

Azure pipelines

Where Azure Pipelines?

- Azure Pipelines are a service within Azure DevOps
- Azure DevOps is a SaaS platform from Microsoft that allows you to develop and deploy the software with all its various services.



What is Azure Pipelines?

- It is an Azure service within Azure DevOps that automatically builds, tests, and deploys code.
- It supports a ton of major languages, project types, and platforms.
- These pipelines can be done with templates or written with YAML.
- Its whats in charge of pushing your build artifact to wherever you need it to.

What does CI/CD stand for?

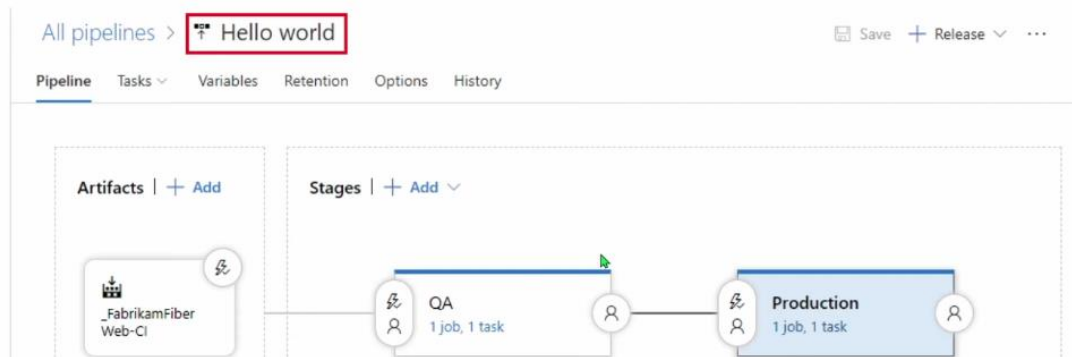
- CI/CD stands for Continuous Integration and Continuous Deployment!
- Continuous Integration means that developers in the team push their code to a repo in our case Azure git repo and it continuously merges the code they push, which then triggers a build which then tests the code.
- Continuous Deployment because as you push code, it will trigger a build and then push

that build artifact to one your environment that you have set up

Yaml file

- So there are two ways of building your pipeline, you can use the visual designer Devops gives you or you can use a YAML file which then can be managed just like any other file in your project.
- I chose to use YAML files for this video since it allows me to have full control as well as use it multiple times if i need to for some reason in the future, it's a big more scalable.
- A Step is the smallest building block of a pipeline and it executes a specific command.
- Writing the YAML isn't that bad since Devops gives you many Tasks which are basically templates you just have to fill in. (we will see this when we create them)

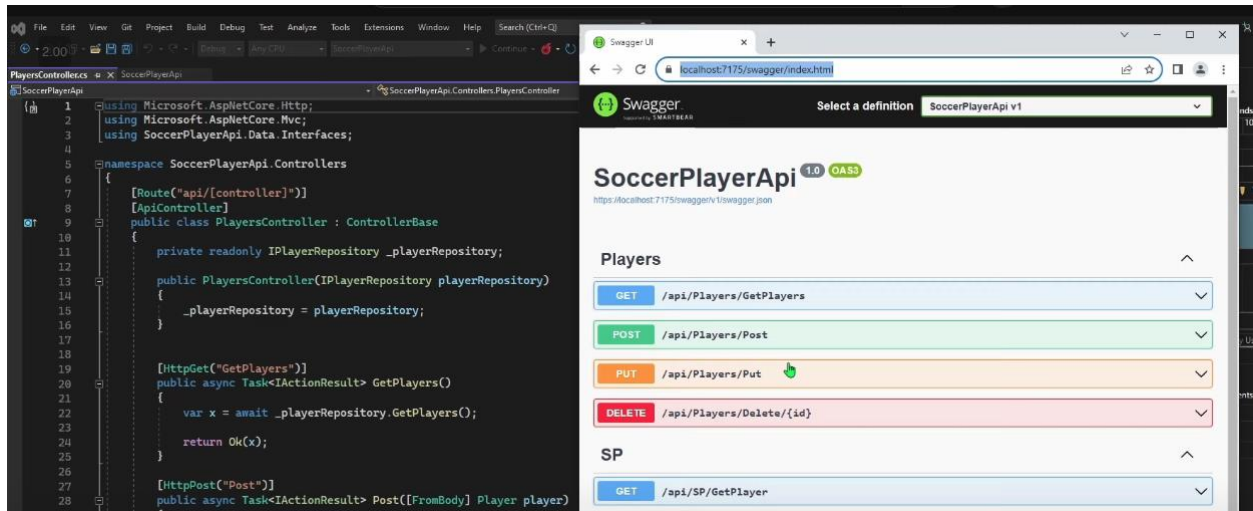
Visual Designer For Pipelines



```
azure-pipelines.yml  ! azure-pipelines-build-template.yml x  ! azure-pipelines-deployment-template.yml

1 parameters: # defaults for any parameters that aren't specified
2   - buildConfiguration: 'Release'
3   - buildPlatform: 'Any CPU'
4   - vmImage: 'vs2017-win2016'
5
6 jobs:
7   - job: Build
8     displayName: 'Build job'
9     pool:
10      vmImage: '${{parameters.vmImage}}'
11     steps:
12      - # DotNet Framework build items
13        task: NuGetToolInstaller@0
14        displayName: 'Use NuGet 4.4.1'
15        inputs:
16          versionSpec: 4.4.1
17
18      - task: NuGetCommand@2
19        displayName: 'NuGet restore'
20        inputs:
21          projects: |
22            SamLearnsAzure/SamLearnsAzure.Database/SamLearnsAzure.Database.sqlproj
23          vstsFeed: '030a758a-428f-4445-bce8-2c19ad9a56b3'
24
25      - task: VSBUILD@1
26        displayName: 'Build database project'
27        inputs:
28          platform: '${{parameters.buildPlatform}}'
29          configuration: '${{parameters.buildConfiguration}}'
30          solution: SamLearnsAzure/SamLearnsAzure.Database/SamLearnsAzure.Database.sqlproj
```

We want to push this website to production environment:



Unit test

```
2:00 PM - Debug - Any CPU - SoccerPlayerAPI - SoccerPlayerAPI - SoccerPlayerAPI
SoccerPlayerTests.cs x PlayersController.cs SoccerPlayerAPI
SoccerPlayerAPITests.SoccerPlayerTests
Get_WithExistingClients_ReturnsAllClients()

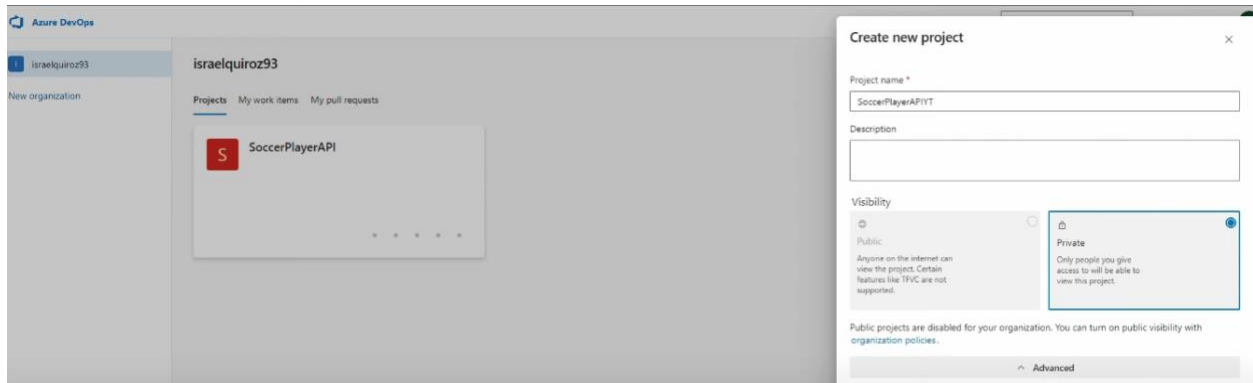
7 using System;
8
9 namespace SoccerPlayerAPITests
10 {
11     [TestClass]
12     public class SoccerPlayerTests
13     {
14         private readonly Mock<IPlayerRepository> _playerRepoMock = new Mock<IPlayerRepository>();
15         Random random = new Random();
16
17         [TestMethod]
18         public async Task Get_WithExistingClients_ReturnsAllClients()
19         {
20             //Arrange
21             var activelist = new List<Player>() { CreateRandomPlayer(), CreateRandomPlayer(), CreateRandomPlayer() };
22
23             _playerRepoMock.Setup(repo => repo.GetPlayers()).ReturnsAsync(activelist);
24
25             var controller = new PlayersController(_playerRepoMock.Object);
26
27             //Act
28
29             var actualPlayers = await controller.GetPlayers();
30             var players = actualPlayers as ObjectResult;
31
32             //Assert
33             players.Value.Should().BeEquivalentTo(
34                 activelist,
35                 options => options.ComparingByMembers<Player>()
36             );
37         }
38
39         private Player CreateRandomPlayer()
40     }
41 }
```

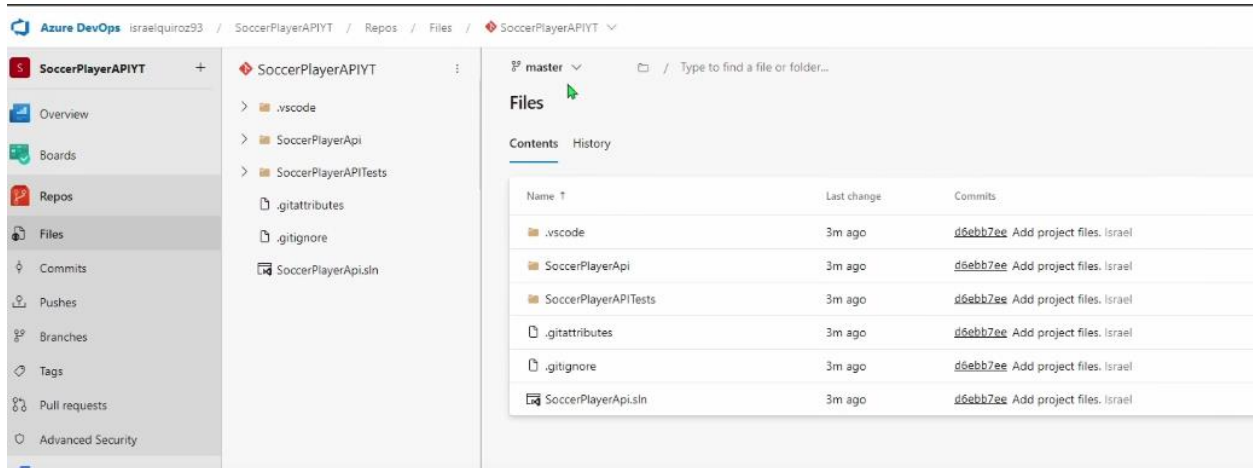
Output

Show output from: Debug

Unhandled exception. System.Exception: The thread 0x0 has exited with code 0 (0x0).

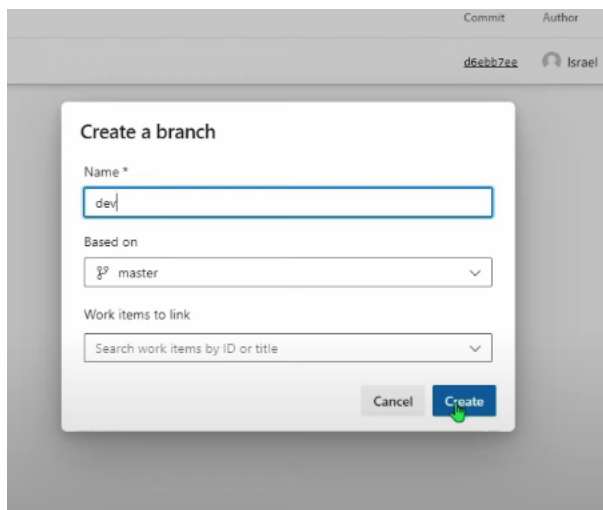
The program 'index.html' has exited with code 4294967295 (0xffffffff).



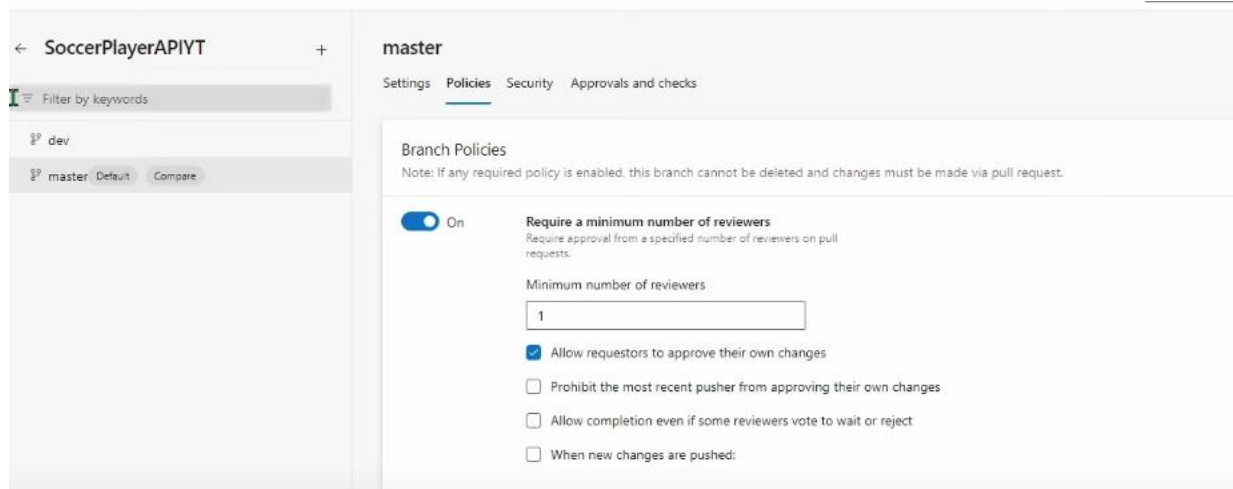


All code pushed

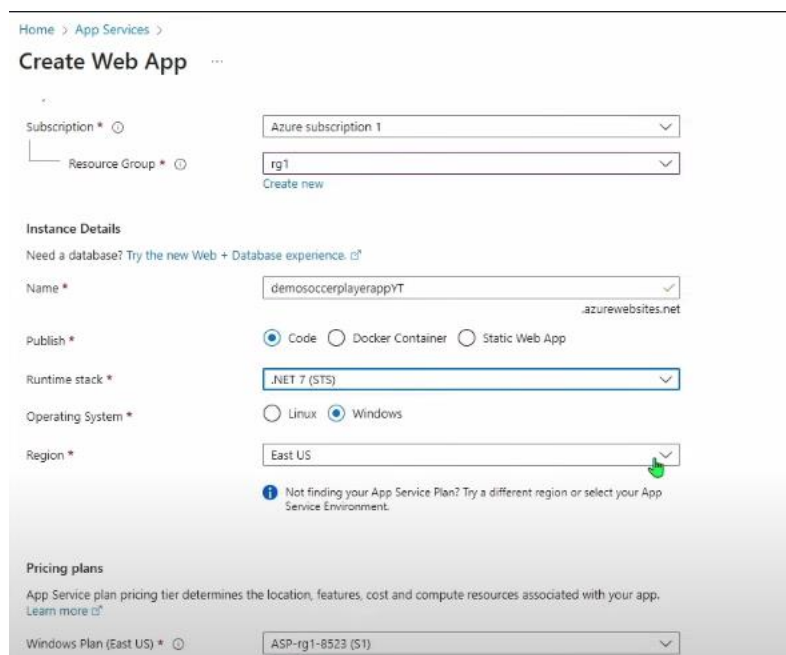
Create a dev branch :



Edit branch policies



Go to App services and create web app



Click in default domain

demosoccerplayerappYT Web App

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Microsoft Defender for Cloud

Events (preview)

Deployment

Deployment slots

Essentials

Resource group (move) : rg1

Status : Running

Location (move) : East US

Subscription (move) : Azure subscription 1

Subscription ID : 53ee3042-d92e-4074-a8b2-93cccf1b384b

Tags (edit) : Add tags

Default domain : demosoccerplayerappyt.azurewebsites.net

App Service Plan : myplan

Operating System : Windows

Health Check : Error fetching health check data. Please try again later.

Properties Monitoring Logs Capabilities Notifications Recommendations

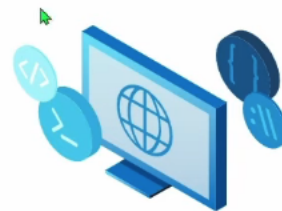
Web app

Webapp waiting for content:



Your web app is running and waiting for your content

Your web app is live, but we don't have your content yet. If you've already deployed, it could take up to 5 minutes for your content to show up, so come back soon.



Supporting Node.js, Java, .NET and more

Haven't deployed yet?
Use the deployment center to publish code or set up continuous deployment.

[Deployment center](#)

Starting a new web site?
Follow our Quickstart guide to get a web app ready quickly.

[Quickstart](#)

Put off

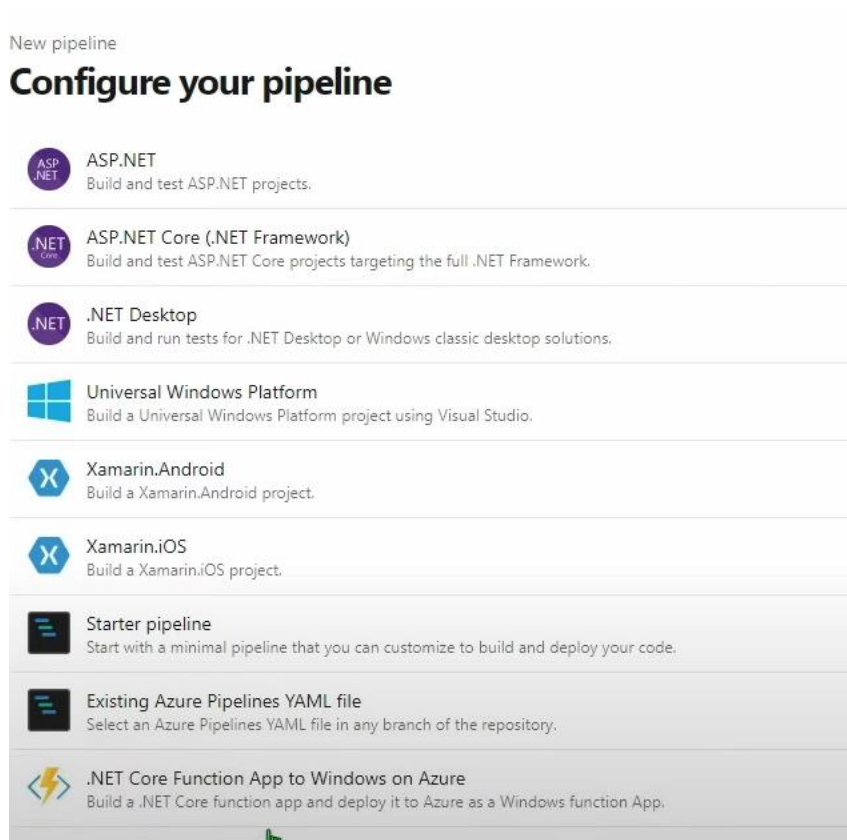
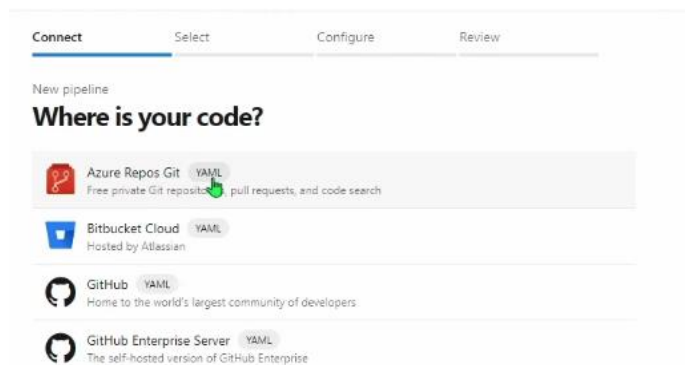
Branch Policies

Note: If any required policy is enabled, this branch cannot be deleted and changes must be made via pull request.

☐ Off

Require a minimum number of reviewers


Require approval from a specified number of reviewers on pull requests.



Starter pipeline:

New pipeline

Review your pipeline YAML

SoccerPlayerAPI / azure-pipelines-1.yml * 

```
1 # Starter pipeline
2 # Start with a minimal pipeline that you can customize to build and deploy your code.
3 # Add steps that build, run tests, deploy, and more:
4 # https://aka.ms/yaml
5
6 trigger:
7   - main
8
9 pool:
10   vmImage: ubuntu-latest
11
12 steps:
13   - script: echo Hello, world!
14     displayName: 'Run a one-line script'
15
16   - script: |
17     echo Add other tasks to build, test, and deploy your project.
18     echo See https://aka.ms/yaml
19     displayName: 'Run a multi-line script'
20
```

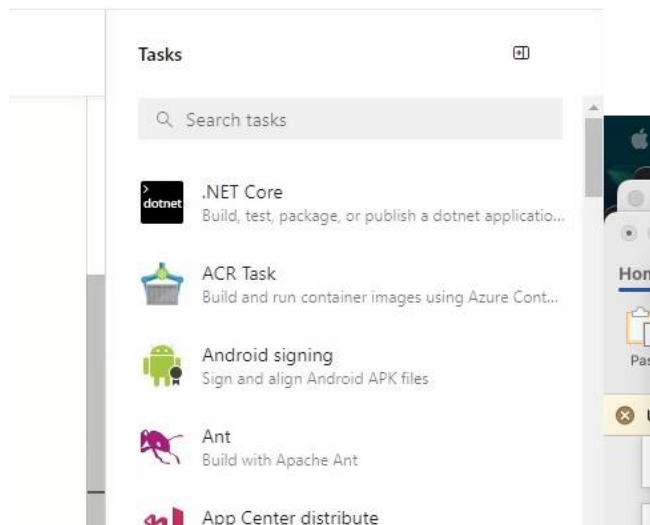
← SoccerPlayerAPI

main

SoccerPlayerAPI / azure-pipelines.yml

```
1 # ASP.NET Core
2 # Build and test ASP.NET Core projects targeting .NET Core.
3 # Add steps that run tests, create a NuGet package, deploy, and more:
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/dotnet-core
5
6 trigger:
7   branches:
8     include:
9       - dev
10
11 pool:
12   vmImage: 'windows-latest'
13
14 variables:
15   buildConfiguration: 'Release'
16   solution: '**/SoccerPlayerApi.sln'
17   project: '**/SoccerPlayerApi.csproj'
18   tests: '**/SoccerPlayerApiTests.csproj'
19
20
21 steps:
22   Settings
23   - task: NuGetToolInstaller@1
24     name: 'NuGetToolInstaller'
```

You can use templates:



main SoccerPlayerAPI / azure-pipelines.yml *

```
17 project: '**/SoccerPlayerApi.csproj'
18 tests: '**/SoccerPlayerApiTests.csproj'
19
20
21 Steps:
22   - task: NuGetToolInstaller@1
23     name: 'NuGetToolInstaller'
24
25   Settings:
26     - task: NuGetCommand@2
27       name: 'RestoreNuget'
28       inputs:
29         command: 'restore'
30         restoreSolution: '$(solution)'
```

Tasks

nuget

- Download GitHub NuGet Packages
Restore your nuget packages using dotnet CLI
- NuGet
Restore, pack, or push NuGet packages, or run
- NuGet authenticate
Configure NuGet tools to authenticate with A
- NuGet tool installer
Acquires a specific version of NuGet from the

Save and run

SoccerPlayerAPI / Pipelines

✓ Connect ✓ Select ✓ Configure Review

New pipeline

Review your pipeline YAML

SoccerPlayerAPI / azure-pipelines-1.yml *

```
1 # Starter pipeline
2 # Start with a minimal pipeline that you can customize to build and deploy your code.
3 # Add steps that build, run tests, deploy, and more:
4 # https://aka.ms/yaml
5
6 trigger:
7   - main
8
9 pool:
10   vmImage: ubuntu-latest
11
12 steps:
13   - script: echo Hello, world!
14     displayName: 'Run a one-line script'
15
16   - script: |
17     echo Add other tasks to build, test, and deploy your project.
18     echo See https://aka.ms/yaml
```

Save

Saving will commit azure-pipelines-1.yml to the repository.

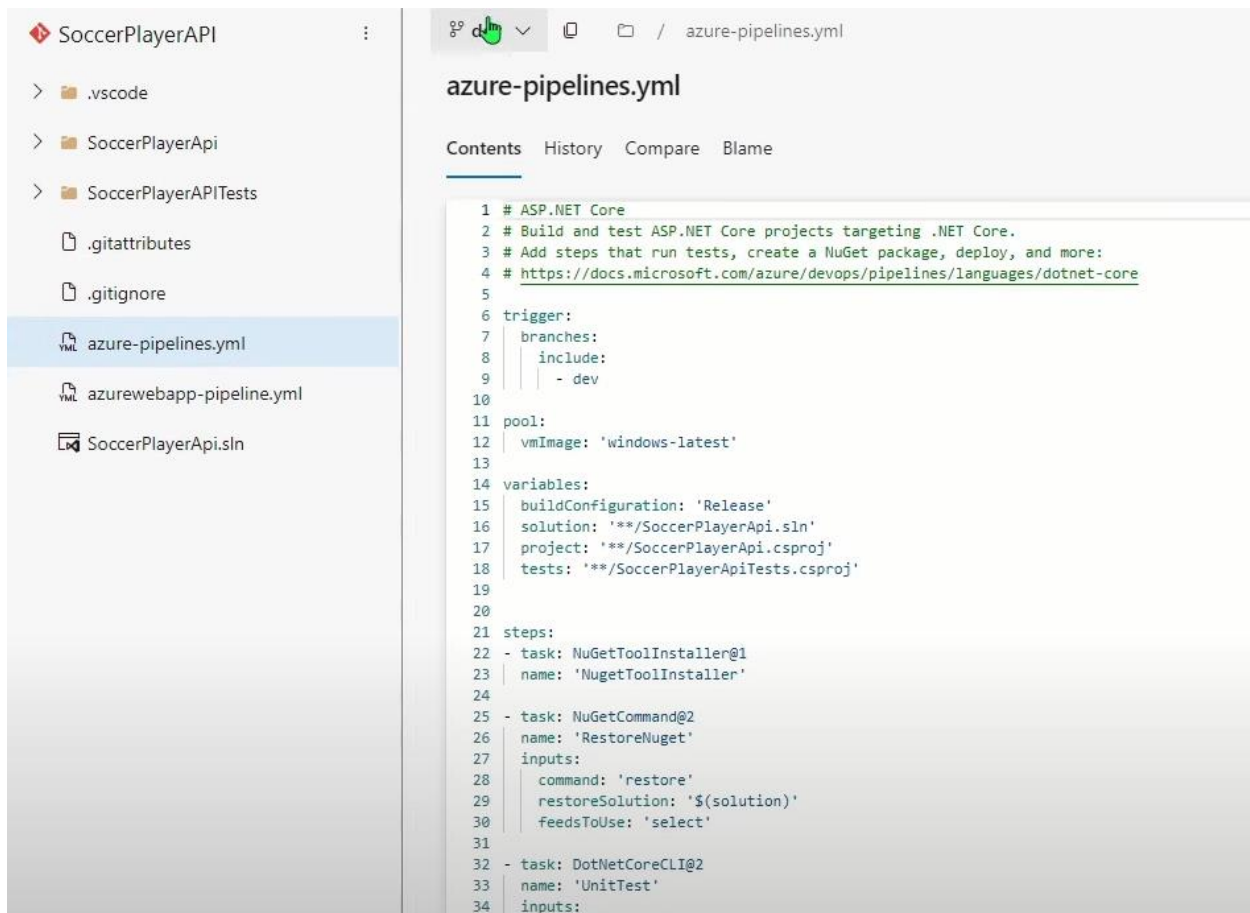
Commit message

Set up CI with Azure Pipelines

Optional extended description

Add an optional description...

- ☒ Commit directly to the main branch
- ☐ Create a new branch for this commit



Do a change, from GetAllPlayers to GetPlayers:

```

10 {
11     private readonly IPlayerRepository _playerRepository;
12
13     public PlayersController(IPlayerRepository playerRepository)
14     {
15         _playerRepository = playerRepository;
16     }
17
18
19     [HttpGet("GetPlayers")]
20     public async Task<IActionResult> GetPlayers()
21     {
22         var x = await _playerRepository.GetPlayers();
23
24         return Ok(x);
25     }
26
27     [HttpPost("Post")]
28     public async Task<IActionResult> Post([FromBody] Player player)
29     {
30         var players = await _playerRepository.CreatePlayer(player);
31         return Ok(players);
32     }
33
34
35     [HttpPut("Put")]
36     public async Task<IActionResult> Put([FromBody] Player player)
37     {
38         var players = await _playerRepository.EditPlayer(player);
39         return Ok(players);
40     }
41
42     [HttpDelete("Delete/{id}")]

```

Push it

Pipelines

Recent All Runs

New pipeline

Filter pipelines

Recently run pipelines

Pipeline	Last run
 SoccerPlayerAPI	#20231025.2 • update Individual CI for 3P dev Just now
 SoccerPlayerAPI (1)	#20231025.1 • Merged PR 12: update Individual CI for 3P main Yesterday 37s

SoccerPlayerAPI

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Pipelines

RecentAllRuns

New pipeline

Filter pipelines

Recently run pipelines

Pipeline	Last run	
SoccerPlayerAPI	#20231025.2 • update Individual CI for dev	Just now
SoccerPlayerAPI (1)	#20231025.1 • Merged PR 12: update Individual CI for main	Yesterday 37s

Jobs in run #20231025.2

SoccerPlayerAPI

obs

Job

1s

Initialize job

1s

Initialize job

1 Starting: Initialize job

2 Agent name: 'Hosted Agent'

3 Agent machine name: 'fv-az129-898'

4 Current agent version: '3.227.2'

5 Operating System

6 Runner Image

7 Runner Image Provisioner

8 Current image version: '20231023.1.0'

9 Agent running as: 'VssAdministrator'

10 Prepare build directory.

11 Set build variables.

12 Download all required tasks.

13 Downloading task: NuGetToolInstaller (1.221.0)

14 Downloading task: NuGetCommand (2.227.0)

Pipelines

RecentAllRuns

Recently run pipelines

Pipeline	Last run
SoccerPlayerAPI	#20231025.2 • update Individual CI for dev
SoccerPlayerAPI (1)	#20231025.1 • Merged PR 12: update Individual CI for main

←

Jobs in run #20231025.2

SoccerPlayerAPI

Jobs

Job	Duration
✓ Initialize job	2m 1s
✓ Checkout SoccerPlayerAPI@dev t...	8s
✓ NugetToolInstaller	5s
✓ RestoreNuget	4s
✓ UnitTest	56s
✓ buildArtifact	34s
✓ publishArtifact	4s
✓ Post-job: Checkout SoccerPlayer...	5s
✓ Finalize Job	<1s
✓ Report build status	<1s

✓

UnitTest

```

6 Author      : Microsoft Corporation
7 Help        : https://docs.microsoft.com/azure/devops/pipelines/tasks/build/dotnet-core-cli
8
9 C:\Windows\system32\chcp.com 65001
10 Active code page: 65001
11 Info: .NET Core SDK/runtime 2.2 and 3.0 are now End of Life(EOL) and have been removed from
12 "C:\Program Files\dotnet\dotnet.exe" test D:\a\1\s\SoccerPlayerAPITests\SoccerPlayerAPITests
13 Determining projects to restore...
14 Restored D:\a\1\s\SoccerPlayerApi\SoccerPlayerApi.csproj (in 641 ms).
15 Restored D:\a\1\s\SoccerPlayerAPITests\SoccerPlayerAPITests.csproj (in 641 ms).
16 D:\a\1\s\SoccerPlayerApi\Models\SampleDatabaseContext.cs(26,10): warning CS1030: #warning: '
17 D:\a\1\s\SoccerPlayerApi\Data\Repositories\SPRepository.cs(31,20): warning CS8603: Possible
18 D:\a\1\s\SoccerPlayerApi\Data\Repositories\PlayerRepository.cs(24,45): warning CS8604: Possi
19 D:\a\1\s\SoccerPlayerApi\Data\Repositories\PlayerRepository.cs(32,13): warning CS8602: Deref
20 SoccerPlayerApi -> D:\a\1\s\SoccerPlayerApi\bin\Release\net7.0\SoccerPlayerApi.dll
21 D:\a\1\s\SoccerPlayerAPITests\SoccerPlayerTests.cs(34,13): warning CS8602: Dereference of a
22 SoccerPlayerAPITests -> D:\a\1\s\SoccerPlayerAPITests\bin\Release\net7.0\SoccerPlayerAPITe
23 Test run for D:\a\1\s\SoccerPlayerAPITests\bin\Release\net7.0\SoccerPlayerAPITests.dll (.NET
24 Microsoft (R) Test Execution Command Line Tool Version 17.7.1 (x64)
25 Copyright (c) Microsoft Corporation. All rights reserved.
26
27 Starting test execution, please wait...
28 A total of 1 test files matched the specified pattern.
29 Results File: D:\a\_temp\VssAdministrator_fv-az129-898_2023-10-25_05_35_26.trx
30
31 Passed! - Failed: 0, Passed: 1, Skipped: 0, Total: 1, Duration: 380 ms - So
32 Result Attachments will be stored in LogStore
33 Run Attachments will be stored in LogStore
34 Info: Azure Pipelines hosted agents have been updated and now contain .Net 5.x SDK/Runtime a

```

1 artifact created:

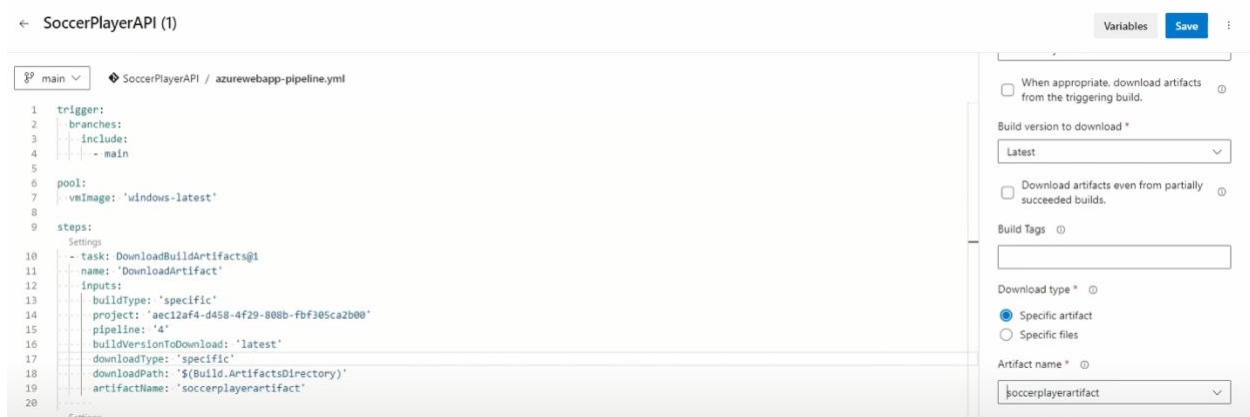
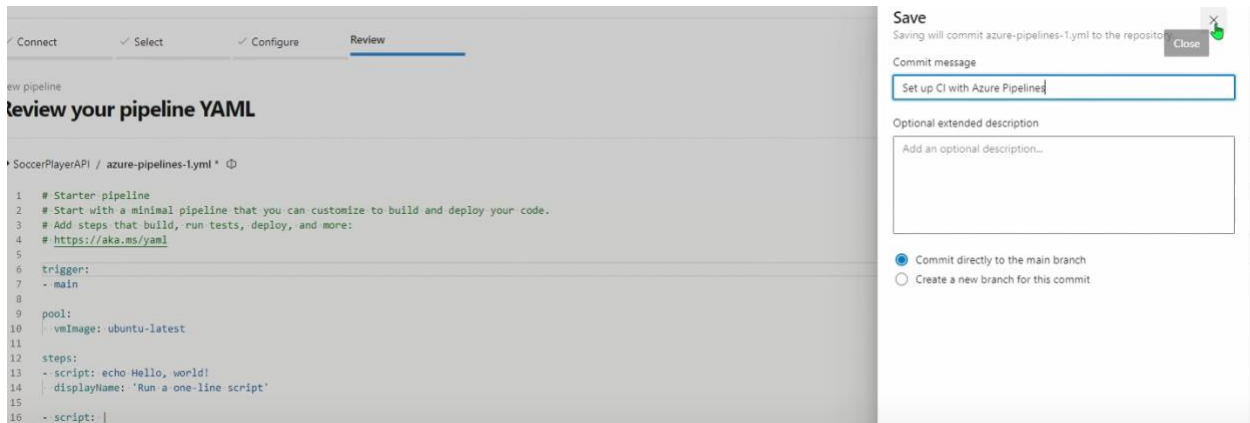
←

Artifacts

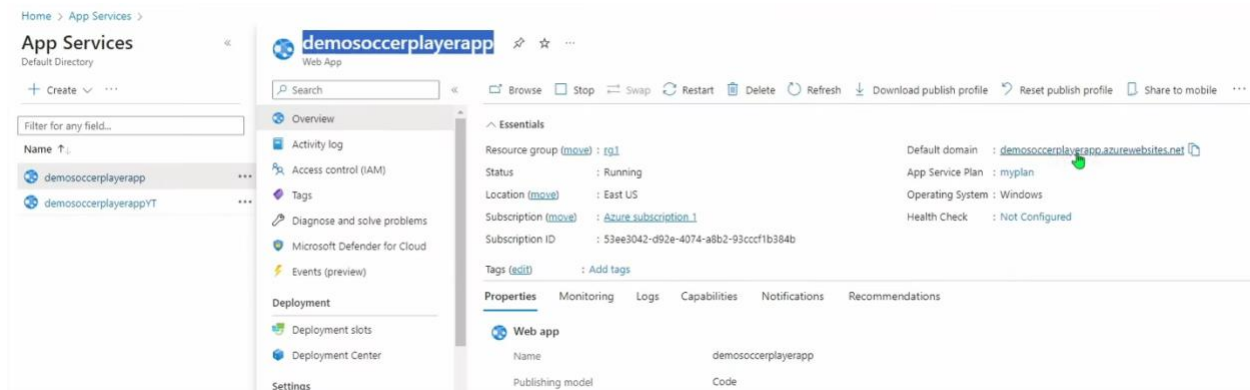
Published

Name
<div> <div>✓</div> <div>soccerplayerartifact</div> </div> <div> <div>📄</div> <div>SoccerPlayerApi.zip</div> </div>

Create a new pipeline:



Click in default domain and we see the app:



SoccerPlayerApi 1.0 OAuth2

<https://demosoccerplayerapp.azurewebsites.net/swagger/v1/swagger.json>

Players

GET	/api/Players/GetAllPlayers	✓
POST	/api/Players/Post	✓
PUT	/api/Players/Put	✓
DELETE	/api/Players/Delete/{id}	✓