1. As part of environment prep, we create ADO environment objects namely “QA” and “Prod” to use them for approvals. We also create a service connection to Azure Repos.
2. the trigger for master to ensure build kicks in immediately there is a commit
3. Directory structure: We have a pipeline for Continuous Integration and one pipeline for Continuous Deployment at the top tier. We have two directories, one for templates that consists of build and deployment templates. Another directory that holds the variables files.

CI\_pipeline.yml

CD\_pipeline.yml

/templates/testAndPublishTemplate.yml

/buildTemplate.yml

/webapi\_deploy\_template.yml

/webApp\_deploy\_template.yml

/variables/variables.dev.yml

/variables.qa.yml

/variables.prod.yml

1. CI\_pipeline.yml has two stages which are to “build” which calls “buildTemplate.yml” and “test and publish” which calls “testAndPublishTemplate.yml” the artifact.
2. buildTemplate.yml performs two ADO native .net tasks. In the first task, we perform NuGet restore to ensure all dependencies are present. While on the second, we build all 3 projects
3. testAndPublishTemplate.yml consists of 3 tasks where the first one tests the projects based on the status of build success of all 3 projects. Next 2 tasks publish the artifacts of WebApp and WebApp. All 3 tasks make use of ADO tasks.
4. CD\_pipeline.yml has 3 stages. One each of Dev, QA and Prod. QA and Prod have environment objects configured namely “QA” and “Prod” as a prerequisite.
5. We use “webApp\_deploy\_template.yml” and “webapi\_deploy\_template.yml” in each Stage to deploy the apps. We ensure variable definition files for each environment that consists of env specific service connections.