**Please find the answers in Blue**

**Q1 - SCENARIO**

A car rental company called FastCarz has a .net Web Application and Web API which are recently migrated from on-premise system to Azure cloud using Azure Web App Service

and Web API Service.

The on-premises system had 3 environments Dev, QA and Prod.

The code repository was maintained in TFS and moved to Azure GIT now. The TFS has daily builds which triggers every night which build the solution and copy the build package to drop folder.

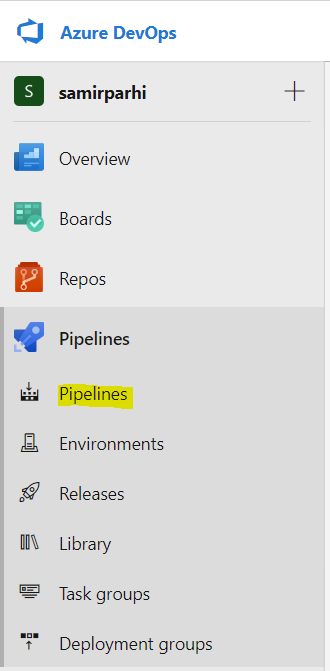
deployments were done to the respective environment manually. The customer is planning to setup Azure DevOps service for below requirements:

1. *The build should trigger as soon as anyone in the dev team checks in code to master branch.*

*Solution: (From GUI)*

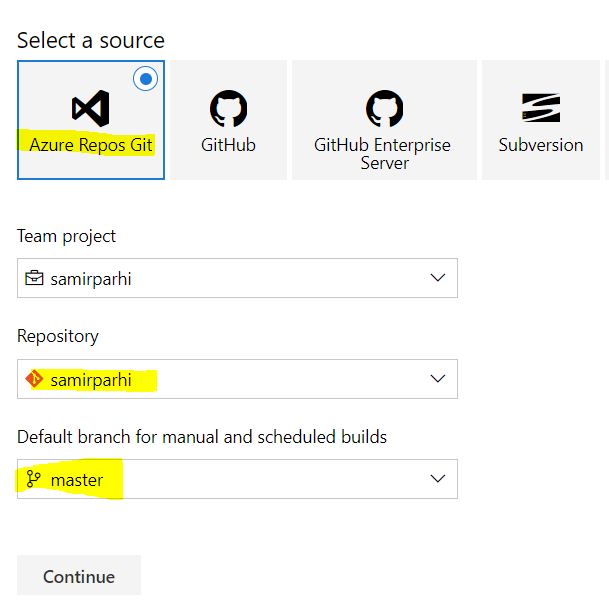
*Create a pipeline if not already there:*

*Step1: Go to Azure devOps and click on pipeline .*



*Now click on create*

*Step 2: provide the source code repository type :*



*Step 3: select the Default branch as master. When you give this as master, as soon as there ia any commit happens to the master branch the pipeline will trigger.*

*Through YAML:*

***trigger:***

***- master***

*When you specify master under the trigger section of YAML configuration file it will take that as a branch to trigger a pipeline*

1. *There will be test projects which will create and maintained in the solution along the Web and API. The trigger should build all the 3 projects - Web, API and test.*

*The build should not be successful if any test fails.*

*Through YAML:*

***stages:***

***- stage: Web***

***dependsOn: test***

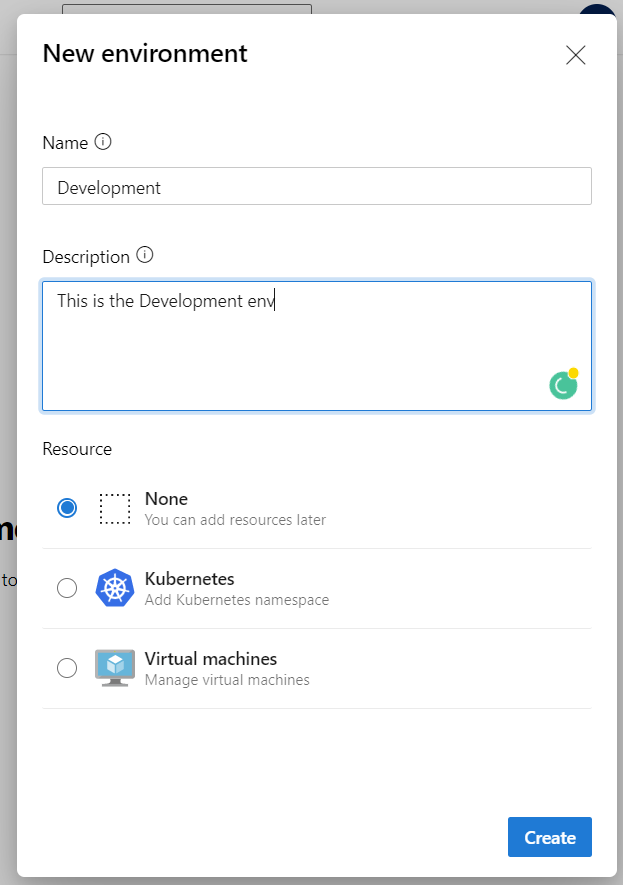
***- stage: API***

***dependsOn: test***

***- stage: test***

1. *The deployment of code and artifacts should be automated to Dev environment.*

*Step-1: Create an environment in the azure pipeline, if not there as follows (Choose virtual Machine or Kubernetes) :*



*Step-2: Then run the registration script*

Example (for linux hosts) : mkdir azagent;cd azagent;curl -fkSL -o vstsagent.tar.gz https://vstsagentpackage.azureedge.net/agent/2.174.1/vsts-agent-linux-x64-2.174.1.tar.gz;tar -zxvf vstsagent.tar.gz; if [ -x "$(command -v systemctl)" ]; then ./config.sh --environment --environmentname "DEV" --acceptteeeula --agent $HOSTNAME --url https://dev.azure.com/samirparhi/ --work \_work --projectname 'samirparhi' --auth PAT --token qauzg53nmqo4jtmxzupeeqplab6uk5p3ng5qkkypdcr34lywfuva --runasservice; sudo ./svc.sh install; sudo ./svc.sh start; else ./config.sh --environment --environmentname "DEV" --acceptteeeula --agent $HOSTNAME --url https://dev.azure.com/samirparhi/ --work \_work --projectname 'samirparhi' --auth PAT --token qauzg53nmqo4jtmxzupeeqplab6uk5p3ng5qkkypdcr34lywfuva; ./run.sh; fi

*Step-3: Once VM is registered, it will start appearing as an environment resource*

*Step -4: Deploy through YAML:*

stages:

- stage: Web

dependsOn: test

- stage: API

dependsOn: test

- stage: test

- stage: Dev deployment

jobs:

- deployment: Web, API, Test Deployment

displayName: Web, API, Test Deployment

environment:

name: Development

resourceType: VirtualMachine

1. *Upon successful deployment to the Dev environment, deployment should be easily promoted to QA and Prod through automated process.*

- stage: Test deployment

dependsOn: Dev deployment

jobs:

- deployment: Web, API, Test Deployment

displayName: Web, API, Test Deployment

environment:

name: QA

resourceType: VirtualMachine

- stage: Prod deployment

dependsOn: Test deployment

jobs:

- deployment: Web, API, Test Deployment

displayName: Web, API, Test Deployment

environment:

name: Prod

resourceType: VirtualMachine

1. *The deployments to QA and Prod should be enabled with Approvals from approvers only.*

*You have to define the deployment approval in the tool , providing the required approver . so that it can be progressed through various stages once the approval is provided in the Azure devOps tool*