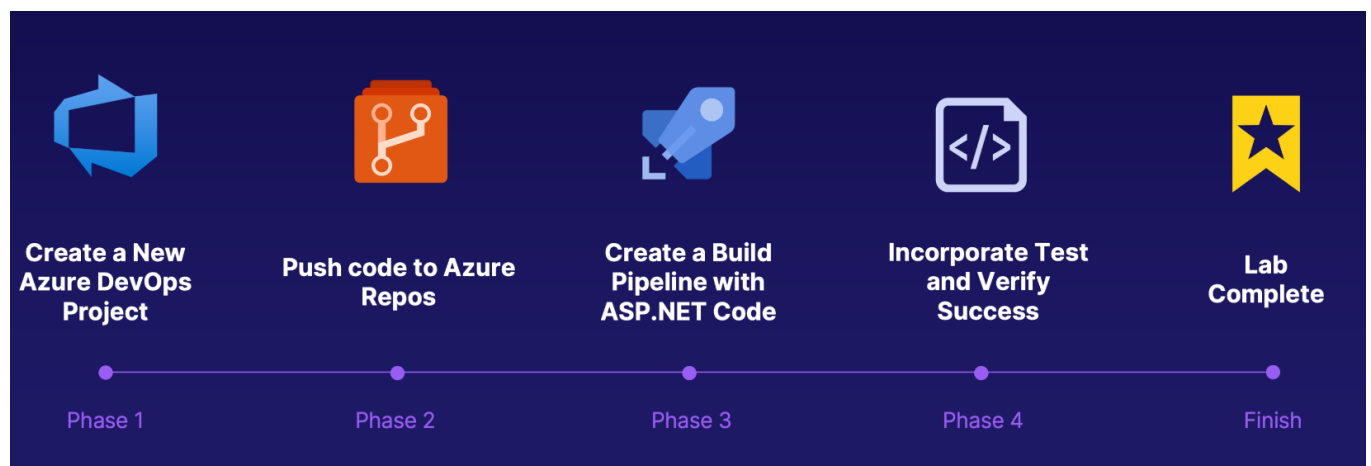


Create and Test an ASP.NET Core App in Azure Pipelines

Introduction

- You have a .NET Core application (with tests) that you need to push to Azure Repos, then build and test. You will be provided with the code, with which you must use Azure DevOps to complete these tasks.
- Log in to the Azure portal, and go to Azure DevOps Organizations from the Services menu.
- Continue to create a new DevOps organization as well as a new project.
- Use the following code from the master branch to build your pipeline:
 - <https://github.com/hosniah/azuredevops-aspnetcorewebapp.git> .
- Create an Azure DevOps Pipeline with the .NET Core template (in the classic editor).
- Change the location of the test .csproj files, add continuous integration, and run the build by pushing changes to the master branch.



1- Create a DevOps Project and Push Code to Azure Repos

1. In the Azure portal, click the hamburger menu in the upper left corner.
2. Click on All services.
3. In the search bar, search for "Azure DevOps". Select Azure DevOps organizations.
4. Scroll down and click on My Azure DevOps Organizations.
5. Select your country/region from the dropdown list.
6. Click on Continue.
7. Select Create new organization > Continue > Continue.
8. Set the following values to create a project:
 - a. Project name: MyFirstProject
 - b. Visibility: Private
9. Click + Create project.
10. In the Linux VM, pull code from the master branch using the URL in the lab instructions:

```
git clone https://github.com/hosniah/azuredevops-aspnetcorewebapp.git
```
11. List the contents of the directory:

```
ls
```
12. Change into the directory shown:

```
cd azuredevops-aspnetcorewebapp/
```
13. List the contents:

```
ls
```
14. View the remote origin:

```
git remote -v
```
15. Remove the remote origin:

```
git remote remove origin
```
16. In Azure DevOps, click on Repos in the left-hand navigation menu.
17. Under Push an existing repository from the command line, copy the commands.
18. Return to cloudhsell and paste in the commands.

19. Back in Azure DevOps, in the upper right-hand corner, click the User settings icon > Personal access tokens.
20. Click on + New Token and set the following values:
Name: gtatoken
Scopes: Full access
21. Click Create.
22. Copy the 'gtatoken' token and save it in a safe location.
23. Click Close.
24. Return to cloudshell and paste in the access token.
25. In Azure DevOps, click on Azure DevOps in the upper left corner.
26. Select your project.
27. Click on Repos in the left-hand navigation menu. Verify that the code was pushed to Azure Repos.

2. Create a New Azure DevOps Pipeline

1. Click on Pipelines in the left-hand navigation menu.
2. Click on Create pipeline.
3. Click on Use the classic editor.
4. Under Select a source, click on Azure Repos Git and make the following selections:
 - a. Team project: MyFirstProject
 - b. Repository: MyFirstProject
 - c. Default branch: master
5. Click on Continue.
6. Scroll down and select the ASP.NET Core template.
7. Click on Apply and set the following value:
 - a. Agent Specification: ubuntu-18.04
 - b. Project(s) to test: `**/*[Tt]est/*.csproj`
8. Click on Save & queue > Save.
9. Click on Triggers and check the box for Enable continuous integration .
10. Click on Save & queue > Save.

11. In the Linux VM, go into the web app:
 - a. `cd.AspNetCoreWebApplication`
12. Change into Views:
 - a. `cd Views/`
13. Change into Home:
 - a. `cd Home`
14. List the contents:
 - a. `ls`
15. Open Index.cshtml:
 - a. `vi Index.cshtml`
16. Make a minor change in the `<h1>` heading. After "Congratulations!", add "You Did It!"
 - a. `<h1> Congratulations! You Did It!</h1>`
17. Press Esc and type `:wq` to save and quit.
18. Commit the changes:
 - a. `git status` `git add .` `git commit -m "changed Index.cshtml"`
19. Push the changes:
 - a. `git push origin master`
20. Paste in the personal access token.
21. In Azure DevOps, click on Pipelines in the left-hand navigation menu.
22. Click on the pipeline shown to open it. Then click into it again.
23. Select Agent job 1.
24. Click on the back arrow.
25. Click on Tests to verify success.