCamp-Pr | OR

Import a Git repository

Repository type: Git

Clone URL *: https://github.com/merranbo1989/newlocalrepo.git

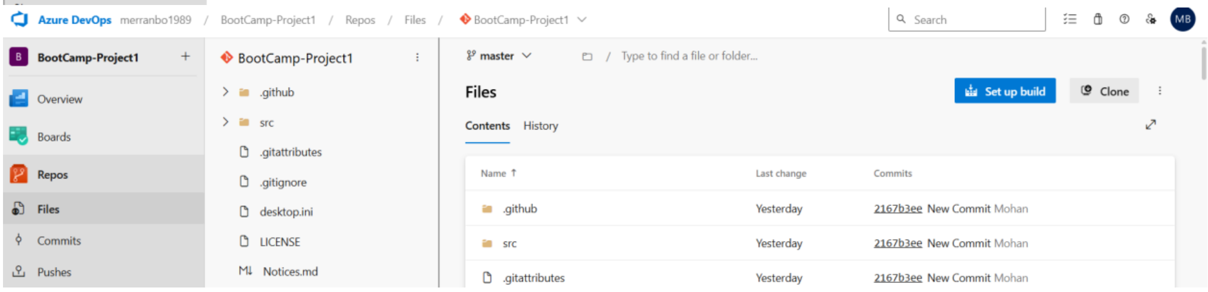
□ Requires Authentication


Azure DevOps merranbo1989 / BootCamp-Project1 / Repos / Files / BootCamp-Project1
Search
MB
BootCamp-Project1
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags

Import Successful!

Congratulations! Your https://github.com/merranbo1989/newlocalrepo.git repository has been successfully imported.

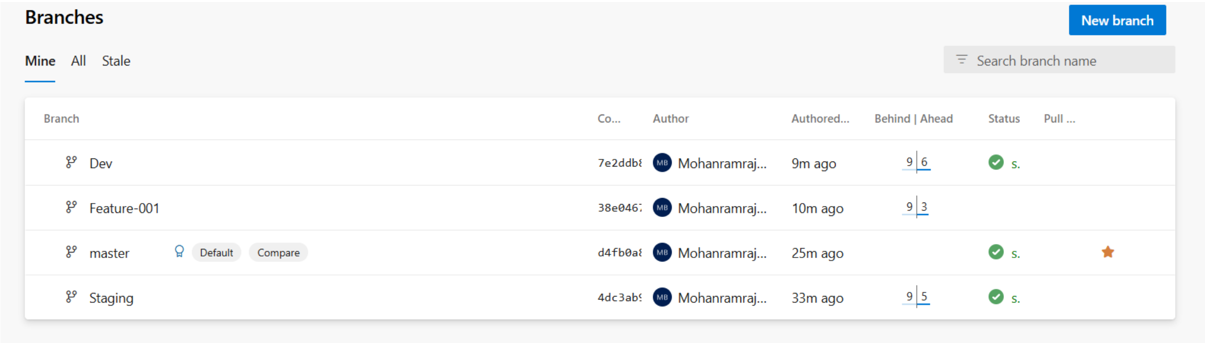
If you are not automatically redirected to your repository page Click here to navigate to code view.



Name ↑	Last change	Commits
.github	Yesterday	2167b3ee New Commit Mohan
src	Yesterday	2167b3ee New Commit Mohan
.gitattributes	Yesterday	2167b3ee New Commit Mohan

C. Create the following branches in the Azure Repos for code management

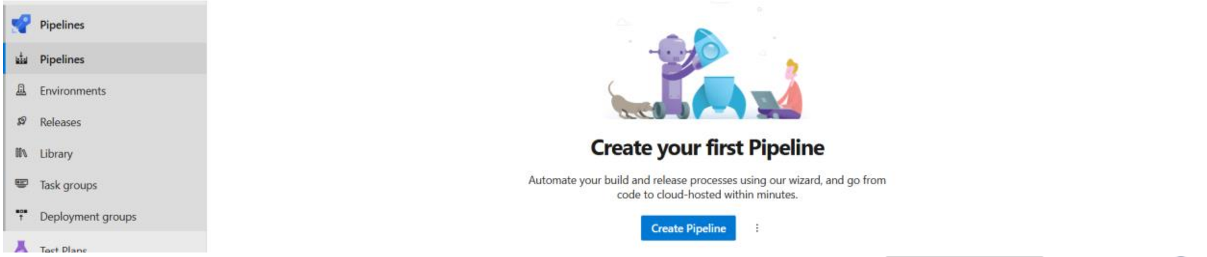
- Master (default): used for stable code to be deployed in PROD environment
- Dev: used by develops normally contains mirror image of PROD environment
- Staging: used for testing or UAT purposes
- Feature: this branch is used to update any new/existing feature in the code and then merged into Dev branch.



Branch	Co...	Author	Authored...	Behind | Ahead	Status	Pull ...
Dev	7e2ddb	Mohanramraj...	9m ago	9 | 6	✓ s.	
Feature-001	38e046	Mohanramraj...	10m ago	9 | 3		
master Default Compare	d4fb0a	Mohanramraj...	25m ago		✓ s.	★
Staging	4dc3ab	Mohanramraj...	33m ago	9 | 5	✓ s.	

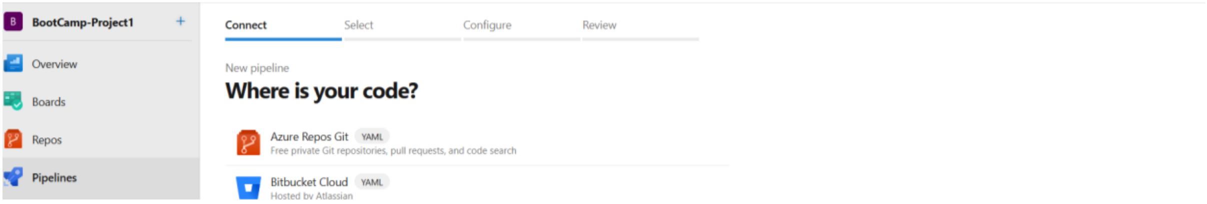
Part 2: Continuous Integration (Build Pipeline)

A. Create an Azure Build pipeline using ASP.NET template. Where a YAML file will be created with pre-loaded script.



Create your first Pipeline

Automate your build and release process using our wizard, and go from code to cloud-hosted within minutes.

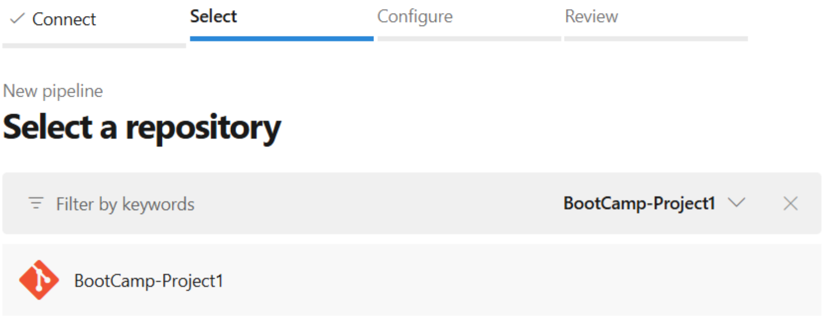


New pipeline

Where is your code?

Azure Repos Git YAML
Free private Git repositories, pull requests, and code search

Bitbucket Cloud YAML
Hosted by Atlassian



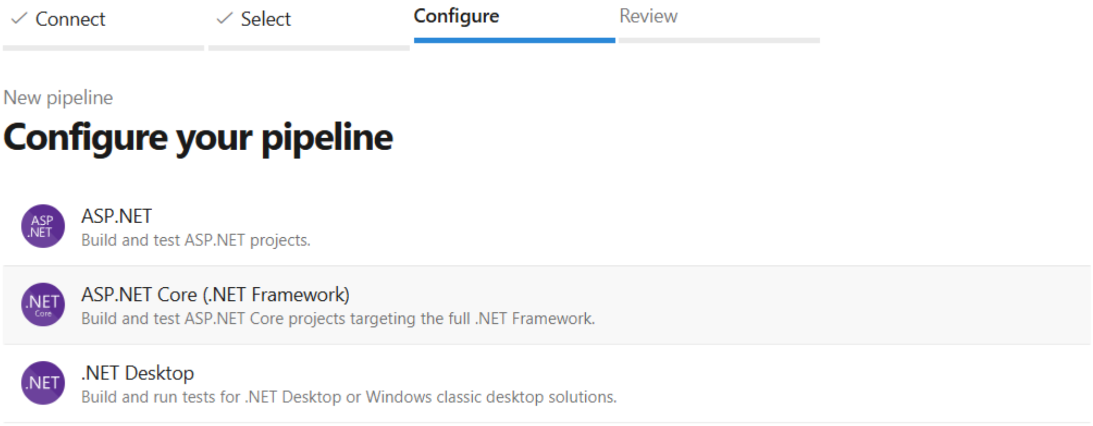
New pipeline

Select a repository

Filter by keywords

BootCamp-Project1

BootCamp-Project1



New pipeline

Configure your pipeline

ASP.NET
Build and test ASP.NET projects.

ASP.NET Core (.NET Framework)
Build and test ASP.NET Core projects targeting the full .NET Framework.

.NET Desktop
Build and run tests for .NET Desktop or Windows classic desktop solutions.

B. Update the Code in the YAML file attached here:

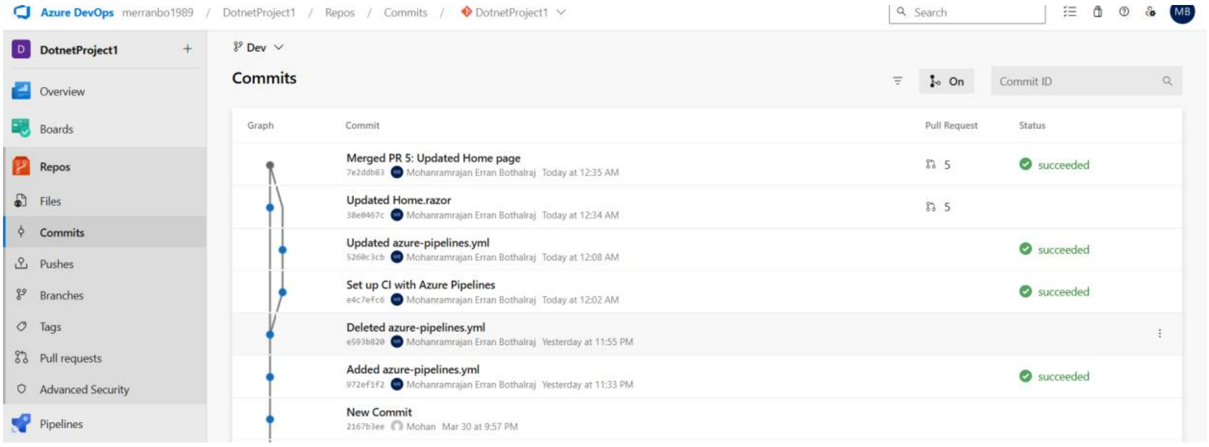
******* Code Starts Here *******

```
# ASP.NET Core (.NET Framework)
# Build and test ASP.NET Core projects targeting the full .NET Framework.
# Add steps that publish symbols, save build artifacts, and more:
# https://docs.microsoft.com/azure/devops/pipelines/languages/dotnet-core

trigger:
  branches:
    include:
      - master
      - Dev
      - Staging

pool:
  vmImage: 'windows-latest'

variables:
  solution: '**/*.sln'
  buildPlatform: 'Any CPU'
  buildConfiguration: 'Release'

steps:
- task: NuGetToolInstaller@1
  displayName: NuGet Install'
```

```
- task: NuGetCommand@2
  displayName: 'Restore Dependencies'
  inputs:
    command: 'restore'
    restoreSolution: '$(solution)'

- task: VSBuild@1
  displayName: 'Build'
  inputs:
    solution: '$(solution)'
    msbuildArgs: '/p:DeployOnBuild=true /p:WebPublishMethod=Package /p:PackageAsSingleFile=true
/p:SkipInvalidConfigurations=true
/p:DesktopBuildPackageLocation="$(build.artifactStagingDirectory)\WebApp.zip"
/p:DeployIisAppPath="Default Web Site"'
    platform: '$(buildPlatform)'
    configuration: '$(buildConfiguration)'

- task: DotNetCoreCLI@2
  displayName: 'Unit Test'
  inputs:
    command: 'test'
    projects: '**/*[Tt]est*/*.csproj'

- task: PublishBuildArtifacts@1
  displayName: 'Publish Artifact'
  inputs:
    PathtoPublish: '$(Build.ArtifactStagingDirectory)'
    ArtifactName: 'drop'
    publishLocation: 'Container'
```

***** **Code Ends Here** *****

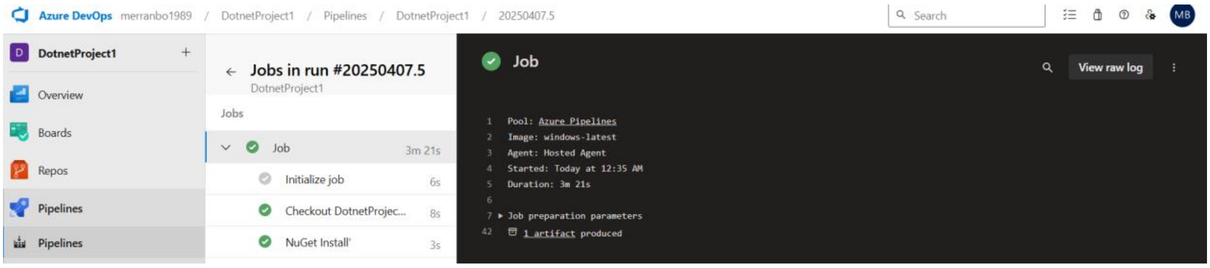
C. Commit all the changes in all the branches



Commits

Graph	Commit	Pull Request	Status
	Merged PR 5: Updated Home page 7e2ddb83 Mohanramrajan Erran Bothalraj Today at 12:35 AM	5	succeeded
	Updated Home.razor 38e0467c Mohanramrajan Erran Bothalraj Today at 12:34 AM	5	
	Updated azure-pipelines.yml 5260c3cb Mohanramrajan Erran Bothalraj Today at 12:08 AM		succeeded
	Set up CI with Azure Pipelines e4c7efc6 Mohanramrajan Erran Bothalraj Today at 12:02 AM		succeeded
	Deleted azure-pipelines.yml e593b820 Mohanramrajan Erran Bothalraj Yesterday at 11:55 PM		
	Added azure-pipelines.yml 972ef1f2 Mohanramrajan Erran Bothalraj Yesterday at 11:33 PM		succeeded
	New Commit 2167b3ee Mohan Mar 30 at 9:57 PM		

D. A new Pipeline will be created and will run automatically



Jobs

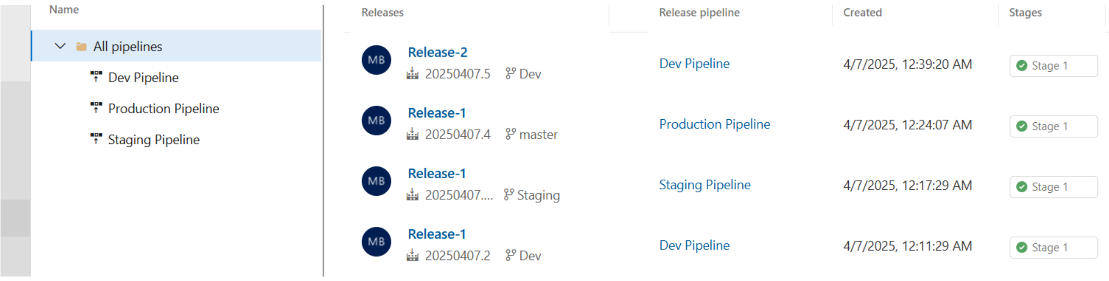
Job	3m 21s
Initialize job	6s
Checkout DotnetProjec...	8s
NuGet Install'	3s

Job

```
Pool: Azure Pipelines
Image: windows-latest
Agent: Hosted Agent
Started: Today at 12:35 AM
Duration: 3m 21s

▶ Job preparation parameters
  1 artifact produced
```

Part 3: Continuous Deployment (Release Pipeline)

A. Create three separate pipelines for Dev, Staging, and PROD environments

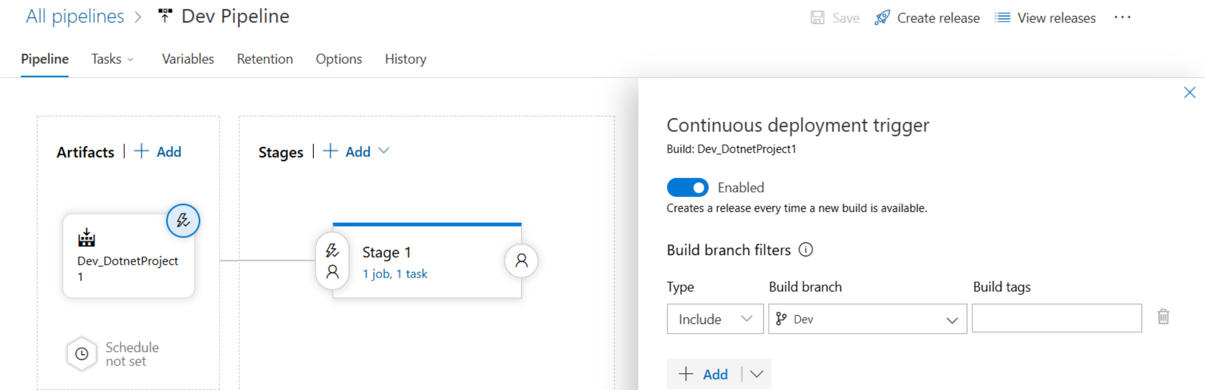


Name

- All pipelines
  - Dev Pipeline
  - Production Pipeline
  - Staging Pipeline

Releases	Release pipeline	Created	Stages
Release-2 20250407.5 Dev	Dev Pipeline	4/7/2025, 12:39:20 AM	Stage 1
Release-1 20250407.4 master	Production Pipeline	4/7/2025, 12:24:07 AM	Stage 1
Release-1 20250407.... Staging	Staging Pipeline	4/7/2025, 12:17:29 AM	Stage 1
Release-1 20250407.2 Dev	Dev Pipeline	4/7/2025, 12:11:29 AM	Stage 1

B. Configure the Dev pipeline and task to ensure the release pipeline is triggered automatically when a new build is triggered from Dev branch.



Continuous deployment trigger

Build: Dev_DotnetProject1

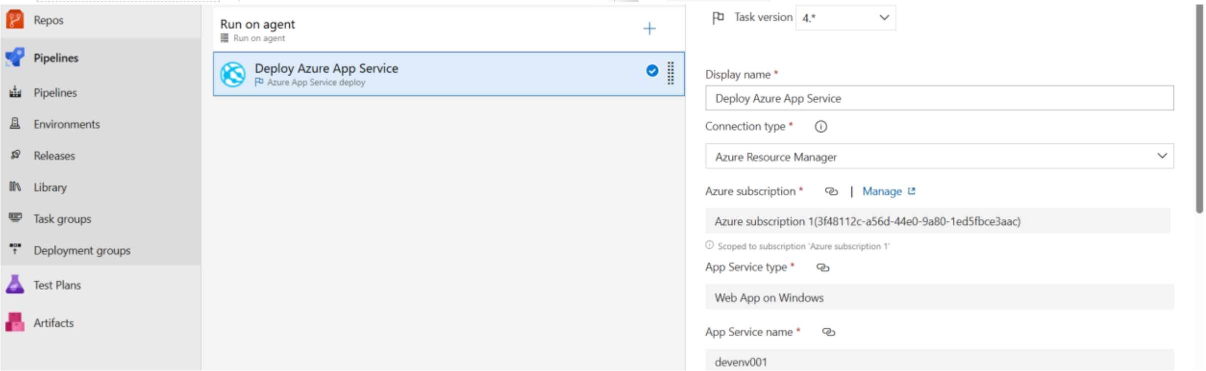
Enabled

Creates a release every time a new build is available.

Build branch filters

Type	Build branch	Build tags
Include	Dev	

+ Add



Task version: 4.*

Display name *: Deploy Azure App Service

Connection type *: Azure Resource Manager

Azure subscription *: Azure subscription 1(3f48112c-a56d-44e0-9a80-1ed5fbce3aac)

Scoped to subscription 'Azure subscription 1'

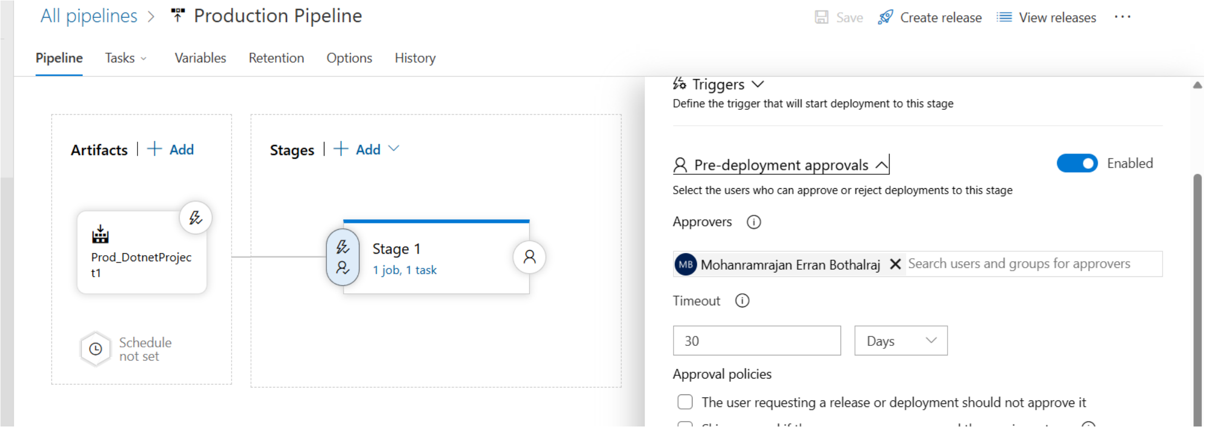
App Service type *: Web App on Windows

App Service name *: devenv001

NOTE: Repeat the step for STAGING and PRODUCTION Environment

Part 4: Approval Gates Configuration

A. Implement Manual approval of a User for Pre-Deployment approvals for Production stage.



Triggers
Define the trigger that will start deployment to this stage

Pre-deployment approvals: Enabled
Select the users who can approve or reject deployments to this stage

Approvers: Mohanramrajan Erran Bothalraj

Timeout: 30 Days

Approval policies
□ The user requesting a release or deployment should not approve it

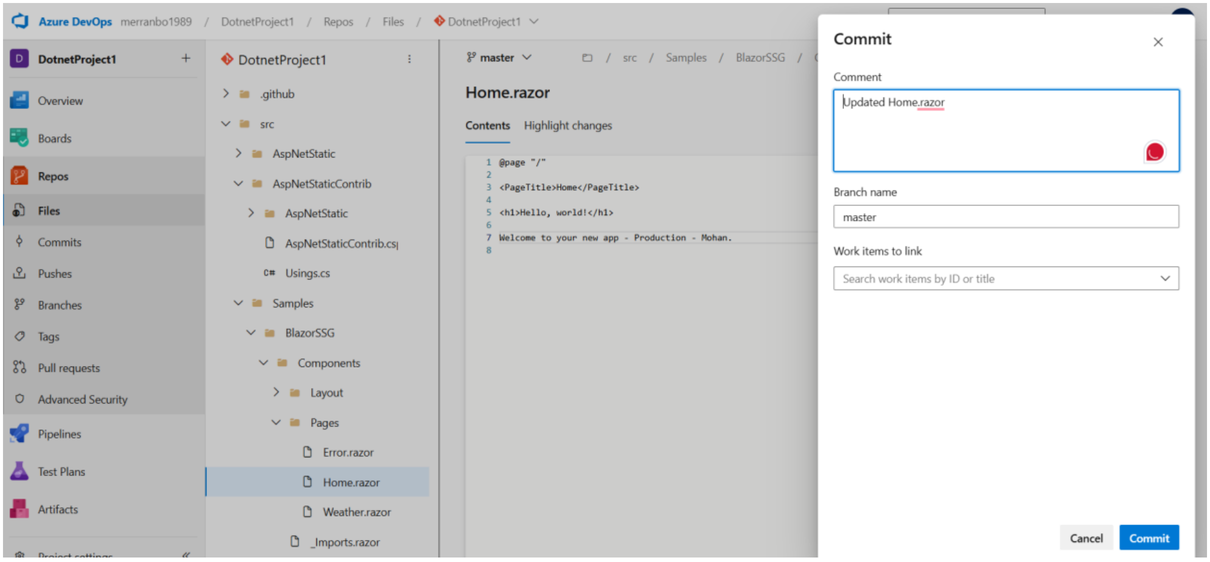
Part 5: Execution and Validation

Once the .Net application code in the “Azure Repos” are committed with any new change.

1. Azure Build Pipeline will be triggered automatically, which will then perform the tasks:
    a. Restore Dependencies
    b. Build
    c. Unit Test
    d. Publish the Artifact in Drop folder
2. After successful completion of the Build pipeline, Release pipeline will be triggered automatically, where the application will be deployed to “Development” environment followed by “Staging” Environment.
3. Finally, after fulfilling the Pre-Deployment checks, Gates, and approvals configured in “Part 4”. The application will be deployed to the “Production” environment.

Validation

A. Commit the changes in Azure Repos. (Master branch)



Home.razor

```
@page "/"

<PageTitle>Home</PageTitle>

<h1>Hello, world!</h1>

Welcome to your new app - Production - Mohan.

```

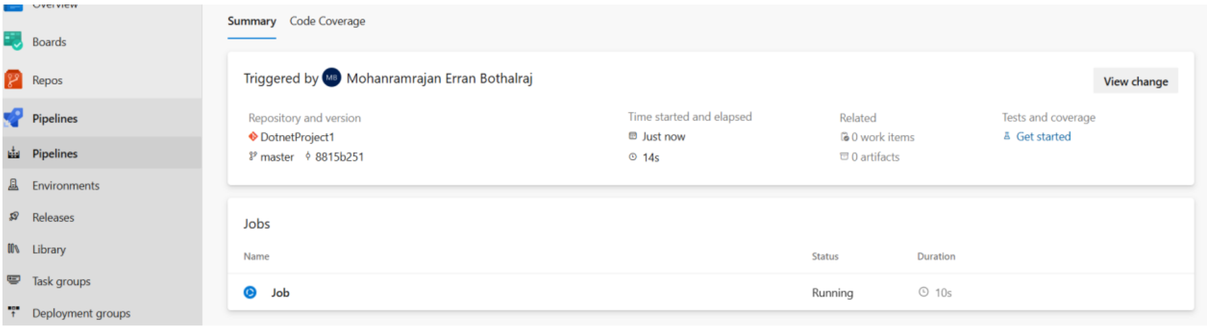
Commit

Comment: Updated Home.razor

Branch name: master

Work items to link: Search work items by ID or title

B. Build Pipeline is triggered

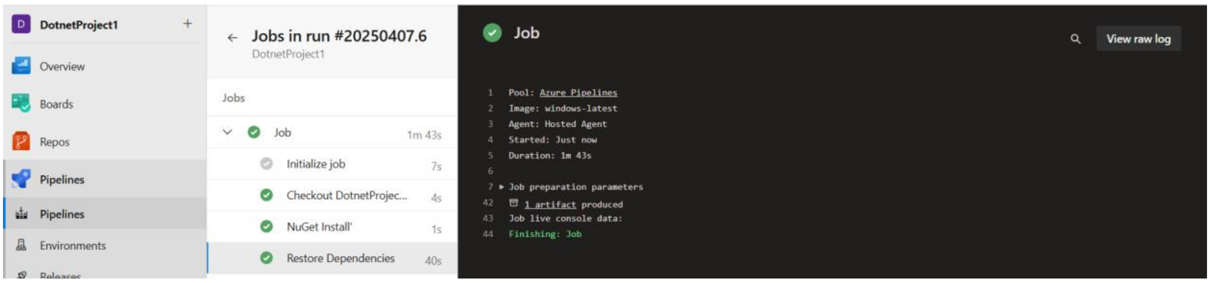


Triggered by MB Mohanramrajan Erran Bothalraj

Repository and version	Time started and elapsed	Related	Tests and coverage
DotnetProject1 master 8815b251	Just now 14s	0 work items 0 artifacts	Get started

Jobs

Name	Status	Duration
Job	Running	10s



Jobs in run #20250407.6

DotnetProject1

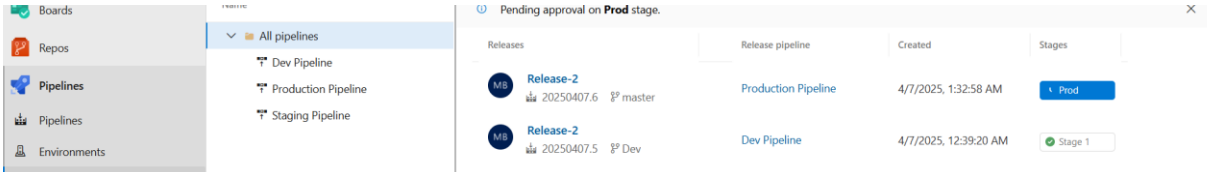
Jobs

Job	1m 43s
Initialize job	7s
Checkout DotnetProjec...	4s
NuGet Install'	1s
Restore Dependencies	40s

Job

```
Pool: Azure Pipelines
Image: windows-latest
Agent: Hosted Agent
Started: Just now
Duration: 1m 43s

▸ Job preparation parameters
1 artifact produced
Job live console data:
Finishing: Job
```

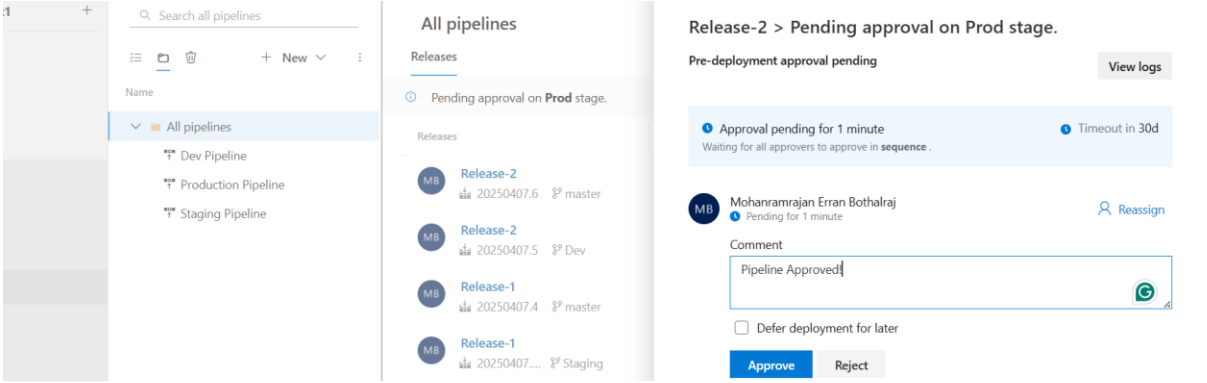
C. New Release pipeline is triggered



Pending approval on **Prod** stage.

Releases	Release pipeline	Created	Stages
Release-2 20250407.6 master	Production Pipeline	4/7/2025, 1:32:58 AM	Prod
Release-2 20250407.5 Dev	Dev Pipeline	4/7/2025, 12:39:20 AM	Stage 1

D. Go to Azure DevOps release pipeline to “Approve” the production deployment.



Pending approval on **Prod** stage.

Releases

- Release-2 20250407.6 master
- Release-2 20250407.5 Dev
- Release-1 20250407.4 master
- Release-1 20250407.... Staging

Release-2 > Pending approval on Prod stage.

Pre-deployment approval pending

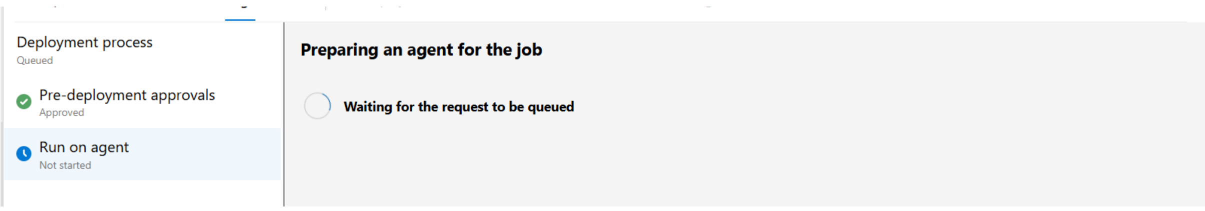
Approval pending for 1 minute — Timeout in 30d

Waiting for all approvers to approve in **sequence** .

Mohanramrajan Erran Bothalraj — Pending for 1 minute

Comment: Pipeline Approved!

□ Defer deployment for later


Deployment process — Queued

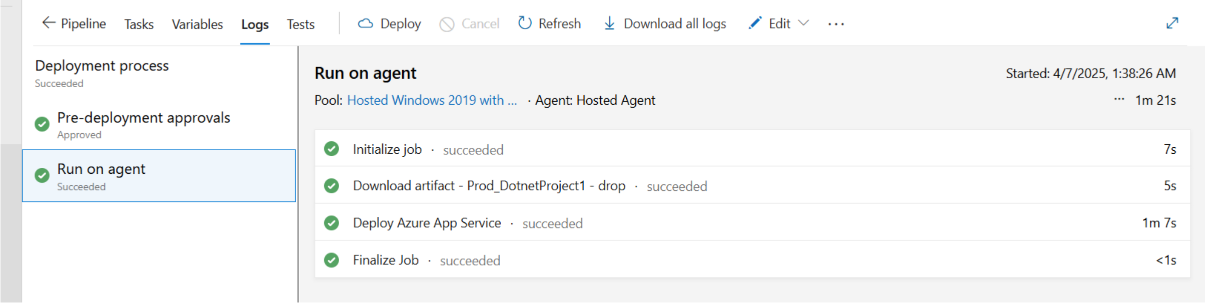
Pre-deployment approvals — Approved

Run on agent — Not started

Preparing an agent for the job

Waiting for the request to be queued

E. Validate the Production environment by launching the web page.



Deployment process — Succeeded

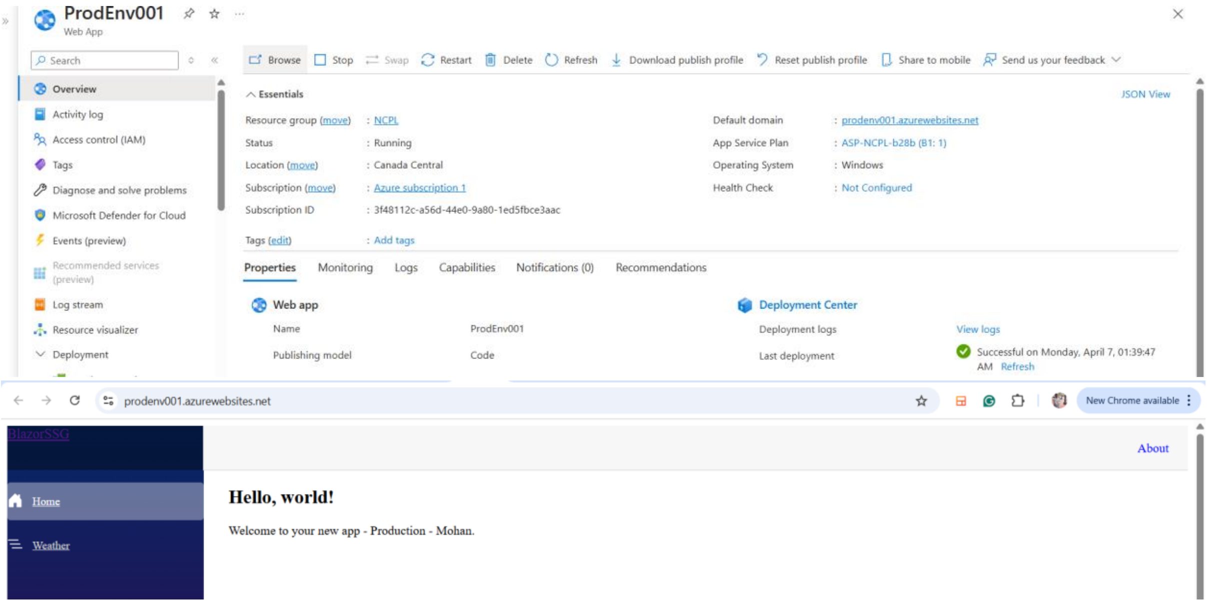
Pre-deployment approvals — Approved

Run on agent — Succeeded

Run on agent — Started: 4/7/2025, 1:38:26 AM

Pool: Hosted Windows 2019 with ... · Agent: Hosted Agent — 1m 21s

Step	Duration
Initialize job · succeeded	7s
Download artifact - Prod_DotnetProject1 - drop · succeeded	5s
Deploy Azure App Service · succeeded	1m 7s
Finalize Job · succeeded	<1s



Resource group (move) : NCPL
Status : Running
Location (move) : Canada Central
Subscription (move) : Azure subscription 1
Subscription ID : 3f48112c-a56d-44e0-9a80-1ed5fbce3aac
Tags (edit) : Add tags
Default domain : prodenv001.azurewebsites.net
App Service Plan : ASP-NCPL-b28b (B1: 1)
Operating System : Windows
Health Check : Not Configured

Web app
Name ProdEnv001
Publishing model Code

Deployment Center
Deployment logs View logs
Last deployment Successful on Monday, April 7, 01:39:47 AM Refresh

Hello, world!

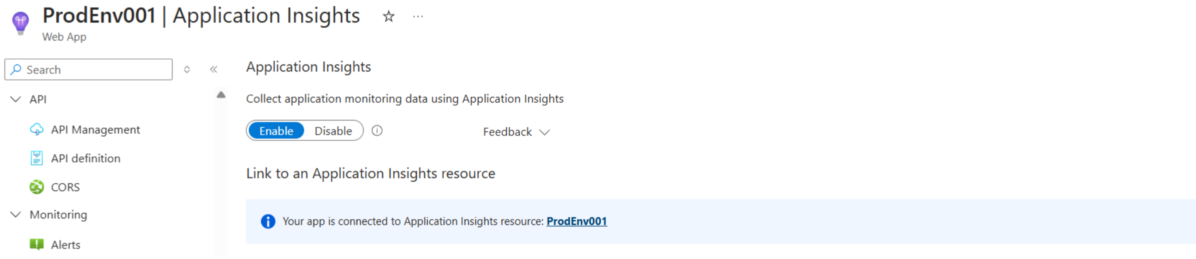
Welcome to your new app - Production - Mohan.

NOTE: Repeat the Steps for Dev, and Staging environments

Part 6: Azure Monitor

Track the application performance using Azure Monitor Application Insights.

A. Enable “Application Insights” in Prod environment.



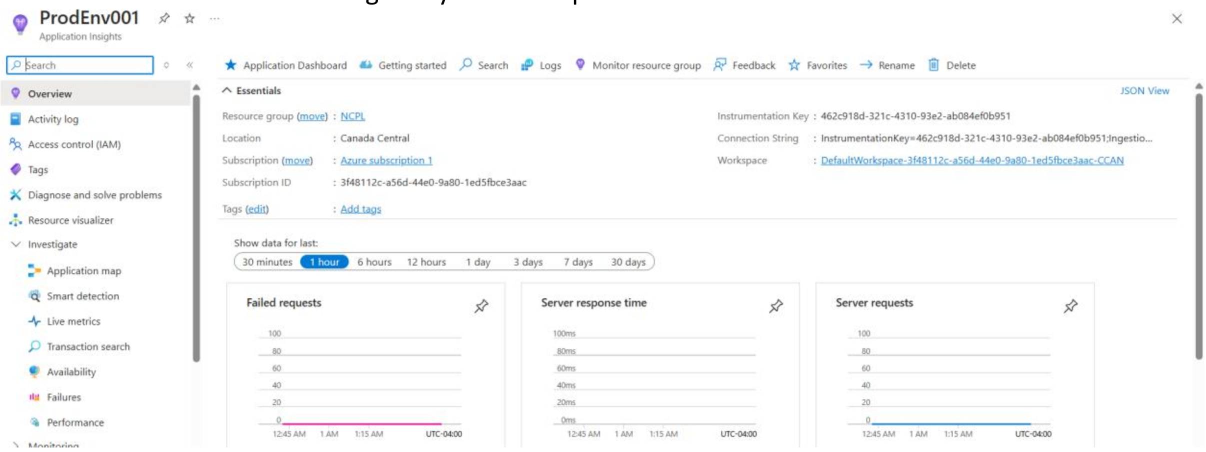
Application Insights

Collect application monitoring data using Application Insights

Link to an Application Insights resource

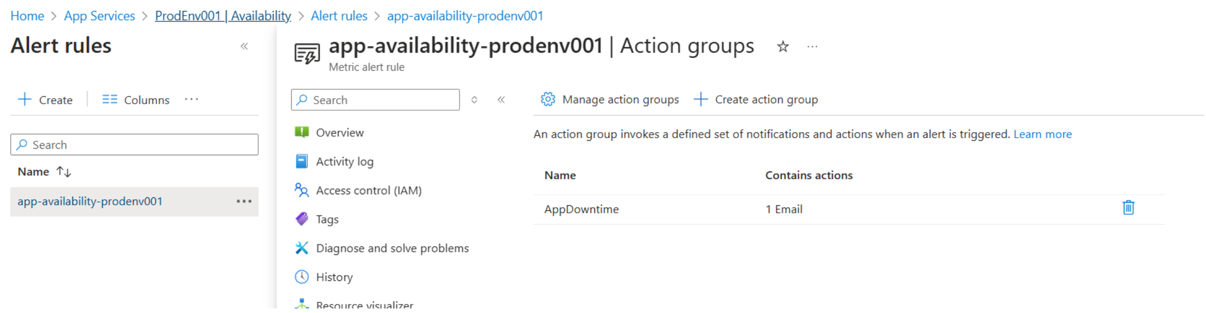
Your app is connected to Application Insights resource: ProdEnv001

B. Create the resource Log Analytics Workspace



Resource group (move) : NCPL
Location : Canada Central
Subscription (move) : Azure subscription 1
Subscription ID : 3f48112c-a56d-44e0-9a80-1ed5fbce3aac
Tags (edit) : Add tags
Instrumentation Key : 462c918d-321c-4310-93e2-ab084ef0b951
Connection String : InstrumentationKey=462c918d-321c-4310-93e2-ab084ef0b951;Ingestio...
Workspace : DefaultWorkspace-3f48112c-a56d-44e0-9a80-1ed5fbce3aac-CCAN

Validation for PROD environment availability using the Alert rule created

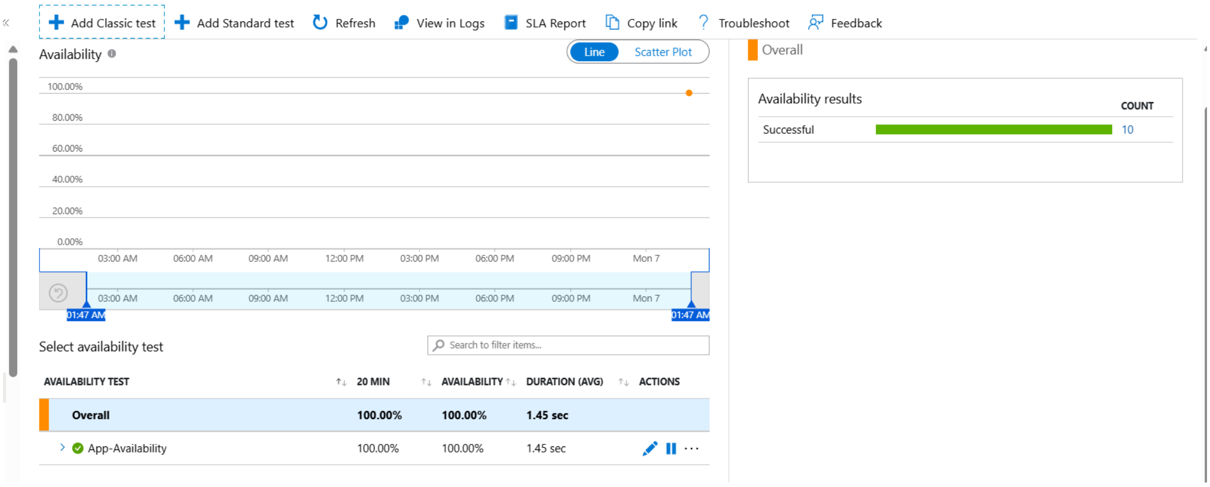


Home > App Services > ProdEnv001 | Availability > Alert rules > app-availability-prodenv001
Alert rules
+ Create
Columns
Search
Name ↑↓
app-availability-prodenv001
app-availability-prodenv001 | Action groups
Metric alert rule
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
History
Resource visualizer
Manage action groups
+ Create action group

An action group invokes a defined set of notifications and actions when an alert is triggered. Learn more

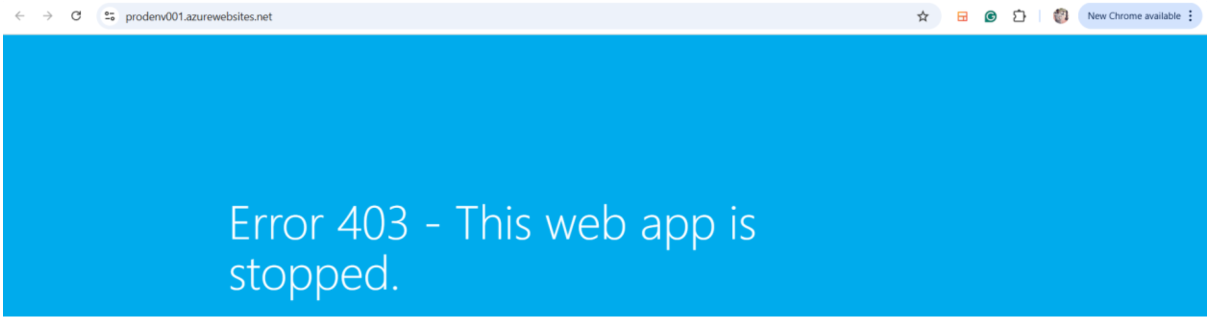
Name	Contains actions
AppDowntime	1 Email

Stop the “Prod” environment to check if the alert rule triggers the failure email to the user configured.



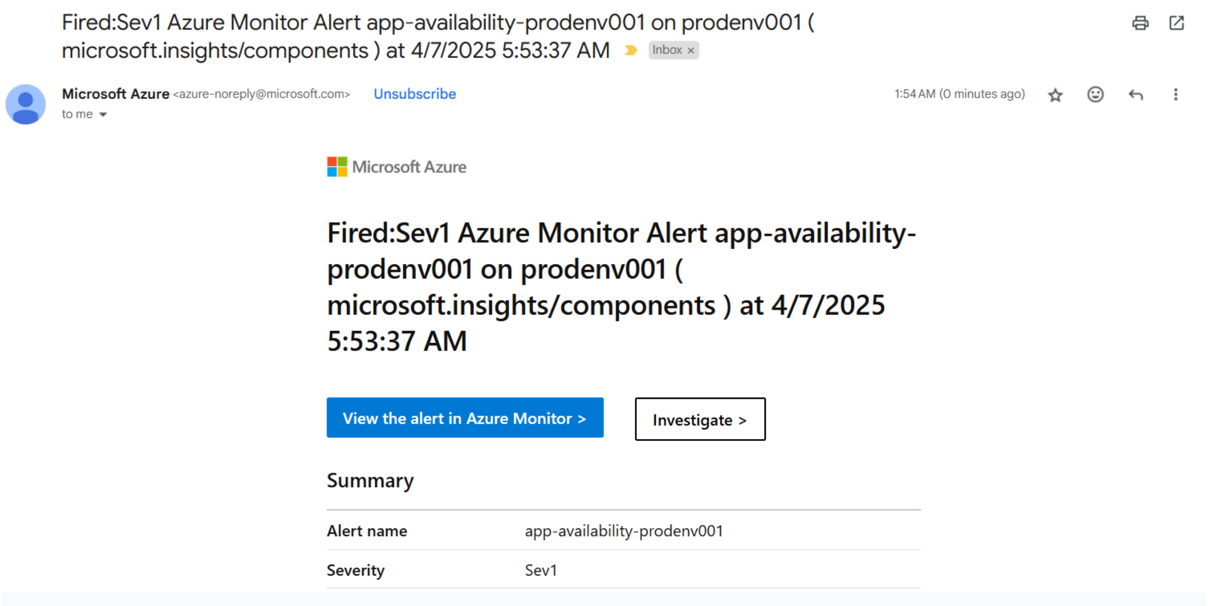
Availability results	COUNT
Successful	10

AVAILABILITY TEST	20 MIN	AVAILABILITY	DURATION (AVG)	ACTIONS
Overall	100.00%	100.00%	1.45 sec	
App-Availability	100.00%	100.00%	1.45 sec	



Error 403 - This web app is stopped.

Check the Application Insights availability section and check the mailbox for failure email



Fired:Sev1 Azure Monitor Alert app-availability-prodenv001 on prodenv001 (microsoft.insights/components) at 4/7/2025 5:53:37 AM

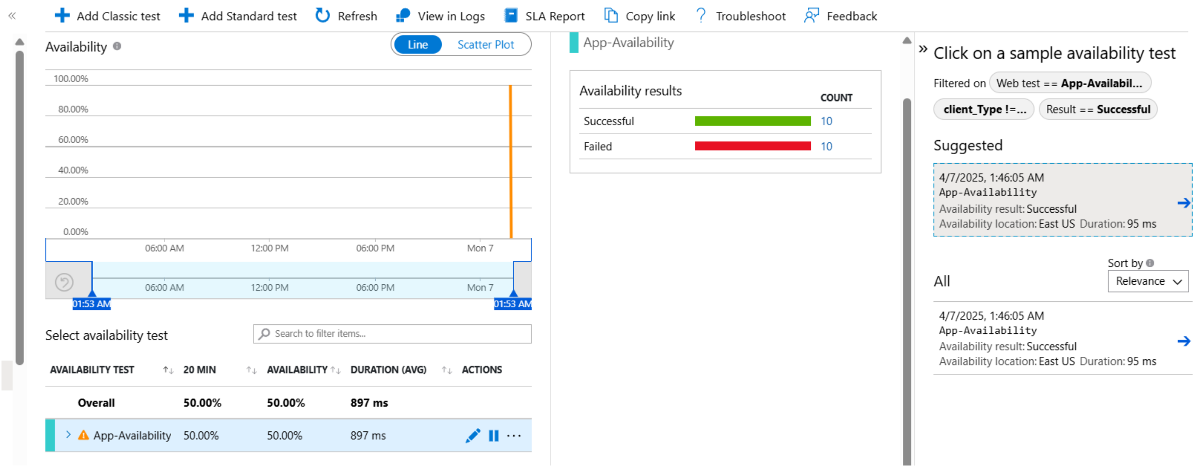
Microsoft Azure <azure-noreply@microsoft.com> Unsubscribe
to me
1:54 AM (0 minutes ago)

Microsoft Azure

Fired:Sev1 Azure Monitor Alert app-availability-prodenv001 on prodenv001 (microsoft.insights/components) at 4/7/2025 5:53:37 AM

Summary

Alert name	app-availability-prodenv001
Severity	Sev1



Select availability test

AVAILABILITY TEST	20 MIN	AVAILABILITY	DURATION (AVG)	ACTIONS
Overall	**50.00%**	**50.00%**	**897 ms**	
› ⚠ App-Availability	50.00%	50.00%	897 ms	✎ ❚❚ ⋯

Availability results

		COUNT
Successful	🟩	10
Failed	🟥	10

Click on a sample availability test

Filtered on Web test == **App-Availabil...** **client_Type !=...** Result == **Successful**

Suggested

4/7/2025, 1:46:05 AM
App-Availability
Availability result: Successful
Availability location: East US Duration: 95 ms →

All

4/7/2025, 1:46:05 AM
App-Availability
Availability result: Successful
Availability location: East US Duration: 95 ms →