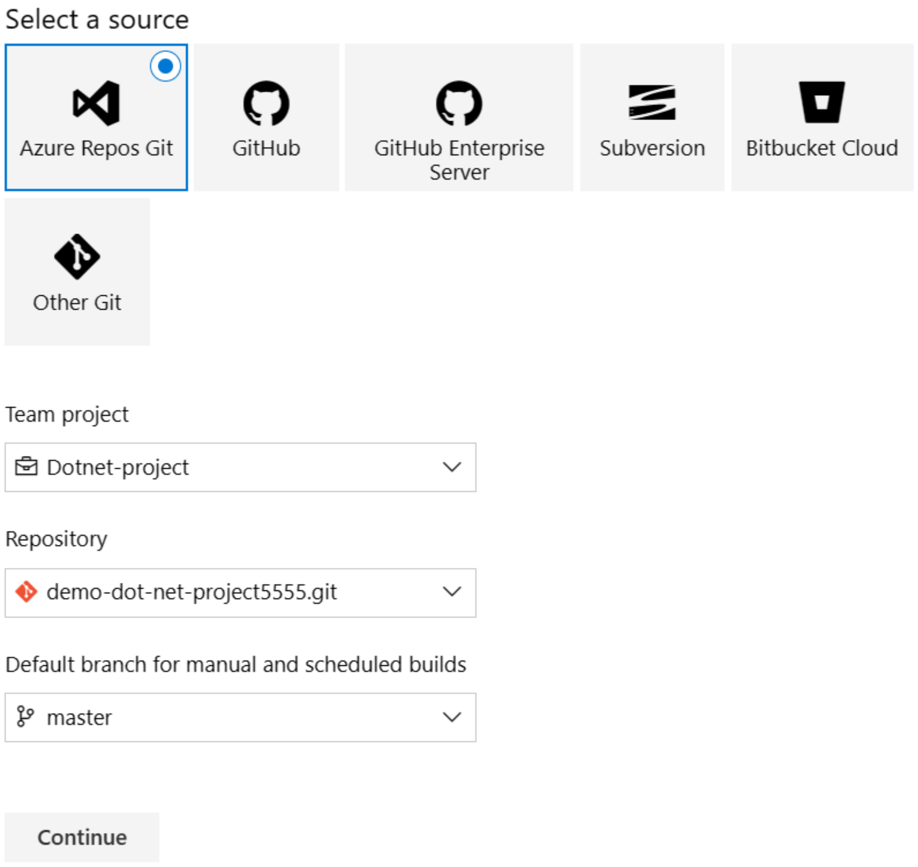
Creating a pipeline using classic editor

Steps to create a pipeline using Microsoft hosted agent for dotnet web app

Prerequisites:

→ **An Azure DevOps account and a project.**
→ **A dotnet application hosted in an Azure Repos Git repository.**

- Go to Dev.azure.com
- Create a project
- Go to Pipelines select the source code, ex: Azure Repos Git, GitHub



Select a source

● Azure Repos Git
GitHub
GitHub Enterprise Server
Subversion
Bitbucket Cloud
Other Git

Team project: Dotnet-project

Repository: demo-dot-net-project5555.git

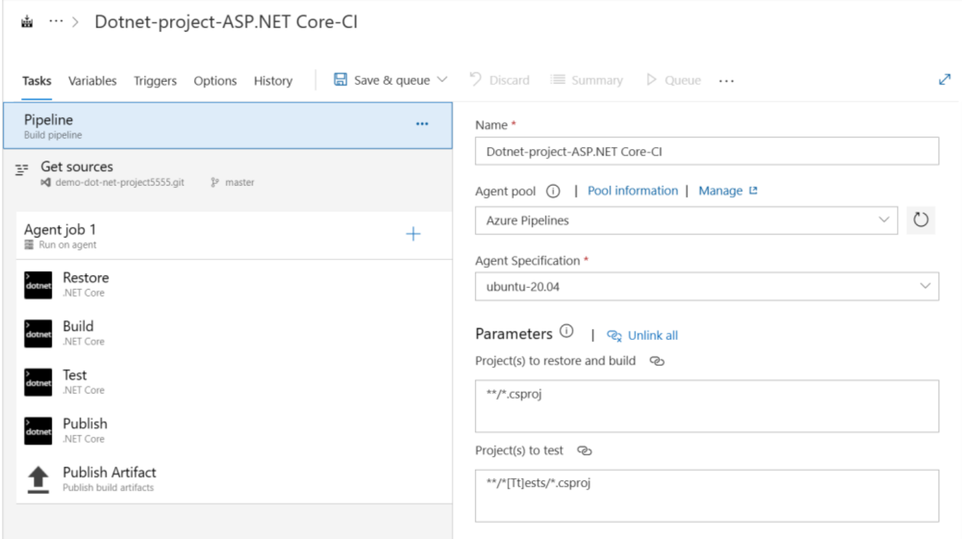
Default branch for manual and scheduled builds: master

- Select the right template ex: Asp.net core for dotnet projects

Under Task

Under Pipelines

- Name: dotnet project
- Agent pool: Azure pipelines
- Agent specification: ubuntu or macOS or anything as per our need
- No changes in parameters



Dotnet-project-ASP.NET Core-CI

Name *: Dotnet-project-ASP.NET Core-CI

Agent pool: Azure Pipelines

Agent Specification *: ubuntu-20.04

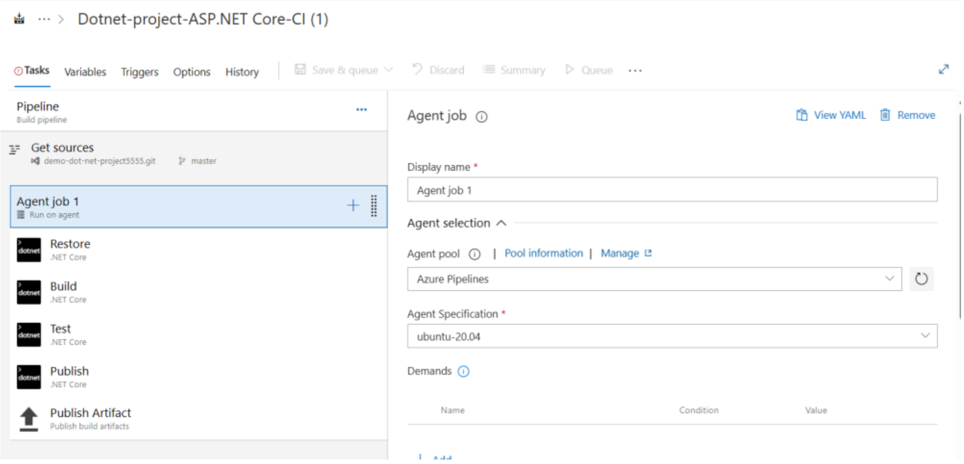
Parameters

Project(s) to restore and build: **/*.csproj

Project(s) to test: **/*[Tt]ests/*.csproj

Under Agent job 1

- Display name: Agent or anything
- Agent pool: Azure pipelines
- Agent specification: it should be the same which we take in pipelines, ex: ubuntu



Dotnet-project-ASP.NET Core-CI (1)

Agent job

Display name *: Agent job 1

Agent selection

Agent pool: Azure Pipelines

Agent Specification *: ubuntu-20.04

Demands

- Save & Queue
- Save & Run

Deploying application in Microsoft Azure

App Services

- Create Web app

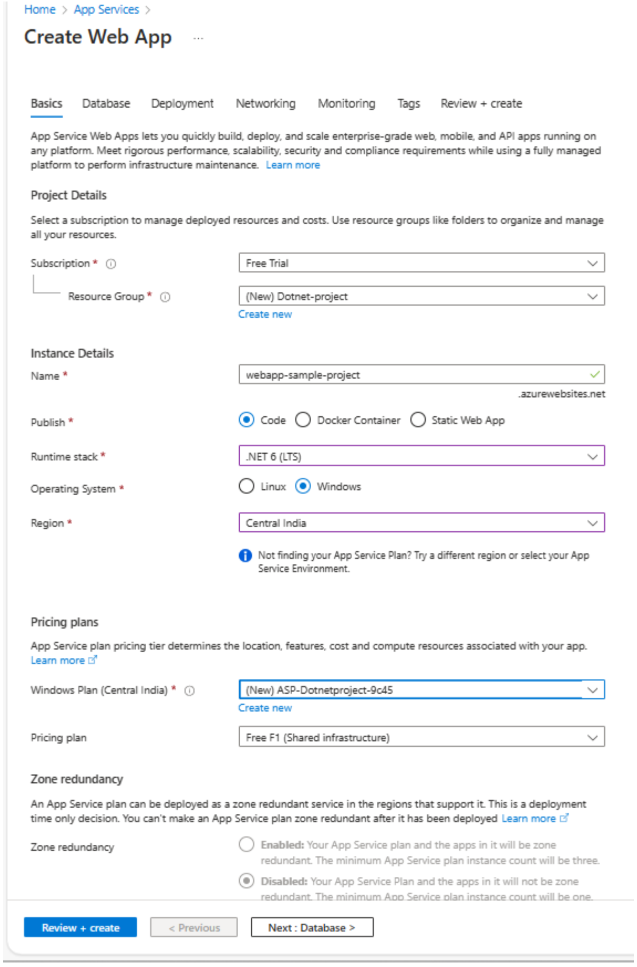
- Subscription: Free trial
- Resource group: Dot-net project

Instance Details:

- Name: webapp-sample-project
- Publish: Code
- Runtime Stack: .NET 6 (as per our project need)
- Operating system: Windows
- Region: Central India

Pricing Plans:

- Windows Plan: Dotnet project under select from dropdown
- Pricing plan: Free F1 Shared infrastructure



Create Web App

Basics Database Deployment Networking Monitoring Tags Review + create

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. Learn more

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *: Free Trial

Resource Group *: (New) Dotnet-project

Create new

Instance Details

Name *: webapp-sample-project .azurewebsites.net

Publish *: ● Code ○ Docker Container ○ Static Web App

Runtime stack *: .NET 6 (LTS)

Operating System *: ○ Linux ● Windows

Region *: Central India

Not finding your App Service Plan? Try a different region or select your App Service Environment.

Pricing plans

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app. Learn more

Windows Plan (Central India) *: (New) ASP-Dotnetproject-9c45

Create new

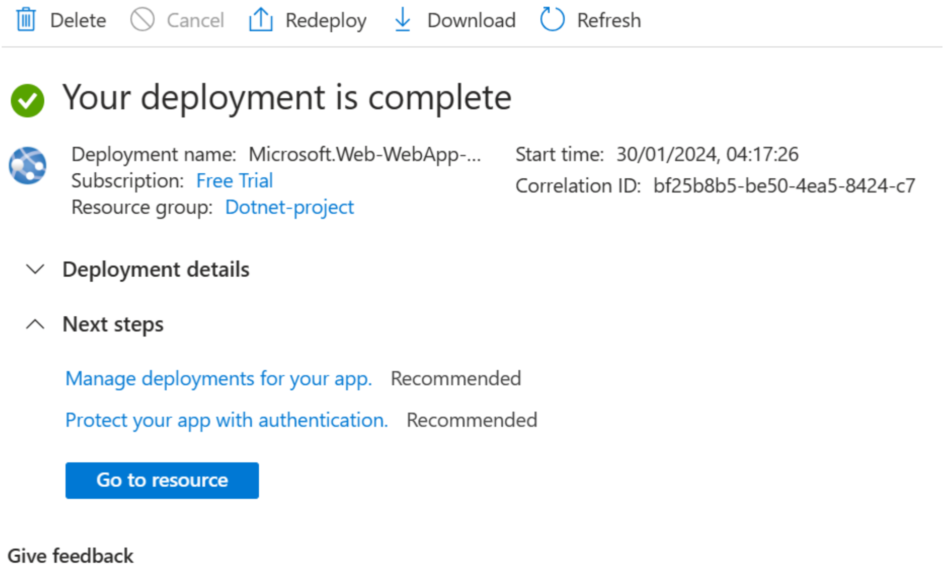
Pricing plan: Free F1 (Shared infrastructure)

Zone redundancy

An App Service plan can be deployed as a zone redundant service in the regions that support it. This is a deployment time only decision. You can't make an App Service plan zone redundant after it has been deployed Learn more

Zone redundancy:
○ Enabled: Your App Service plan and the apps in it will be zone redundant. The minimum App Service plan instance count will be three.
● Disabled: Your App Service Plan and the apps in it will not be zone redundant. The minimum App Service plan instance count will be one.

- As of now no changes in Database, Deployment, Networking, Monitoring & Tag
- Review & Create



Your deployment is complete

Deployment name: Microsoft.Web-WebApp-...
Subscription: Free Trial
Resource group: Dotnet-project

Start time: 30/01/2024, 04:17:26
Correlation ID: bf25b8b5-be50-4ea5-8424-c7

Deployment details

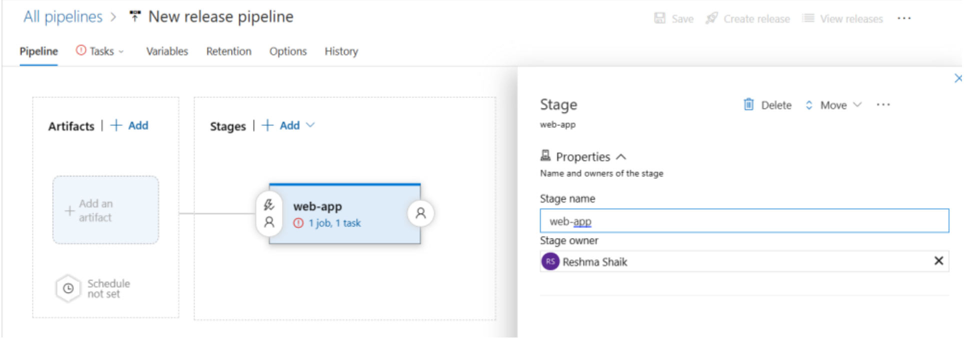
Next steps

Manage deployments for your app. Recommended

Protect your app with authentication. Recommended

Release Part

- Go to Pipelines
- Releases
- Select the right template ex: Azure app services deployment
- **Under Stage:** stage name: web-app



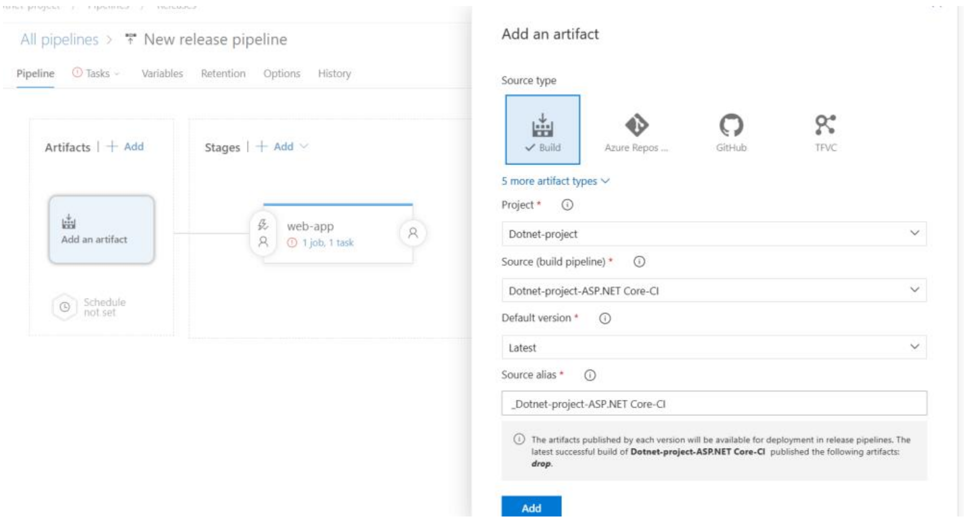
Stage
web-app

Properties
Name and owners of the stage

Stage name: web-app

Stage owner: Reshma Shaik

- Under Artifact



Add an artifact

Source type: Build

Project *: Dotnet-project

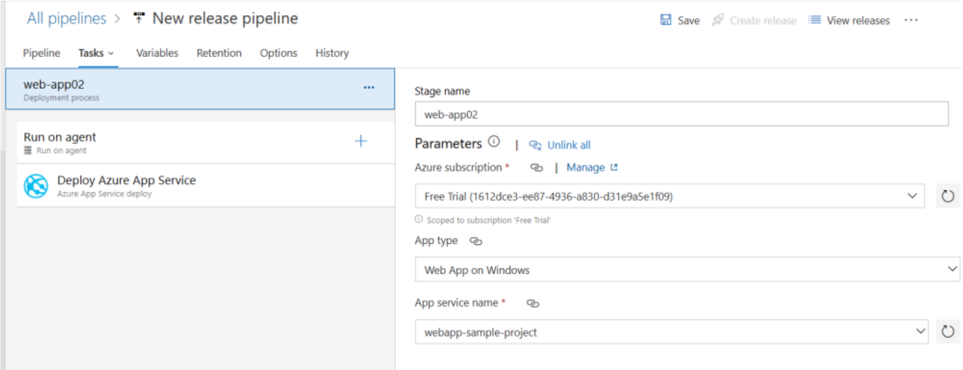
Source (build pipeline) *: Dotnet-project-ASP.NET Core-CI

Default version *: Latest

Source alias *: _Dotnet-project-ASP.NET Core-CI

The artifacts published by each version will be available for deployment in release pipelines. The latest successful build of **Dotnet-project-ASP.NET Core-CI** published the following artifacts: ***drop***.

Now again go to pipeline release under stage



Stage name: web-app02

Azure subscription *: Free Trial (1612dce3-ee87-4936-a830-d31e9a5e1f09)

Scoped to subscription 'Free Trial'

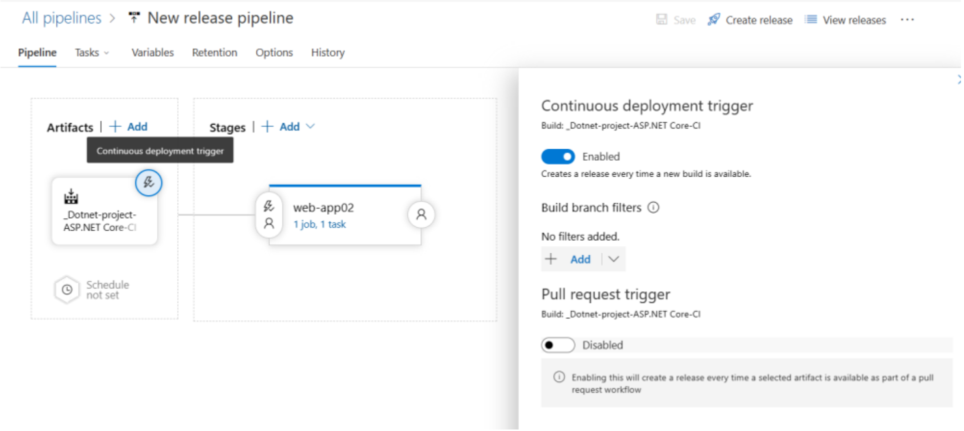
App type: Web App on Windows

App service name *: webapp-sample-project

Hit Save and create release

For Continuous integration (CI) go to pipelines – dotnet project – Edit – Trigger – Enable continuous integration (CI) - Save

For Continuous Deployment (CD) go to release – edit – continuous deployment trigger – enable – save



Continuous deployment trigger

Build: _Dotnet-project-ASP.NET Core-CI

● Enabled

Creates a release every time a new build is available.

Build branch filters

No filters added.

Pull request trigger

Build: _Dotnet-project-ASP.NET Core-CI

○ Disabled

Enabling this will create a release every time a selected artifact is available as part of a pull request workflow