



Azure DevOps

PALESTRA III

AZURE DEVOPS – CI/CD

NÁDIO DIB – ENG. SOFTWARE

UNIPROJEÇÃO

SUMÁRIO



Azure DevOps

- Sobre o Azure Pipelines e como é configurado?
- Azure Pipelines triggers
- Azure Pipelines CI/CD
- Web Apps: Azure Resource Group & Plan
- Web Apps: implantação no Azure Services (deployment)



SOBRE O AZURE PIPELINES E COMO É CONFIGURADO?

SOBRE O AZURE PIPELINES



- É uma rotina de processos em cascata no qual executa procedimentos, pré-configurados pelo desenvolvedor, em uma ou mais jobs em paralelo por uma máquina virtual;
- São realizados procedimentos como: **build, dependencies restoring, unit test, execução de scripts terminal, geração de artefatos** etc.; e
- É possível verificar: **falhas em builds pelo log gerado, notificações de aviso sobre vulnerabilidades** etc.;



CONFIGURAÇÃO AZURE-PIPELINES.YML

Através do arquivo **azure-pipelines.yml** é possível configurar as predefinições dos procedimentos que serão submetidas durante as jobs dentro do Azure Pipelines.

```
1 trigger:
2   - production
3   - staging
4   - development
5
6 pool:
7   - vmImage: 'windows-latest'
8
9 variables:
10  - solution: '**/*.sln'
11  - buildPlatform: 'Any-CPU'
12  - buildConfiguration: 'Release'
13
14 steps:
15   Settings
16   - task: NuGetToolInstaller@1
17
18   Settings
19   - task: NuGetCommand@2
20     inputs:
21       - restoreSolution: '$(solution)'
22
23   Settings
24   - task: VSBUILD@1
25     inputs:
26       - solution: '$(solution)'
27       - platform: '$(buildPlatform)'
28       - configuration: '$(buildConfiguration)'
29
30   Settings
31   - task: ArchiveFiles@2
32     inputs:
33       - rootFolderOrFile: '$(Build.BinariesDirectory)'
34       - includeRootFolder: true
35       - archiveType: 'zip'
36       - archiveFile: '$(Build.ArtifactStagingDirectory)/TKDevOps.AppEngineAPI.Artifact-$(Build.BuildId).zip'
37       - replaceExistingArchive: true
```





















AZURE PIPELINES TRIGGERS

AZURE PIPELINES TRIGGERS



- Triggers são gatilhos nos quais executam procedimentos quando é satisfeito um ou mais condições;
- Condições podem ser, por exemplo: **push / pull de commits / branches em um repositório**; e
- Quando condições de um trigger são satisfeitas, operações de deployment podem ser ativadas e automatizadas.

Description	Stages	
#20201008.1 Tweaks Individual CI for  development  efb6383 		 Oct 8  54s
#20201007.2 Merge branch 'development' of https://github.com/TK-Games/tk-netm-api into development Individual CI for  development  0d4b305 		 Oct 7  58s
#20201007.1 Update README.md Individual CI for  development  414a2d8 		 Oct 6  53s

TRIGGERS

Exemplo de como os triggers se comportam quando condições são satisfeitas.





AZURE PIPELINES CI/CD

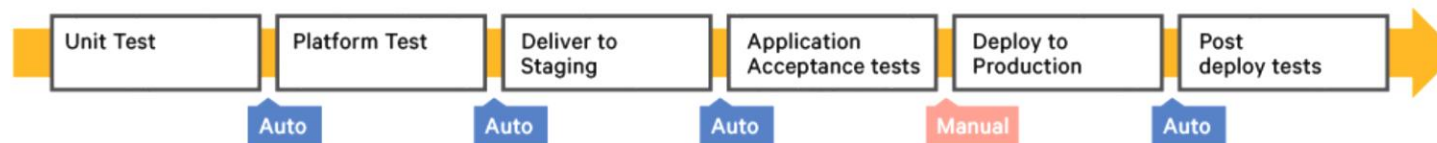
AZURE PIPELINES CI/CD



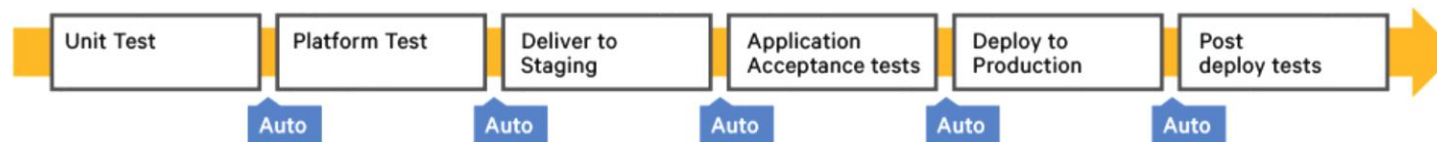
- **CI – Continuous Integration**: também conhecido como **integração contínua**;
- **CD – Continuous Deployment**: também conhecido como **implantação contínua**; e
- Não confundir Continuous Deployment com Continuous Delivery (entrega contínua).



Continuous Delivery



Continuous Deployment



CONTINUOUS DELIVERY VS. CONTINUOUS DEPLOYMENT

Continuous Delivery: implantação manual

Continuous Deployment: implantação automatizada



AZURE PIPELINES CI

As operações executadas pela(s) job(s) geram um log que contém informações sobre o status daquela build, podendo ser:

succeeded ou **failed**.

← Jobs in run #20201008.1
TK-Games.tk-netm-api

Jobs

✓ Job	47s
✓ Initialize job	9s
✓ Checkout TK-Games/tk...	5s
✓ NuGetToolInstaller	1s
✓ NuGetCommand	17s
✓ VSBUILD	11s
✓ ArchiveFiles	<1s
✓ Post-job: Checkout TK...	<1s
✓ Finalize Job	<1s

✓ Job

```
1 Pool: Azure Pipelines
2 Image: windows-latest
3 Agent: Hosted Agent
4 Started: Oct 8 at 6:12 PM
5 Duration: 47s
6
7 ▼ Job preparation parameters
8 ContinueOnError: False
9 TimeoutInMinutes: 60
10 CancelTimeoutInMinutes: 5
11 Expand:
12   MaxConcurrency: 0
13   ##### System Pipeline Decorator(s) #####
14
15   Begin evaluating template 'system-pre-steps.yml'
16 Evaluating: eq('true', variables['system.debugContext'])
17 Expanded: eq('true', Null)
18 Result: False
19 Evaluating: resources['repositories']['self']
20 Expanded: Object
21 Result: True
22 Evaluating: not(containsValue(job['steps']['*']['task']['id'], '6d15af64-176c-496d-b583-fd2ae21d4df4'))
23 Expanded: not(containsValue(Object, '6d15af64-176c-496d-b583-fd2ae21d4df4'))
24 Result: True
25 Evaluating: resources['repositories']['self']['checkoutOptions']
26 Result: Object
27 Finished evaluating template 'system-pre-steps.yml'
28 #####
29 Template and static variable resolution complete. Final runtime YAML document:
30 steps:
31 - task: 6d15af64-176c-496d-b583-fd2ae21d4df4@1
32   inputs:
33     repository: self
34
35
36   MaxConcurrency: 0
```



WEB APPS: AZURE RESOURCE GROUP & PLAN

WEB APPS: AZURE RESOURCE GROUP & PLAN



- **Azure Resource Group** – é um container que contém metadados necessários para rodar a aplicação através do artefato gerado, cujo principal papel é reduzir a complexidade de orquestração da distribuição do serviço; e
- **Azure Resource Plan** – os Resource Groups devem ser atribuídos a um plano de inscrição vinculado a conta, onde podem ser: **por reserva** ou **on-demand**.

DÚVIDAS?



Azure DevOps



NÁDIO DIB

Contatos



OBRIGADO!



Azure DevOps