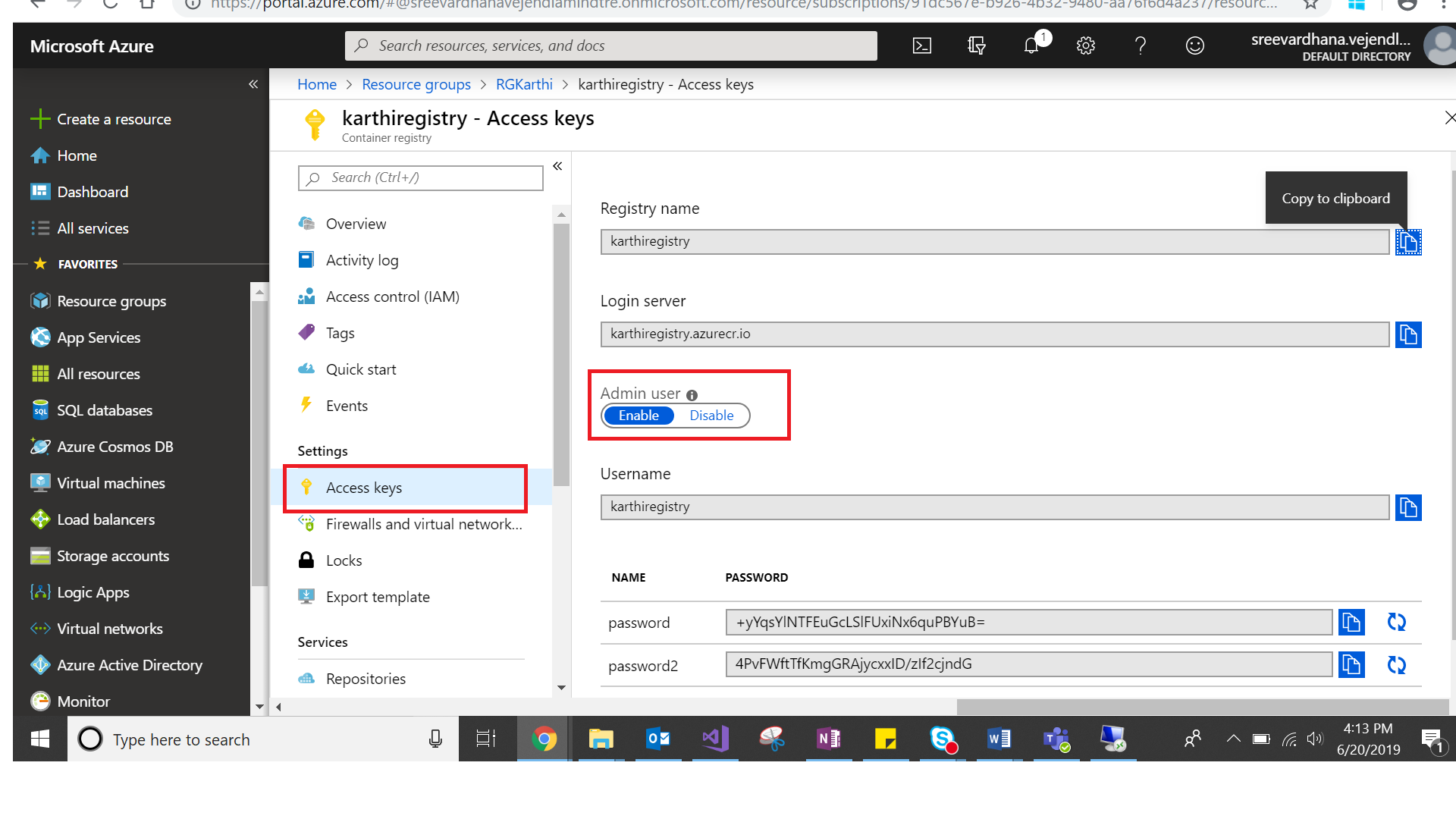
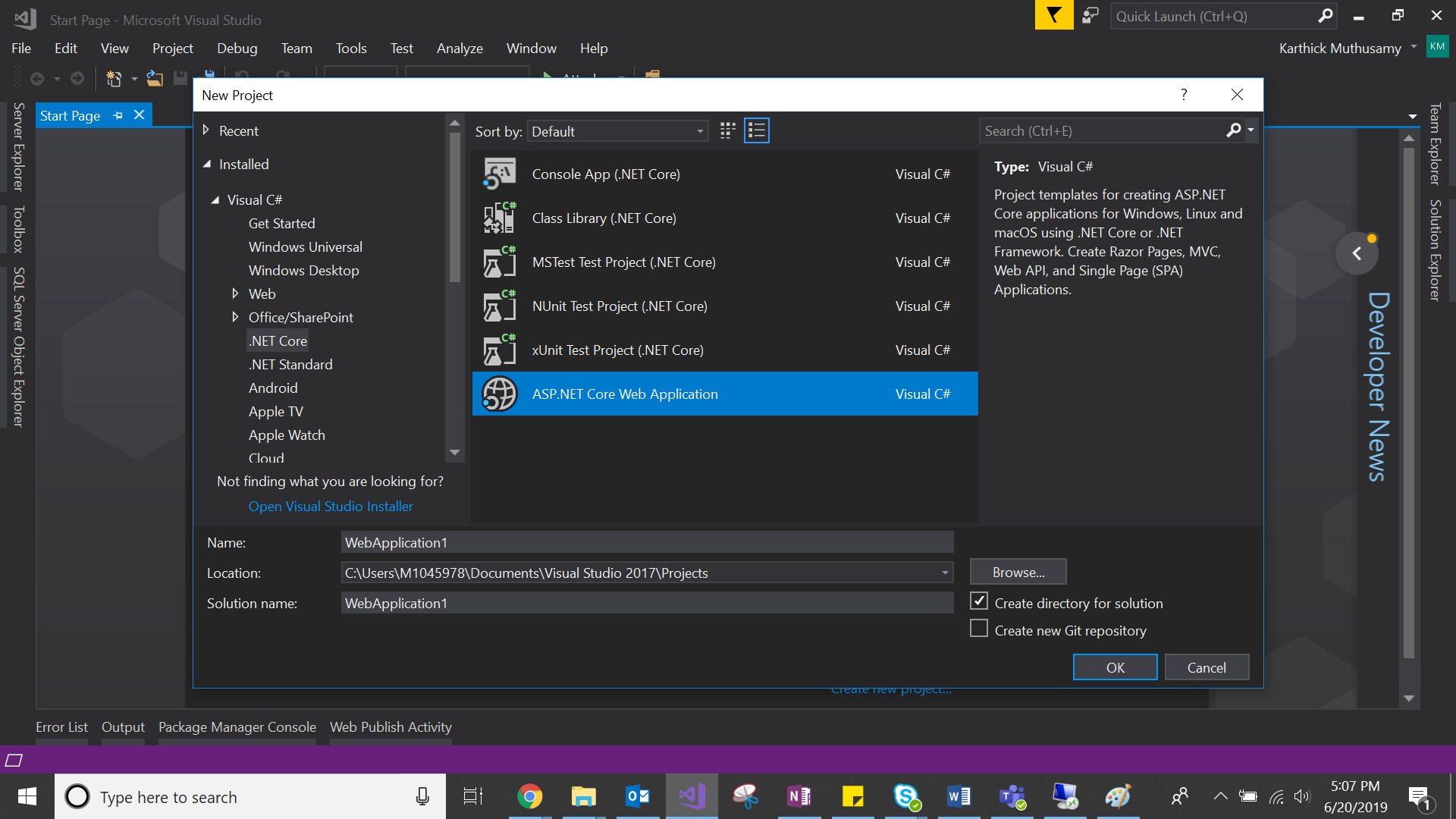
1. Create container registry in portal
2. Create Kubernetes services in portal

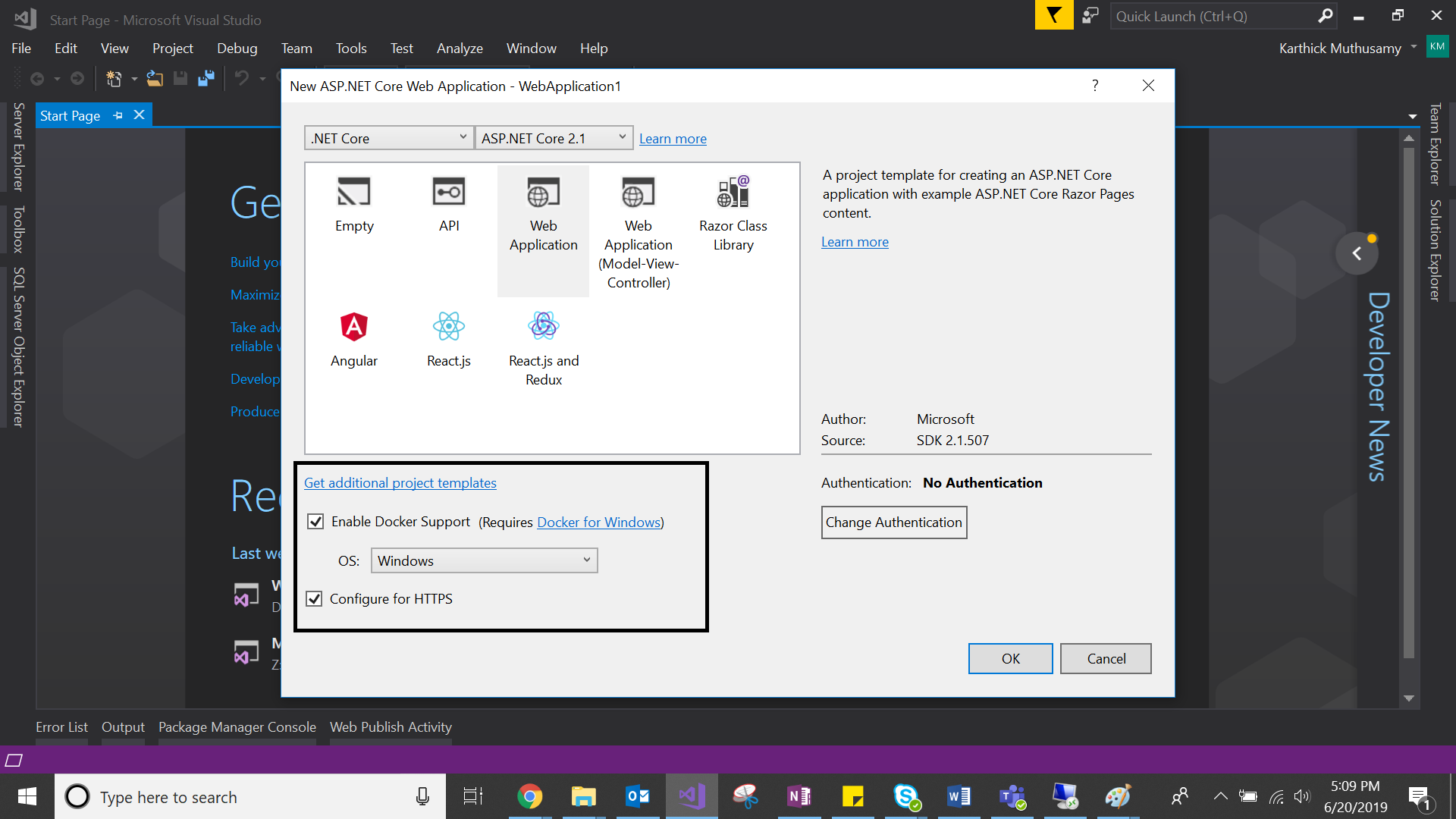
Note: While creating container registry make sure, that have admin access

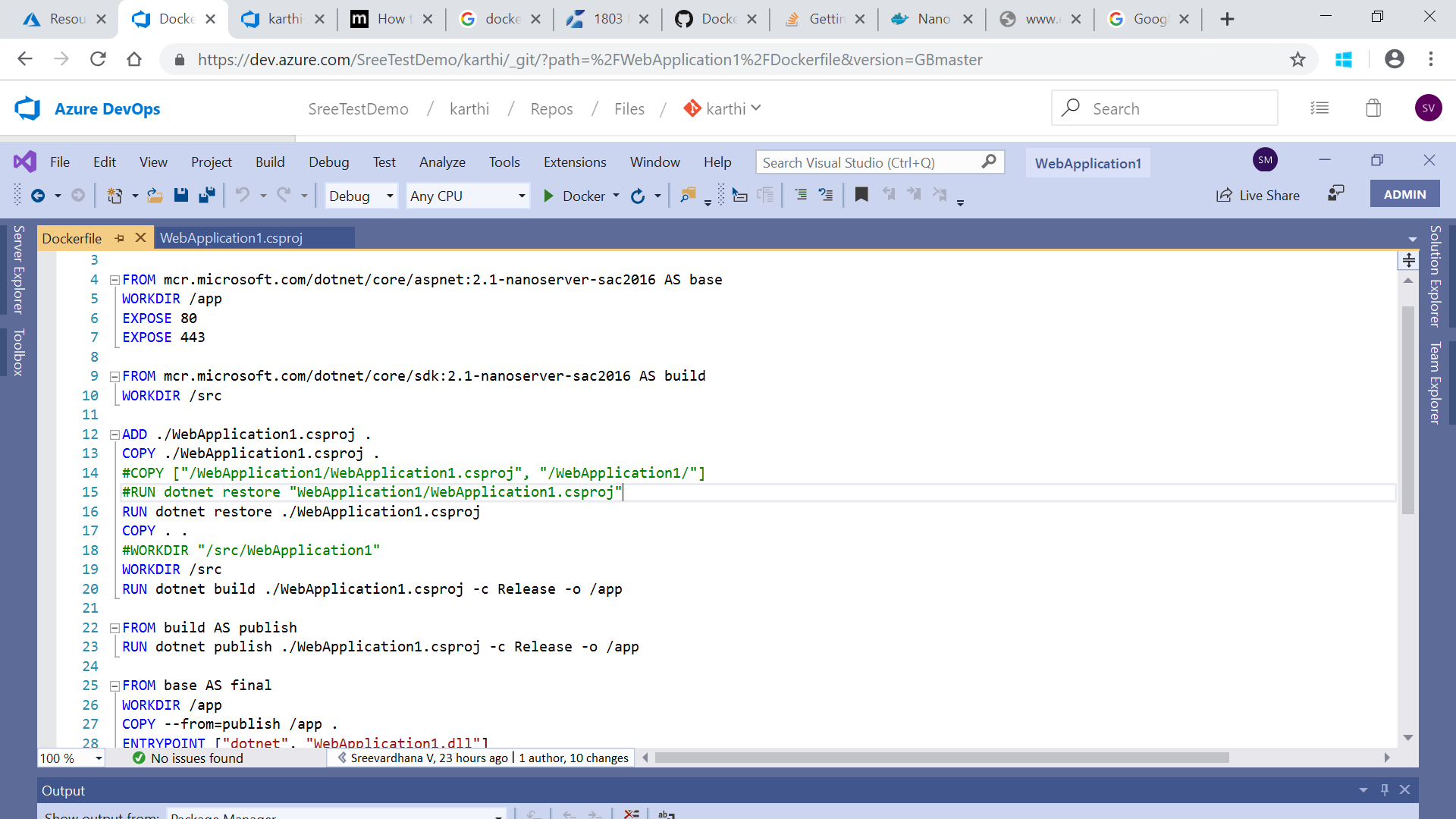
Refer below screenshot to give admin access



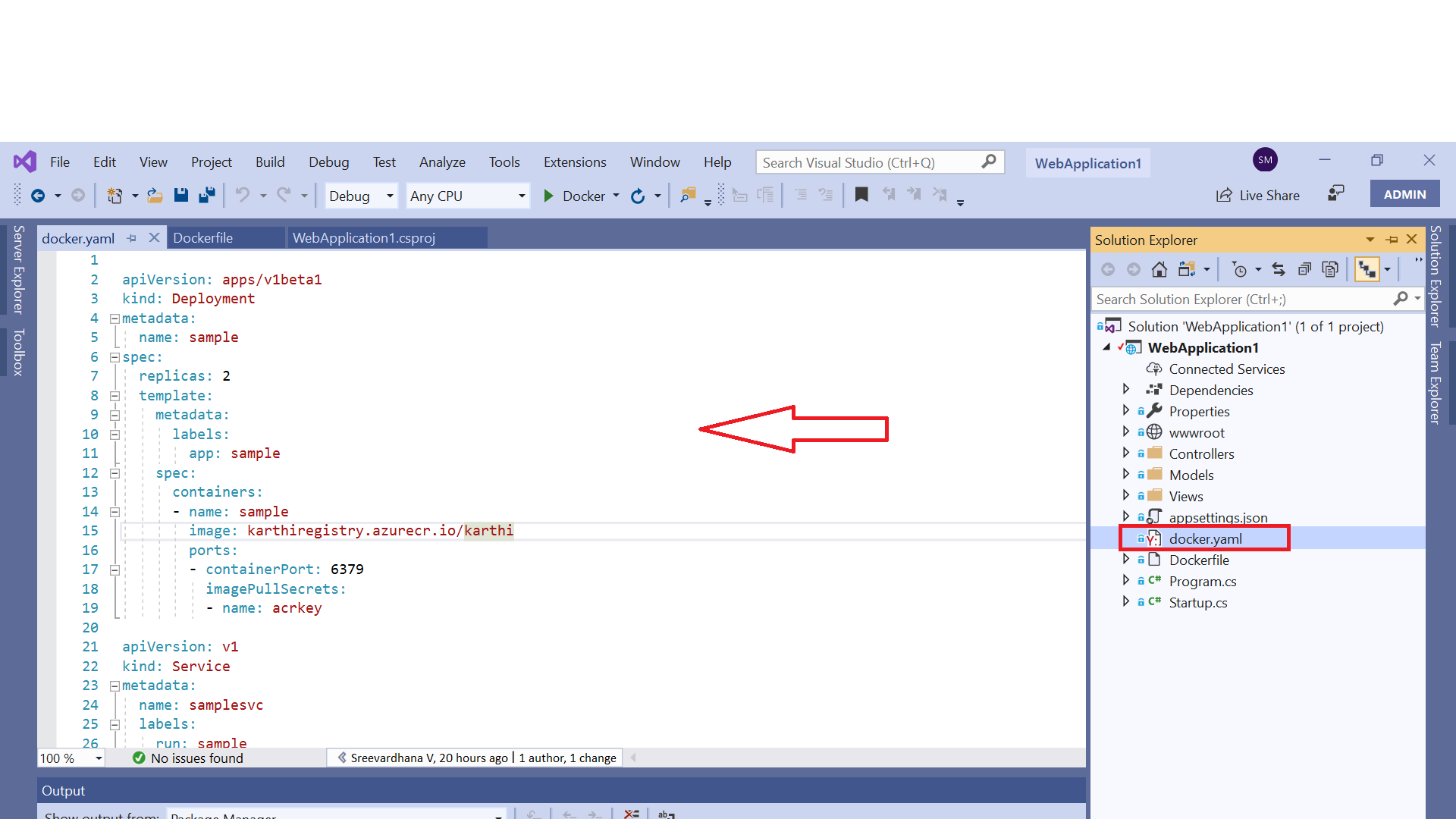
1. Create .net core project with docker support







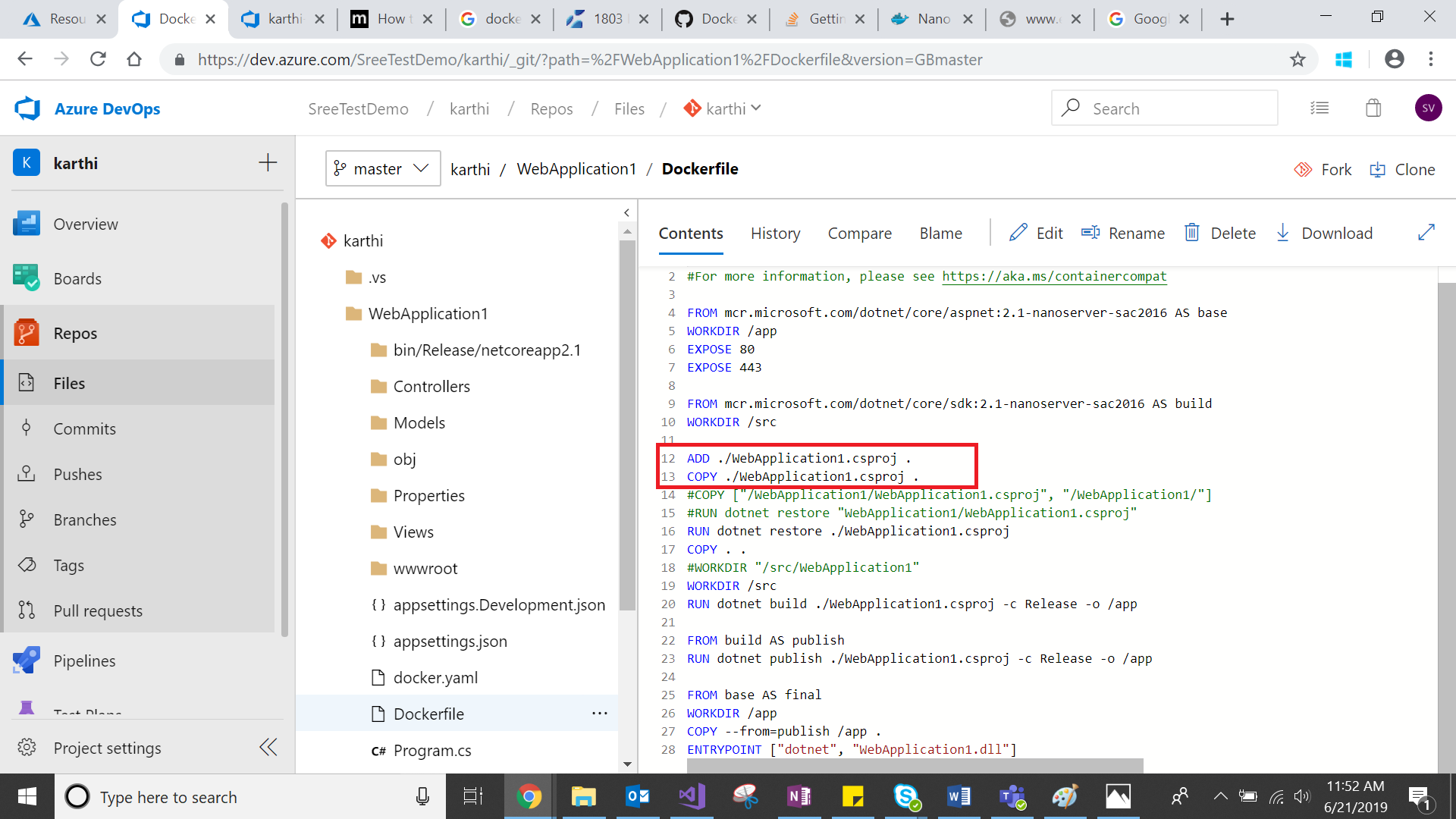
1. Add yaml file into project and configure required details



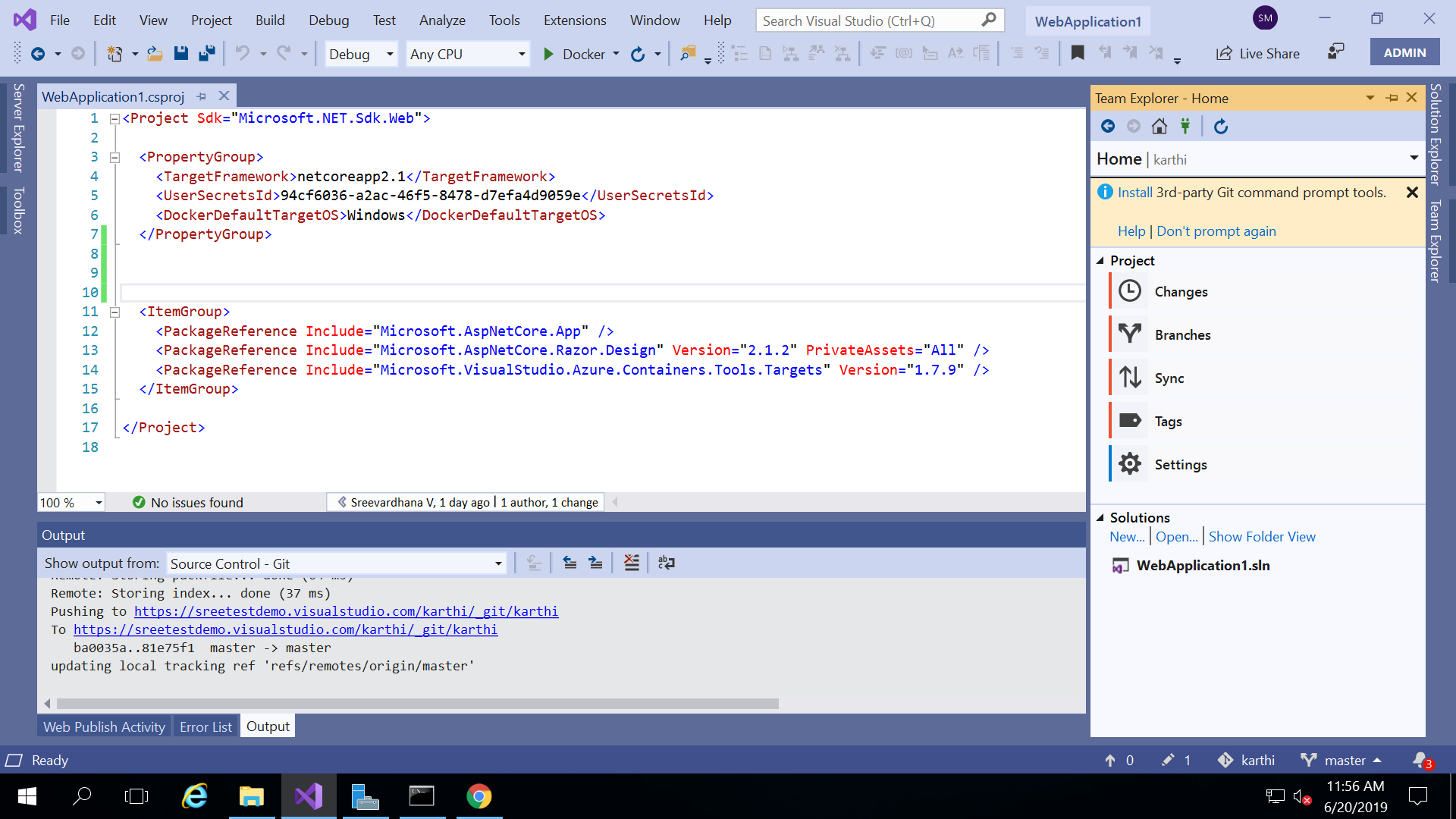
Attached docker.yaml file and dockerfile for reference

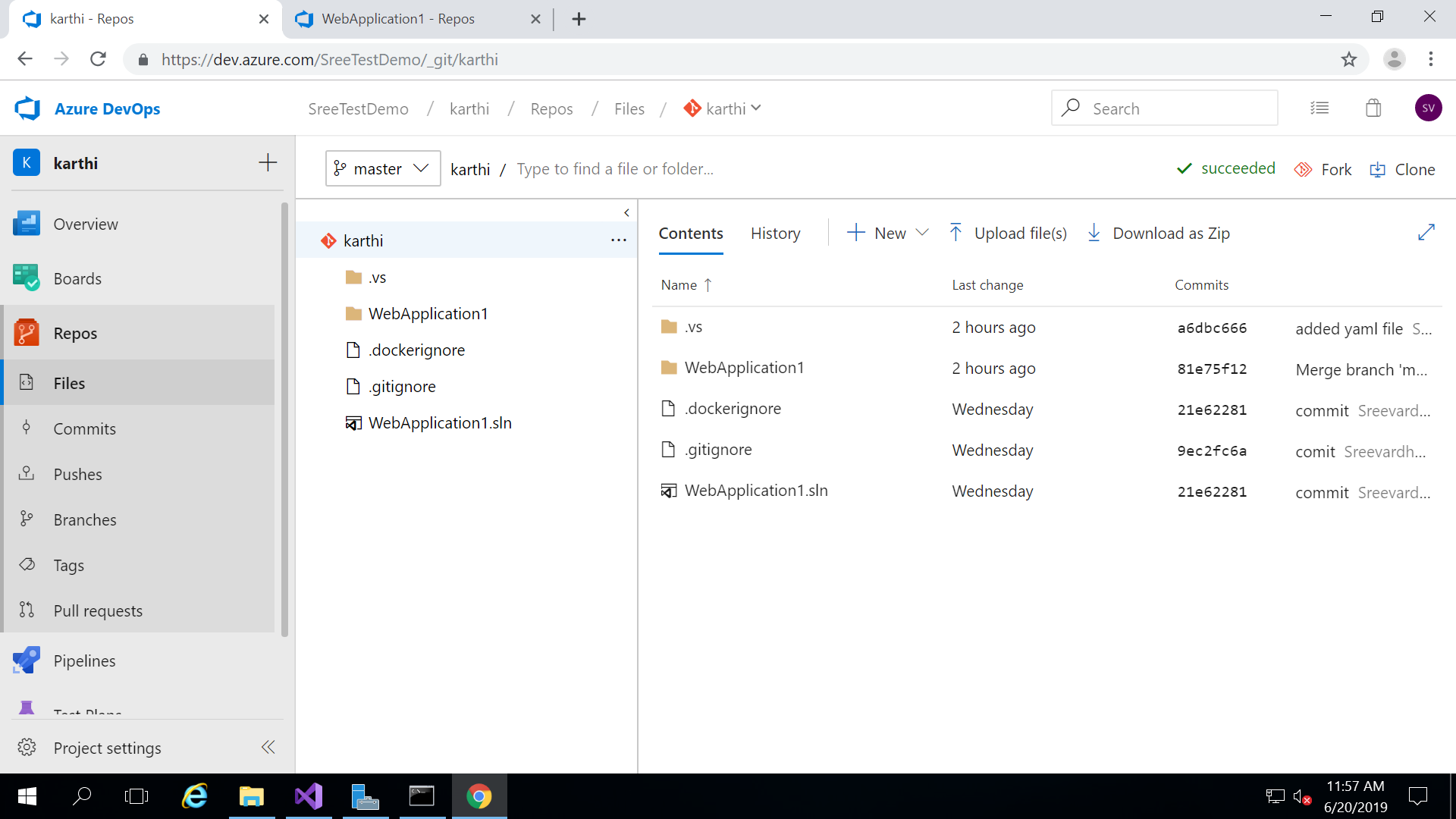


We need to change few lines to avoid errors while building image

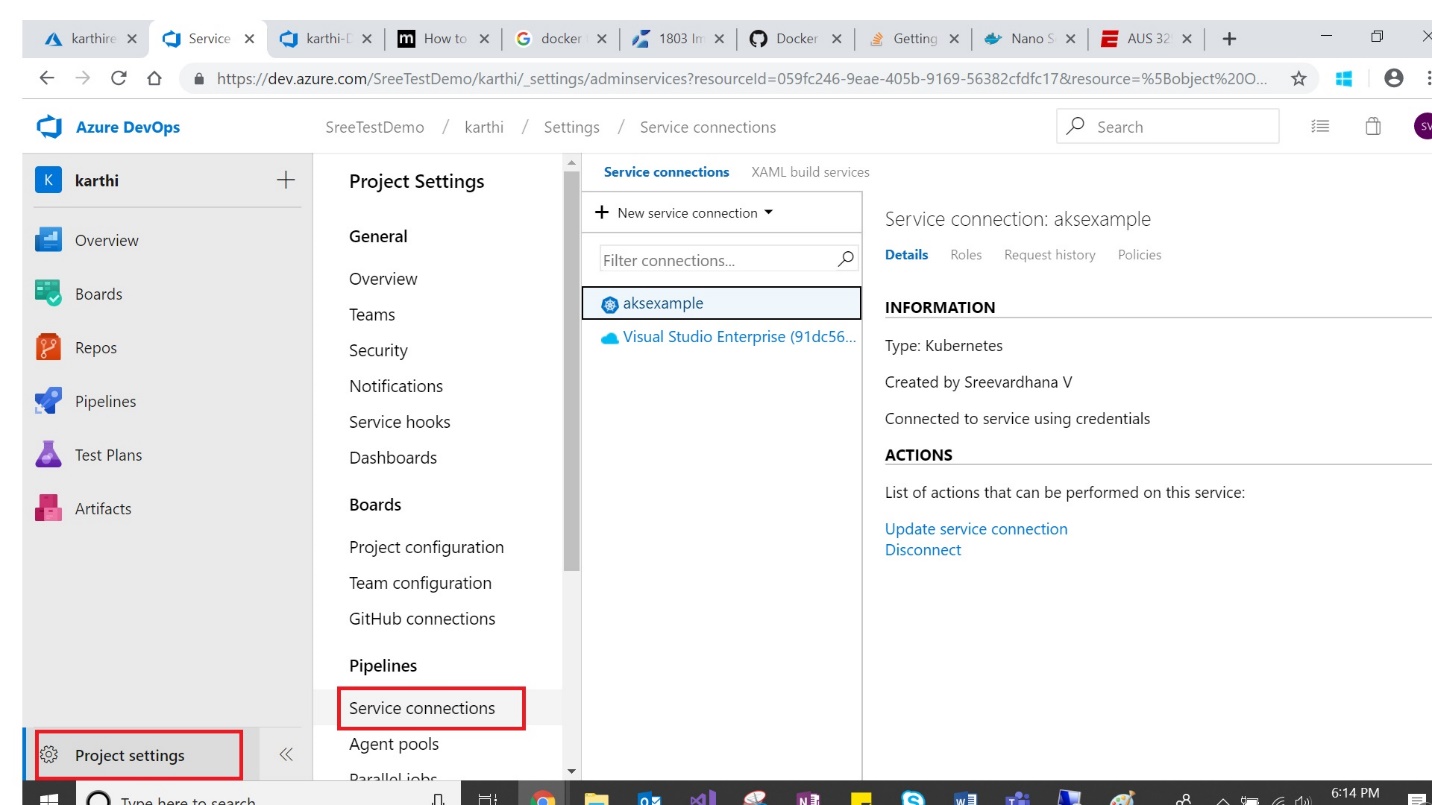


1. Login into azure devops and create repository as well
2. Push code from visual studio to azure devops

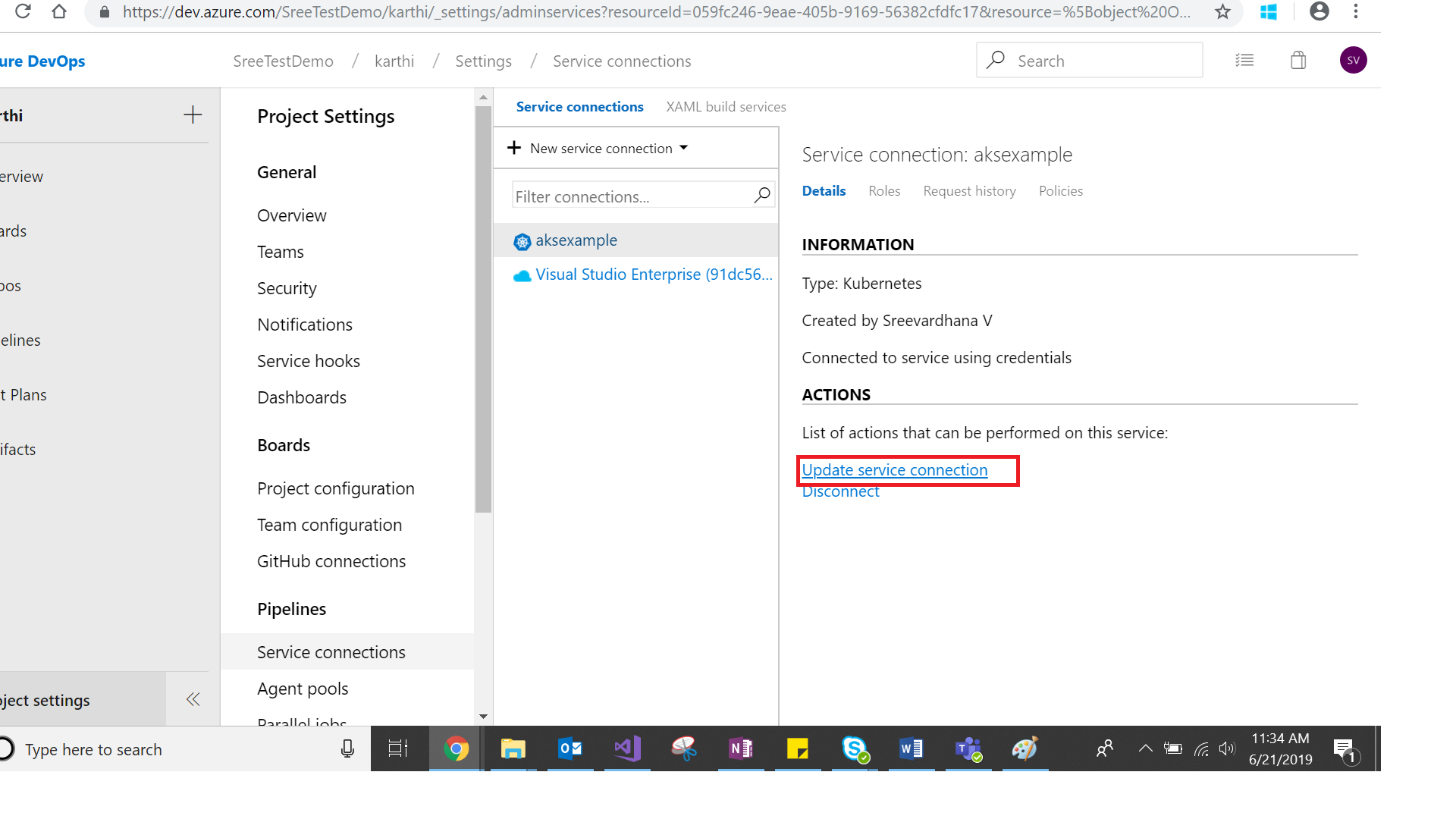




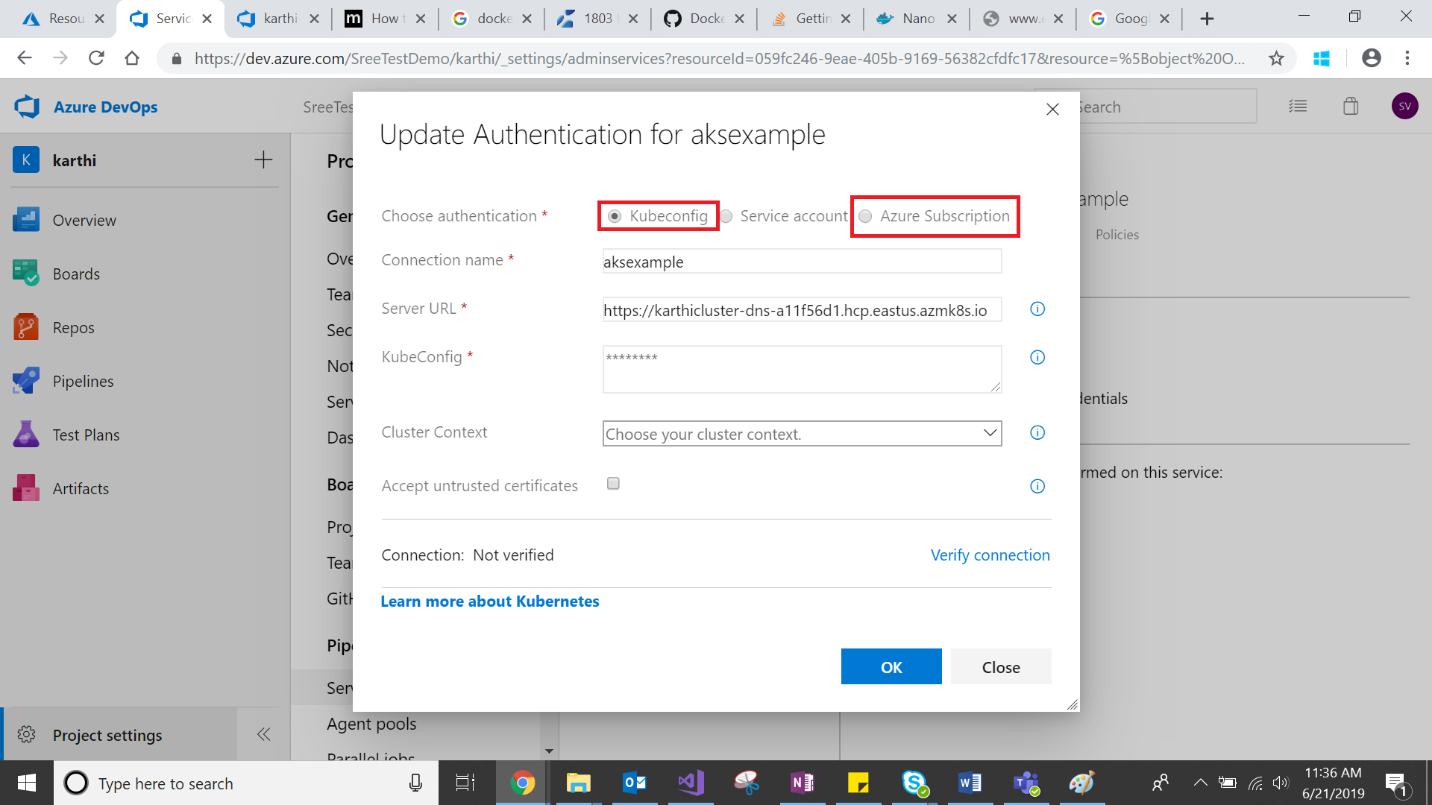
1. Configure service connections



Click update service connection



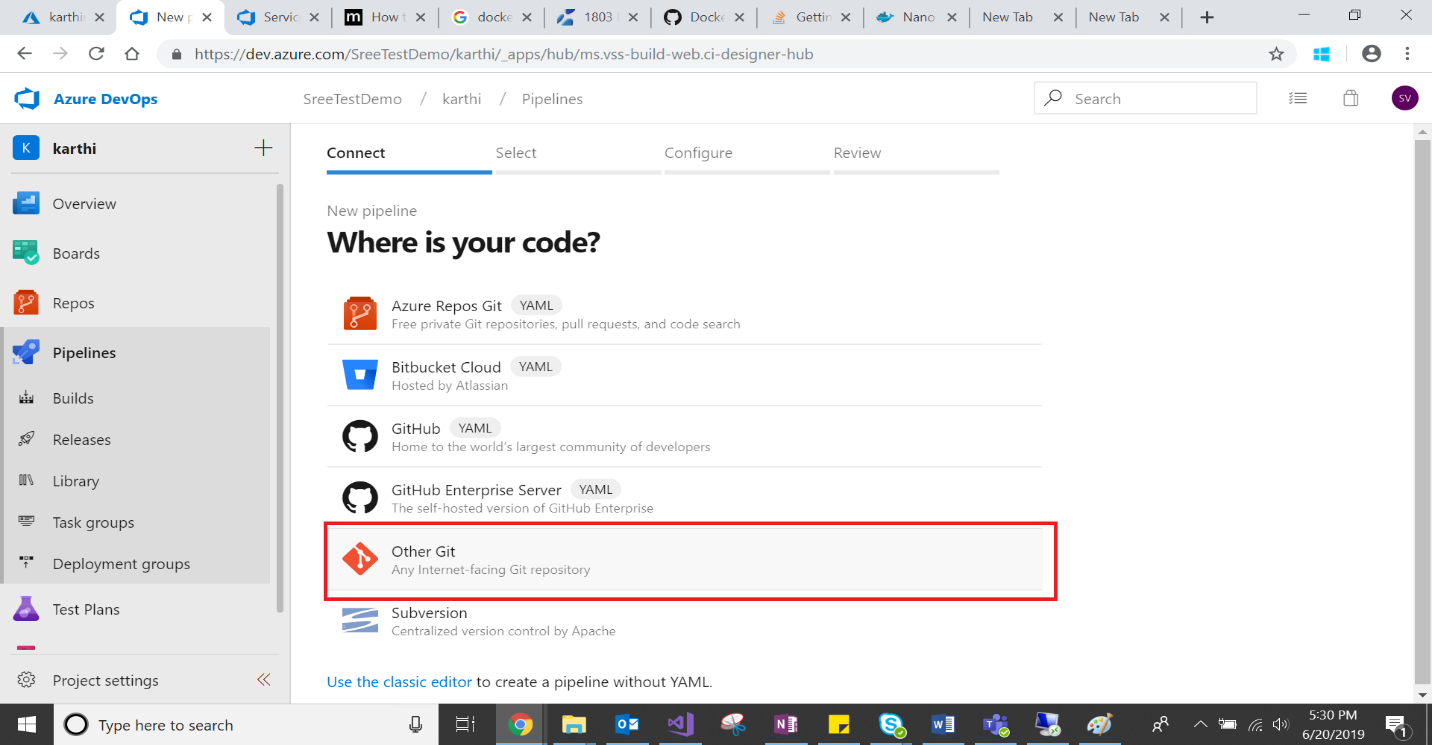
Update the Kubeconfig and azure subscription

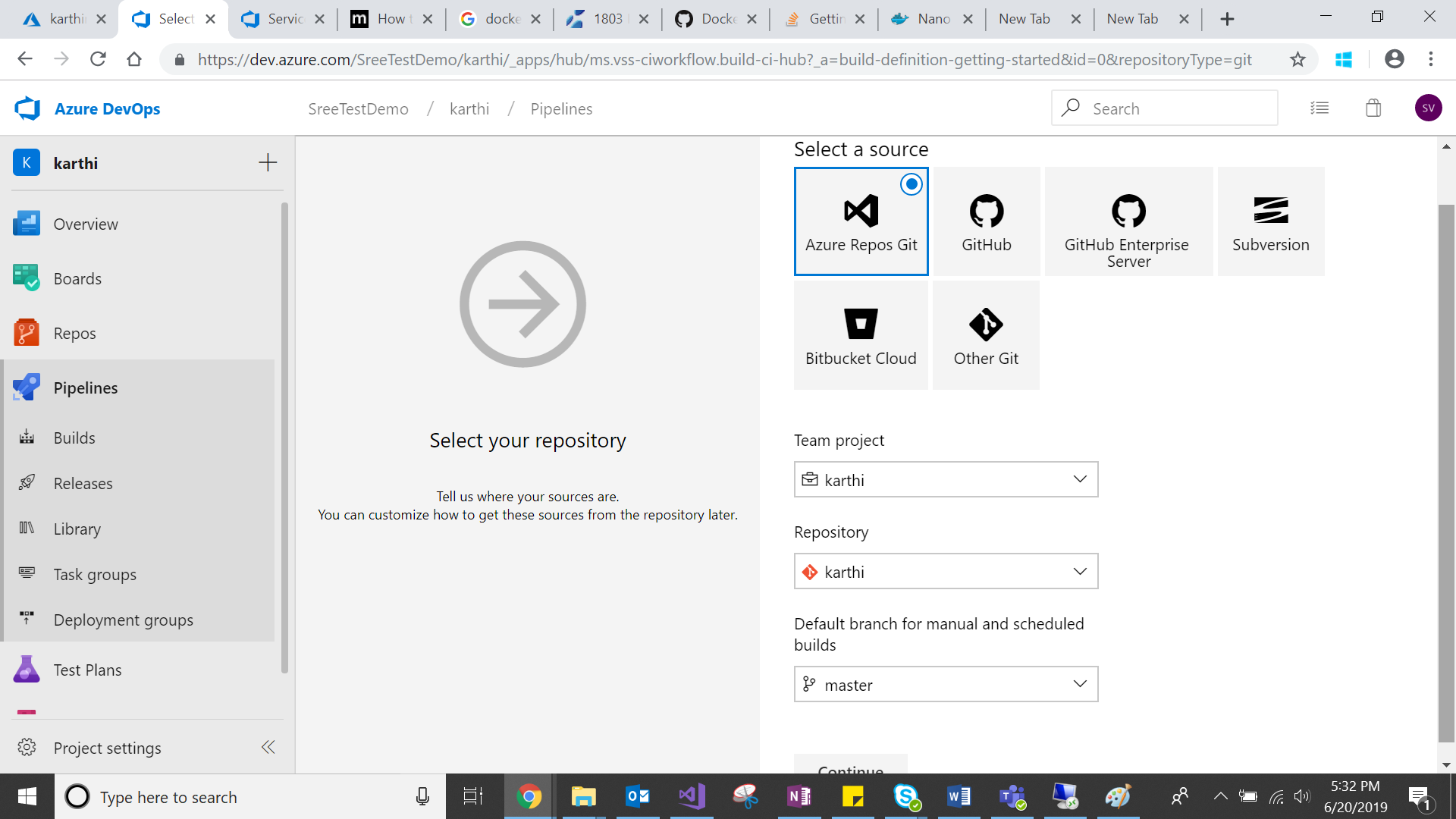


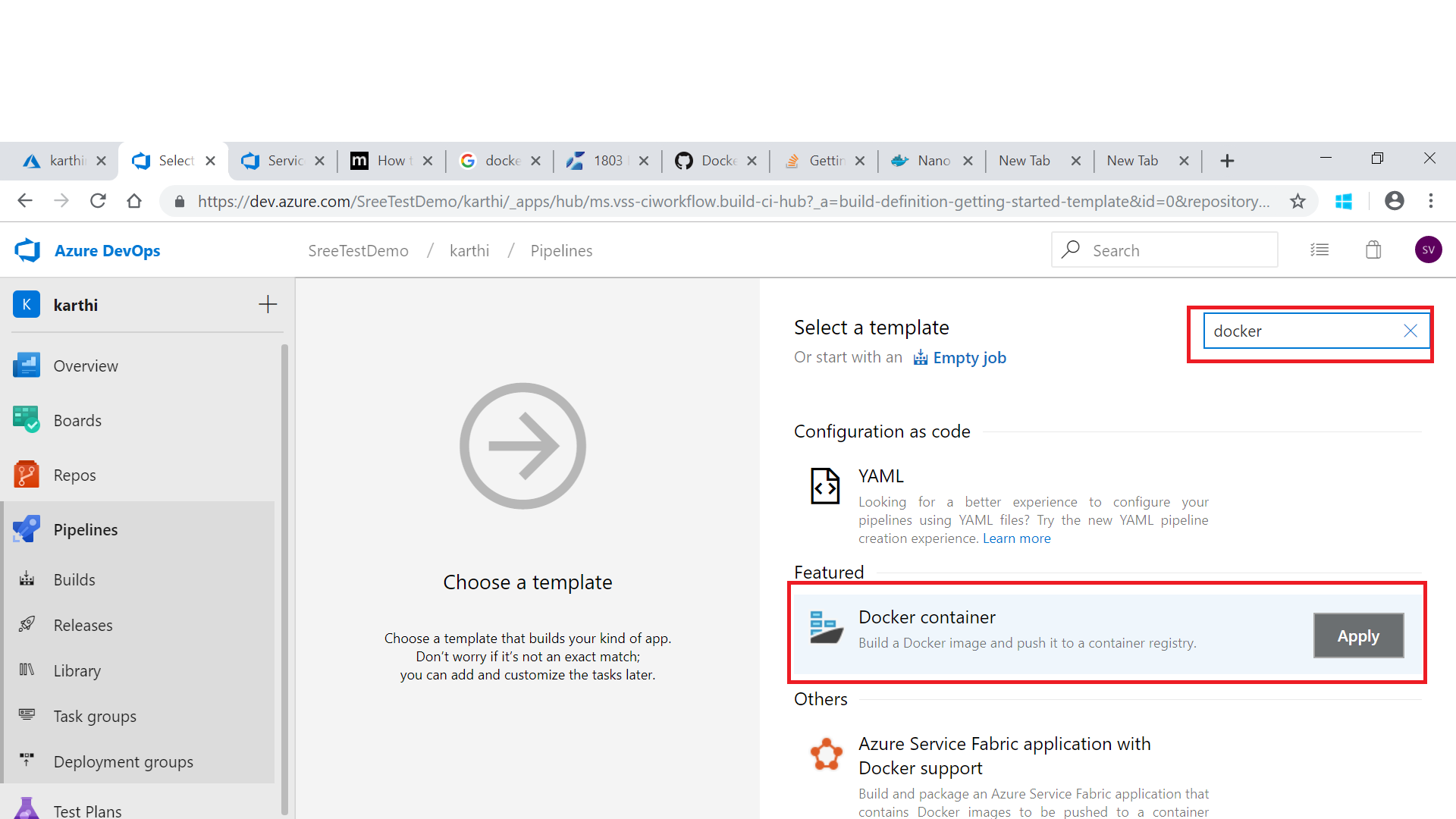
Ref below link for how to get kubeconfig and server url inorder to update kube config

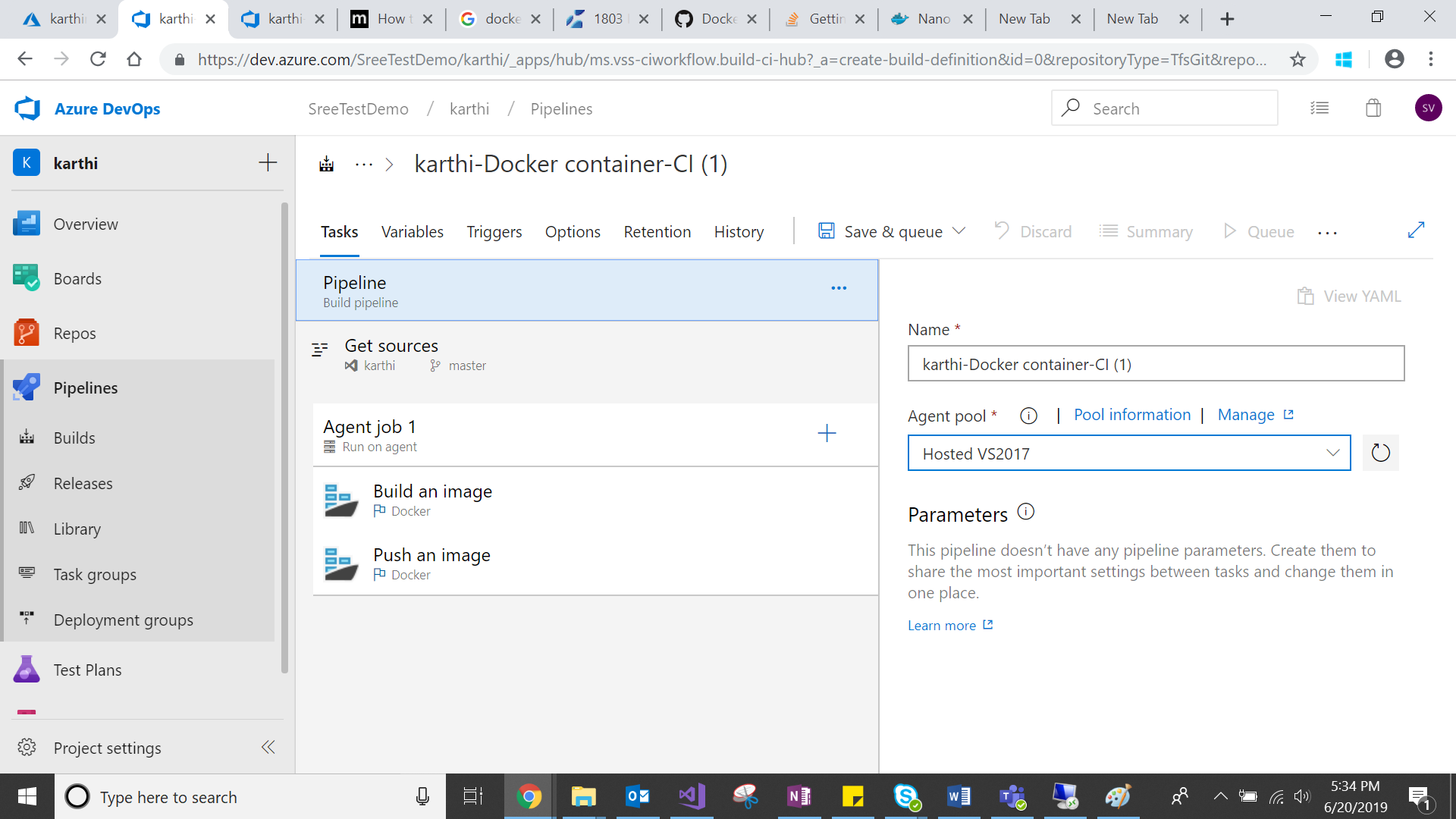
<https://blogs.msdn.microsoft.com/aseemb/2018/07/03/how-to-create-a-new-kubernetes-service-endpoint-for-aks/>

1. Create new build pipeline

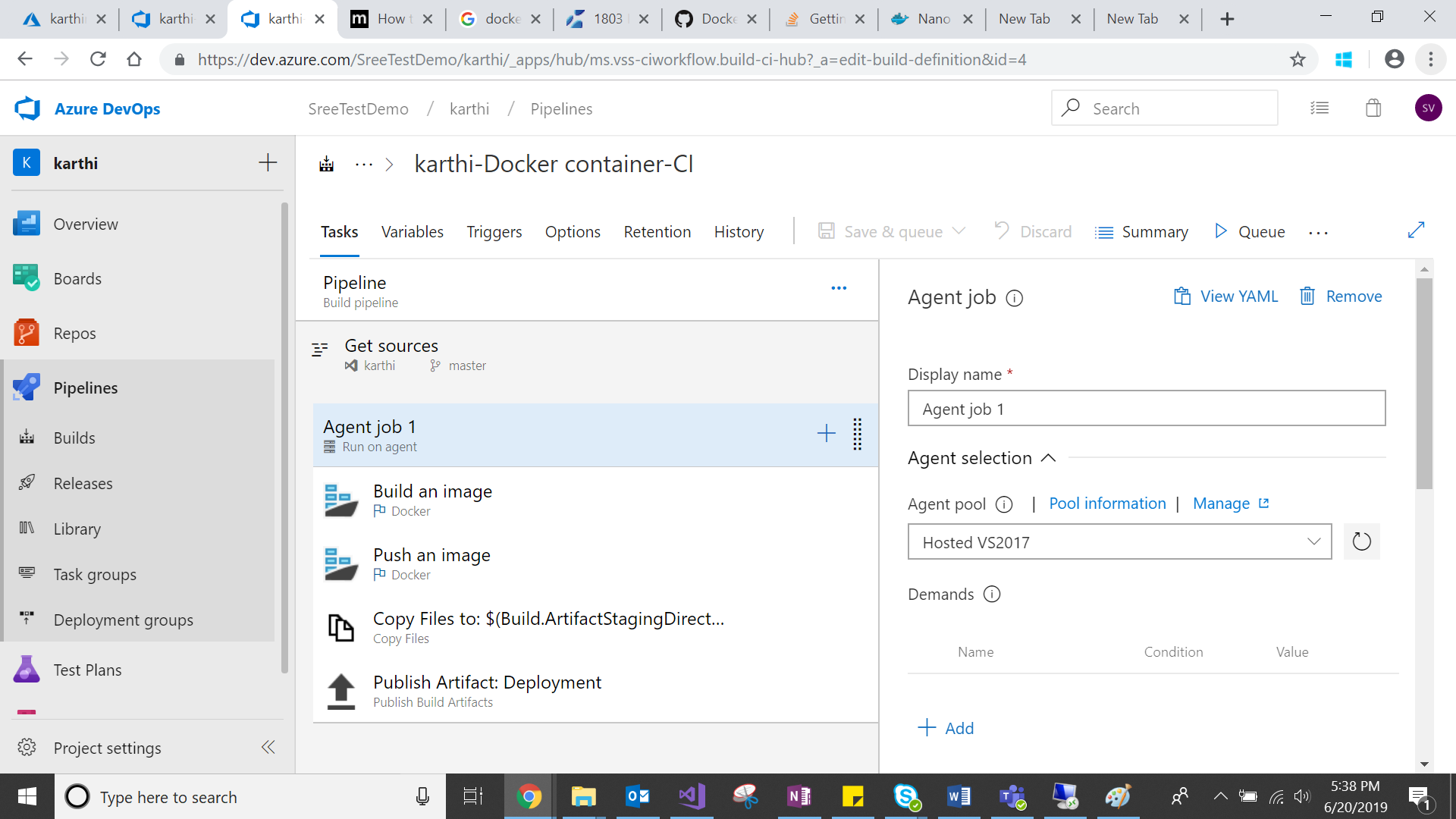


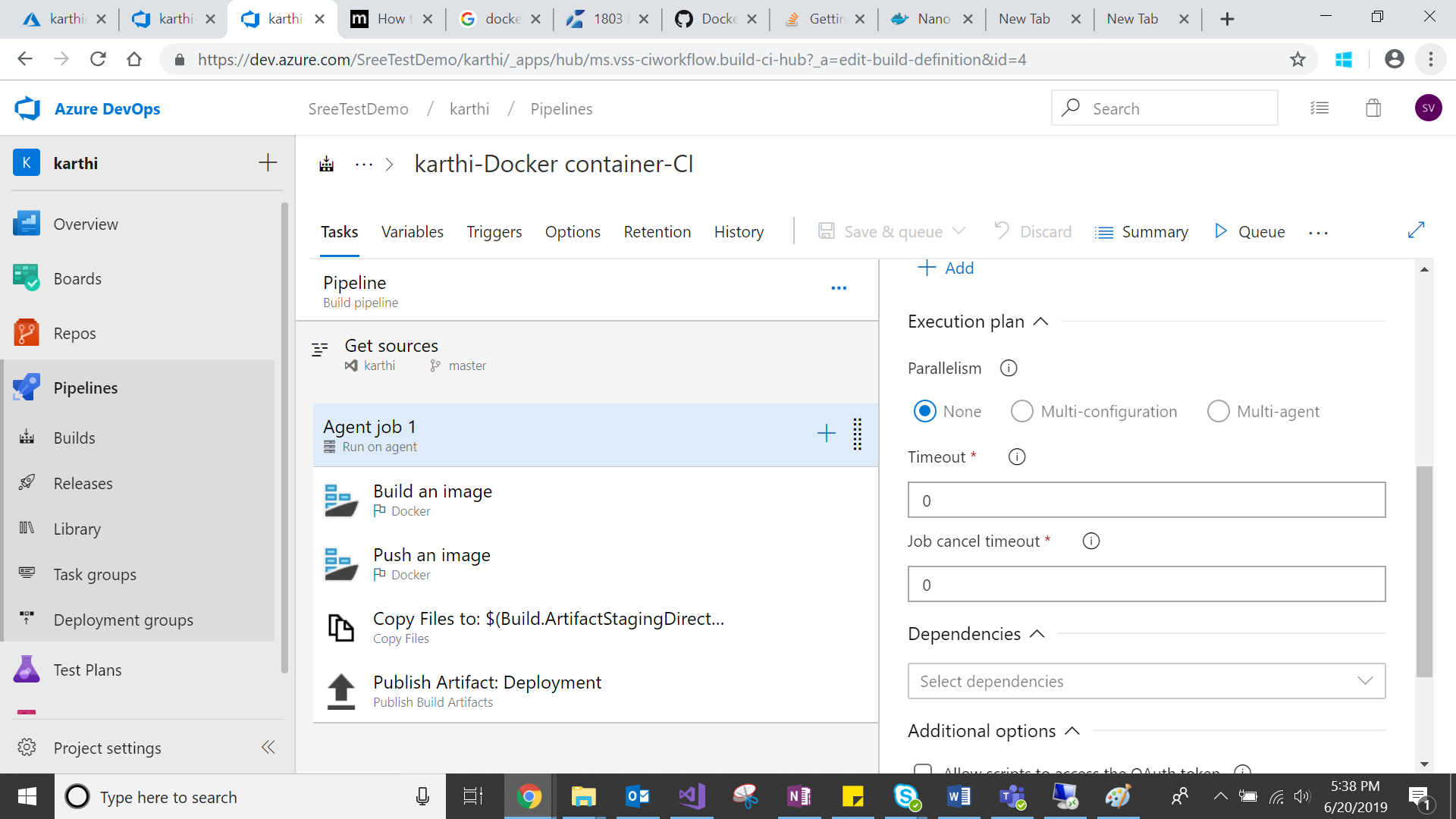


1. Search docker and select docker container
2. Click Pipeline and configure like below screenshot

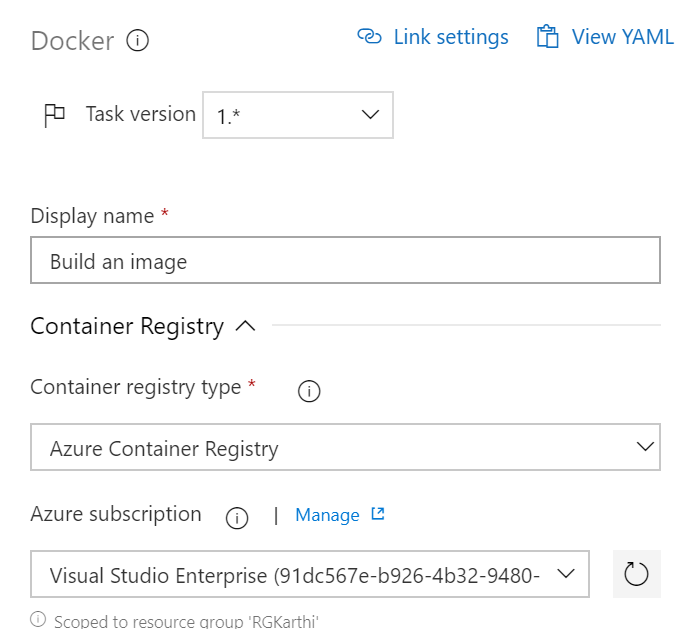


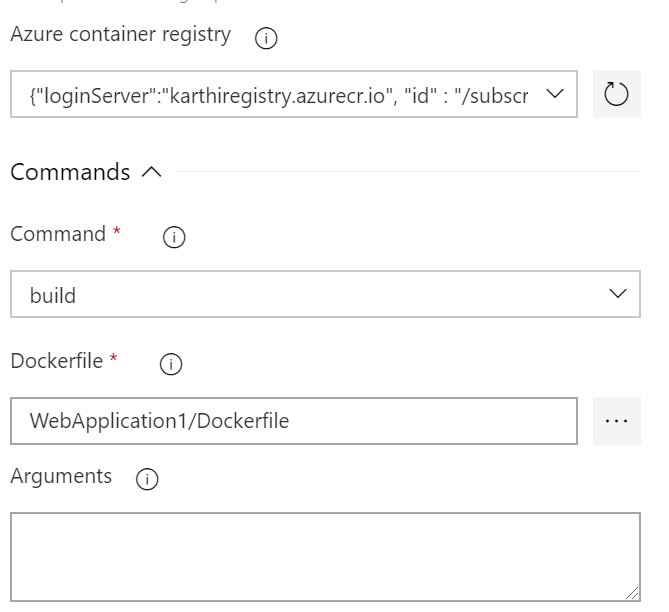
1. Click Agent Job and configure

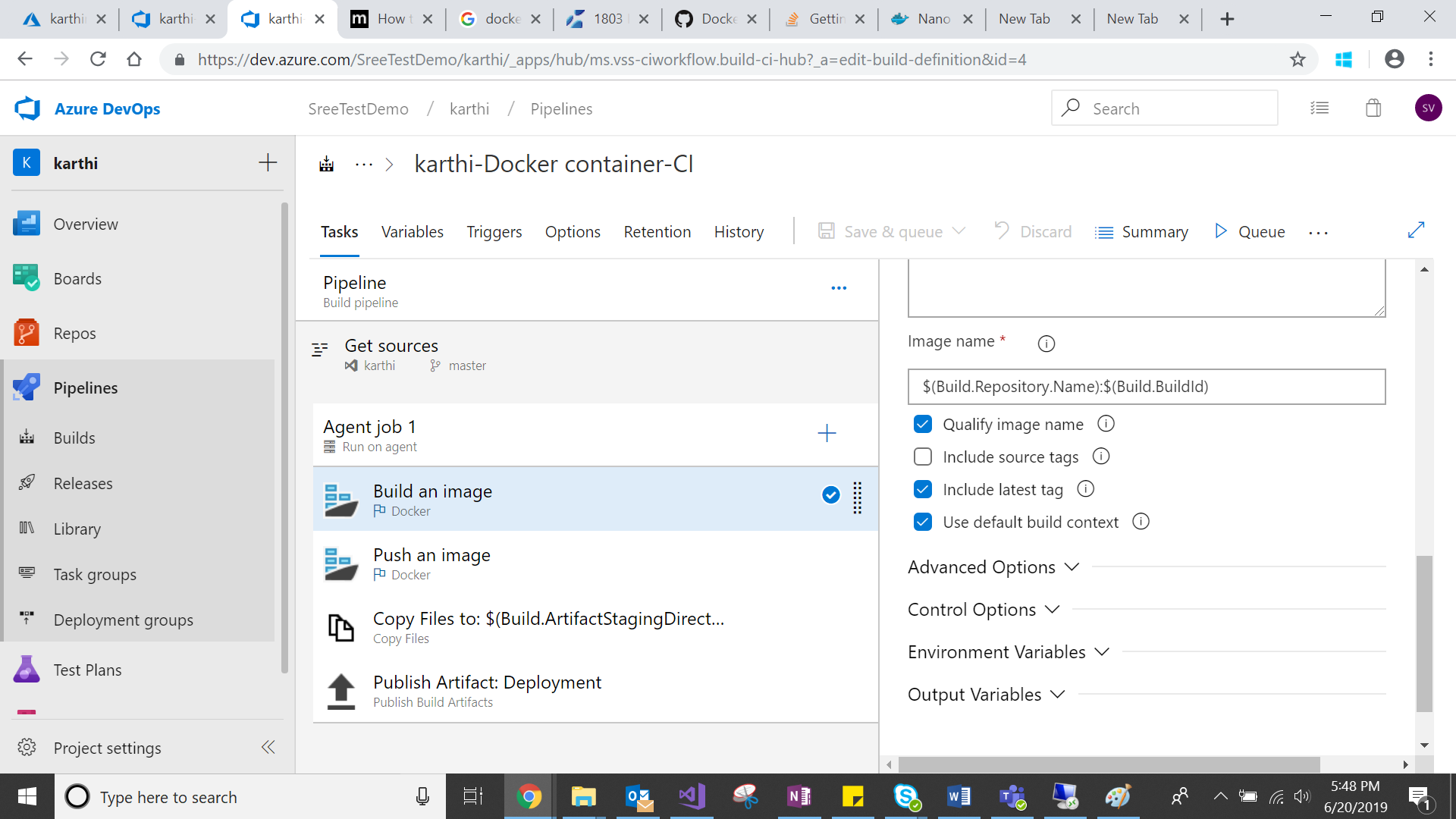




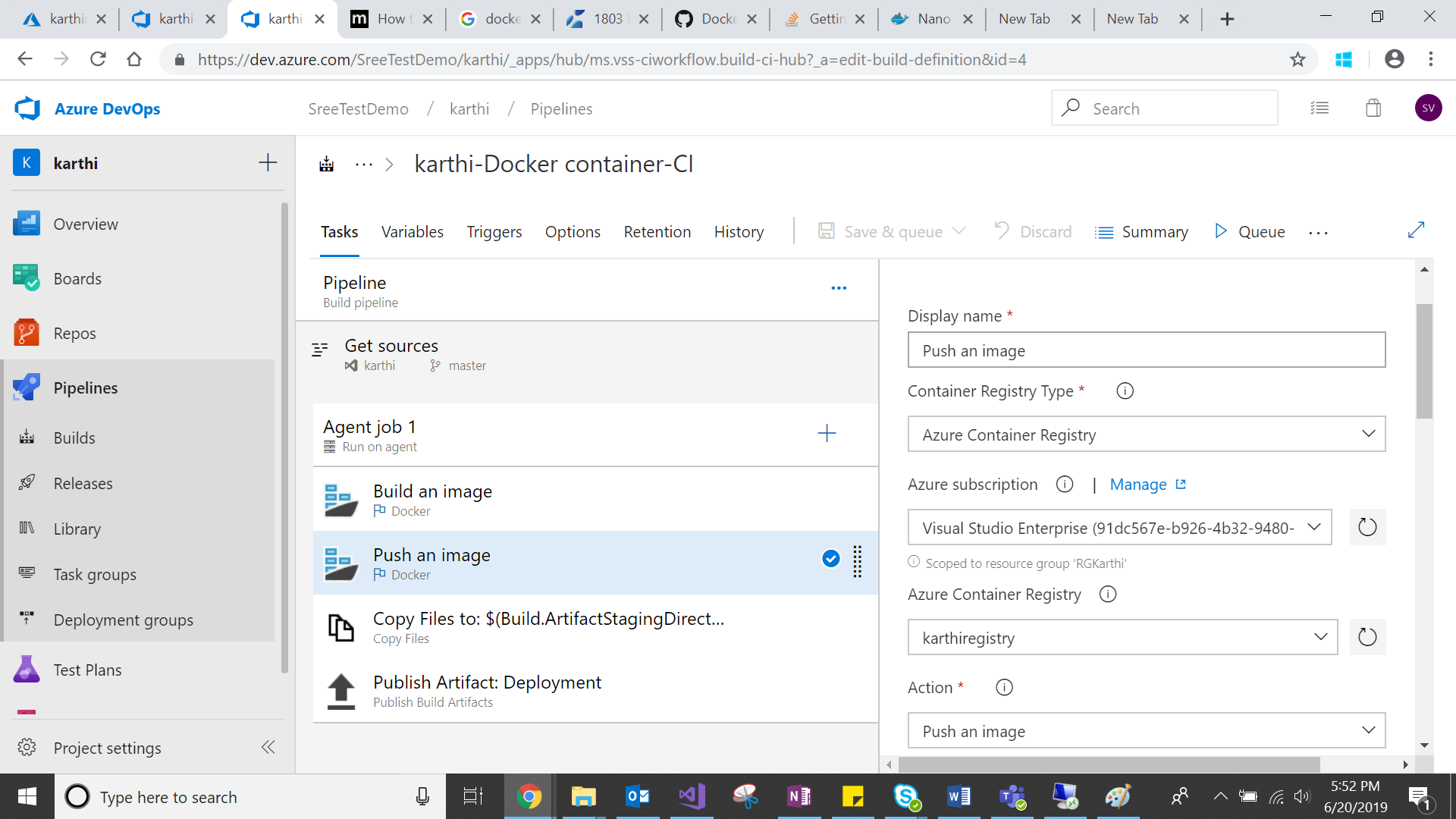
1. Click build an image and configure

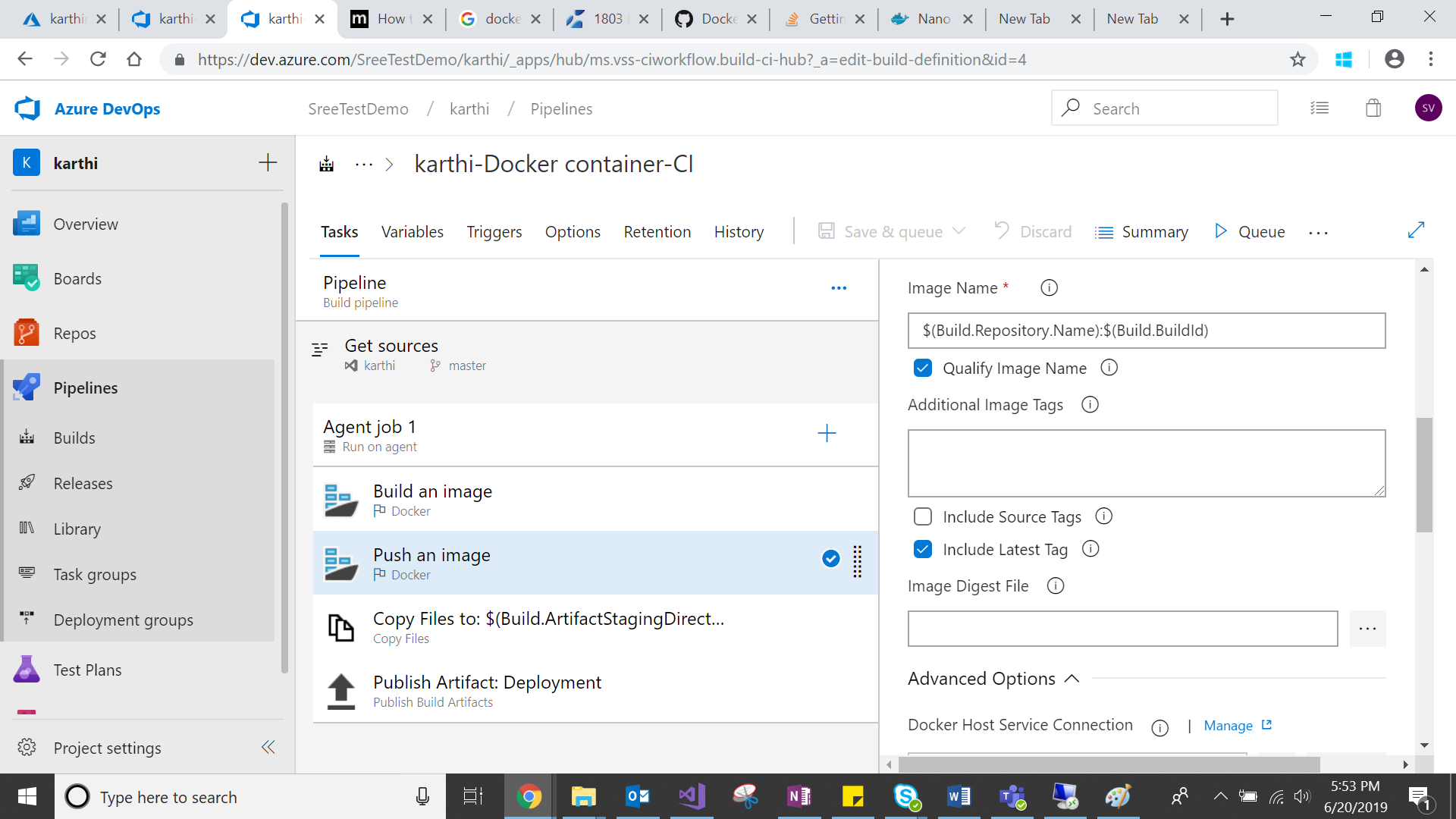


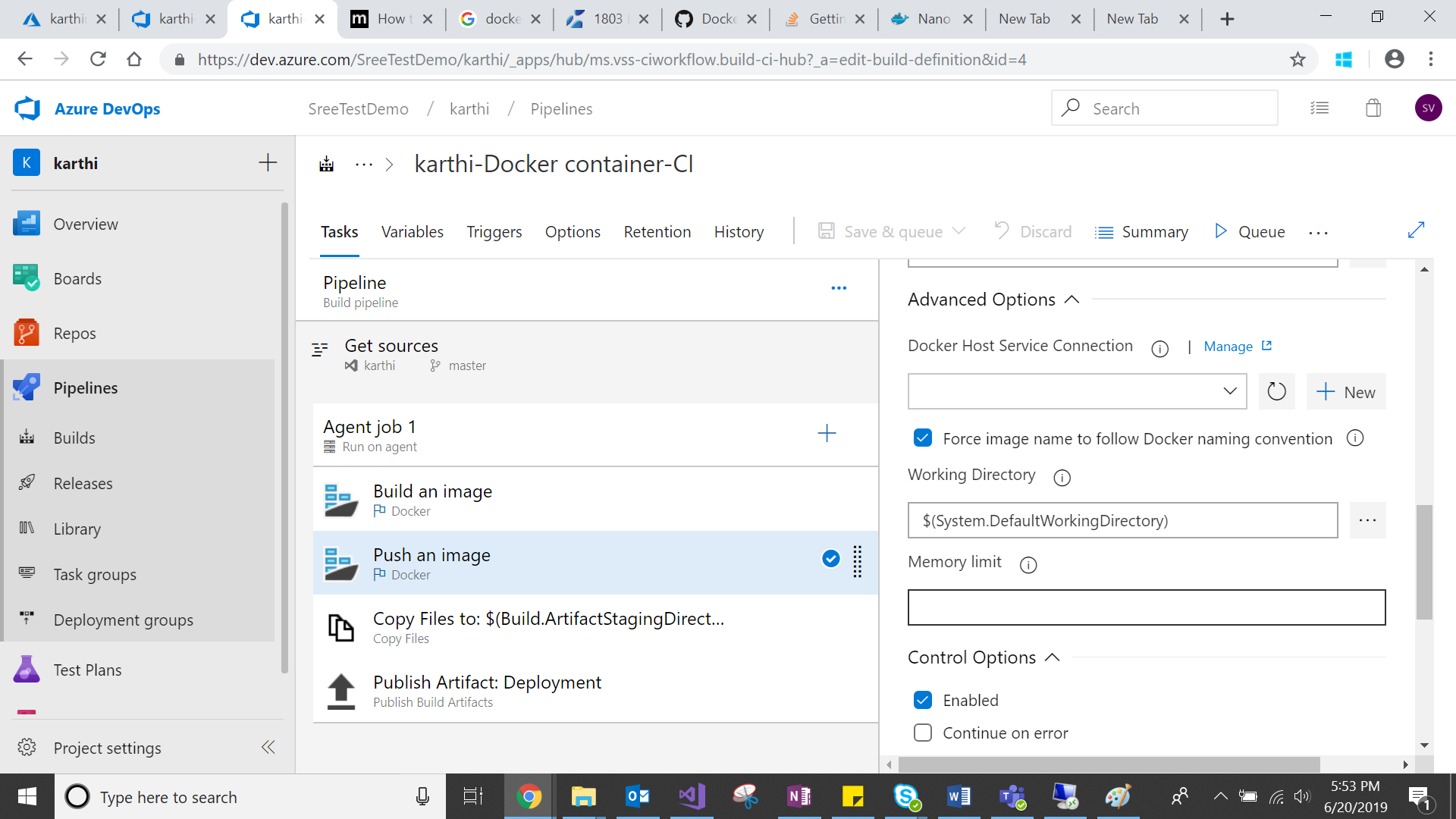




1. Click push image and configure

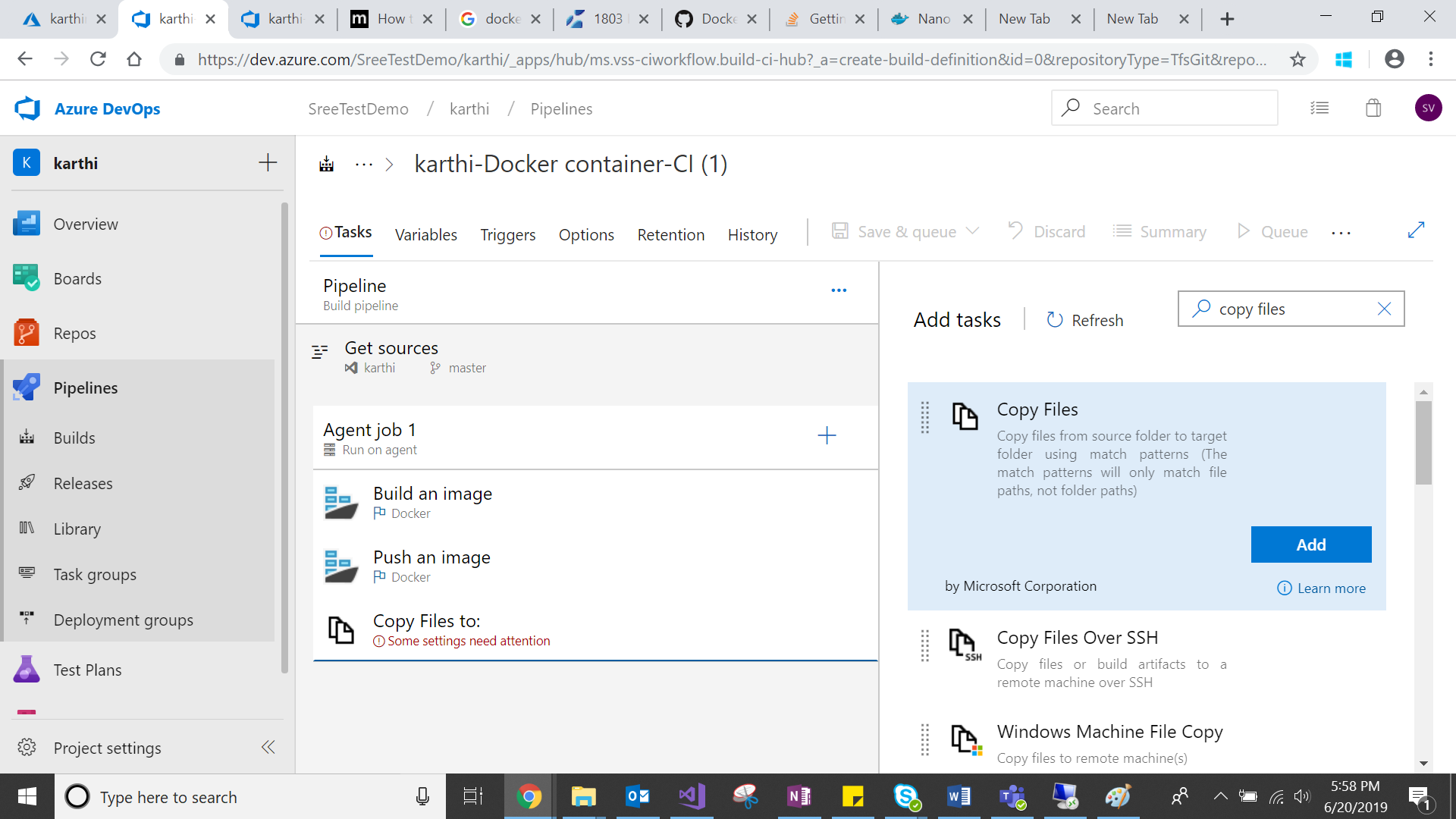


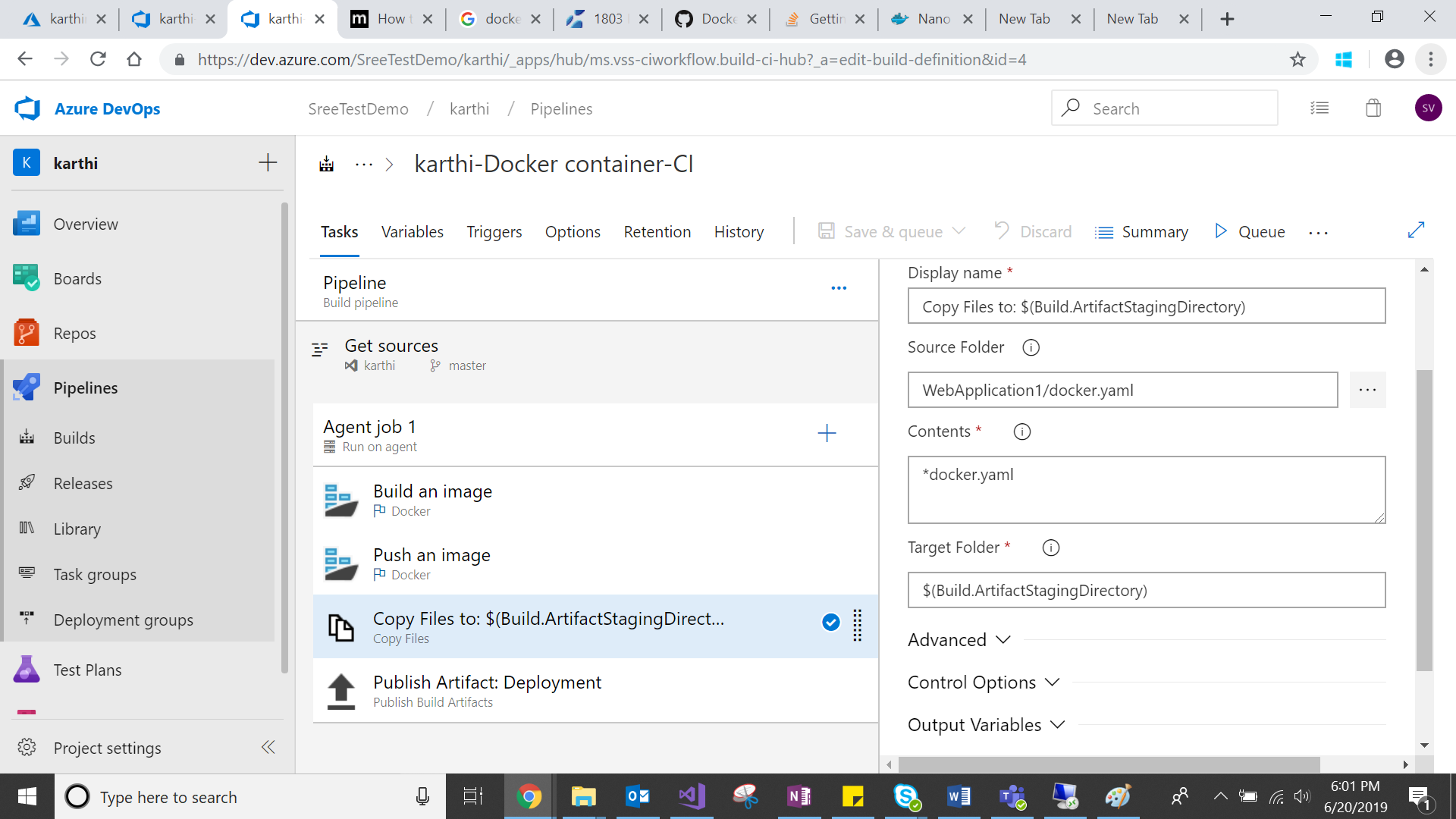




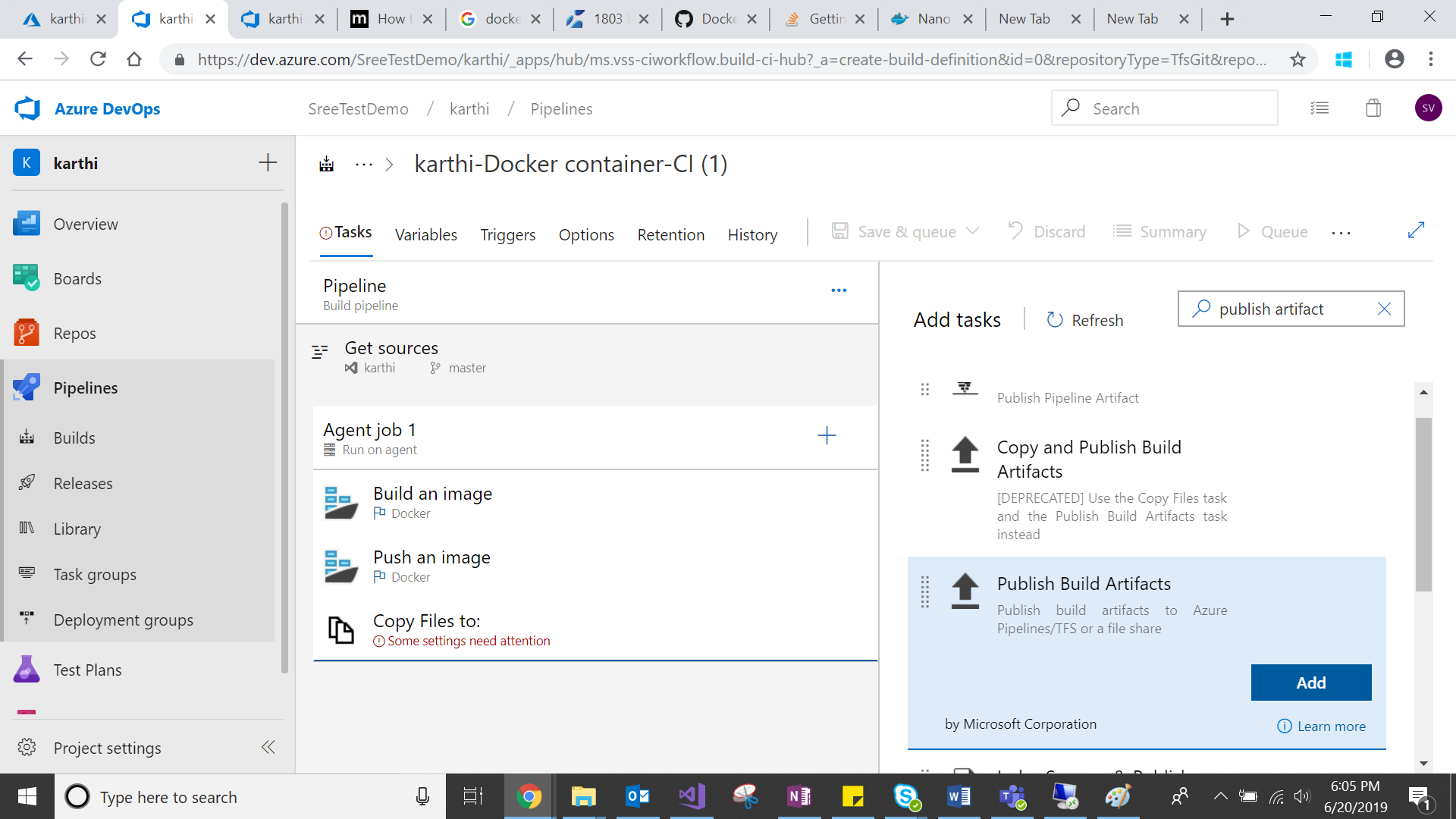


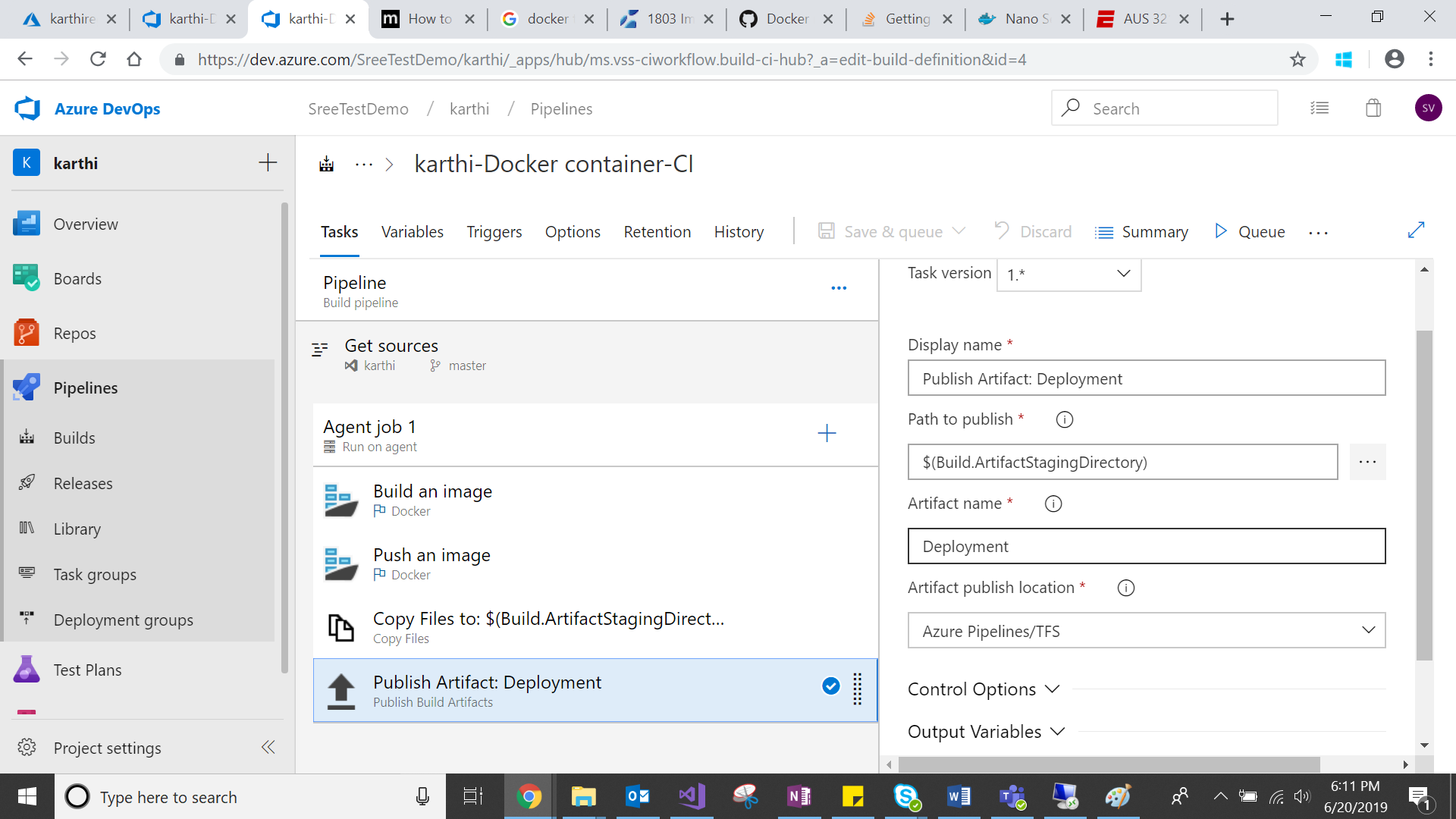
1. Click new task from Agent job & search copy file and select Copy Files then configure



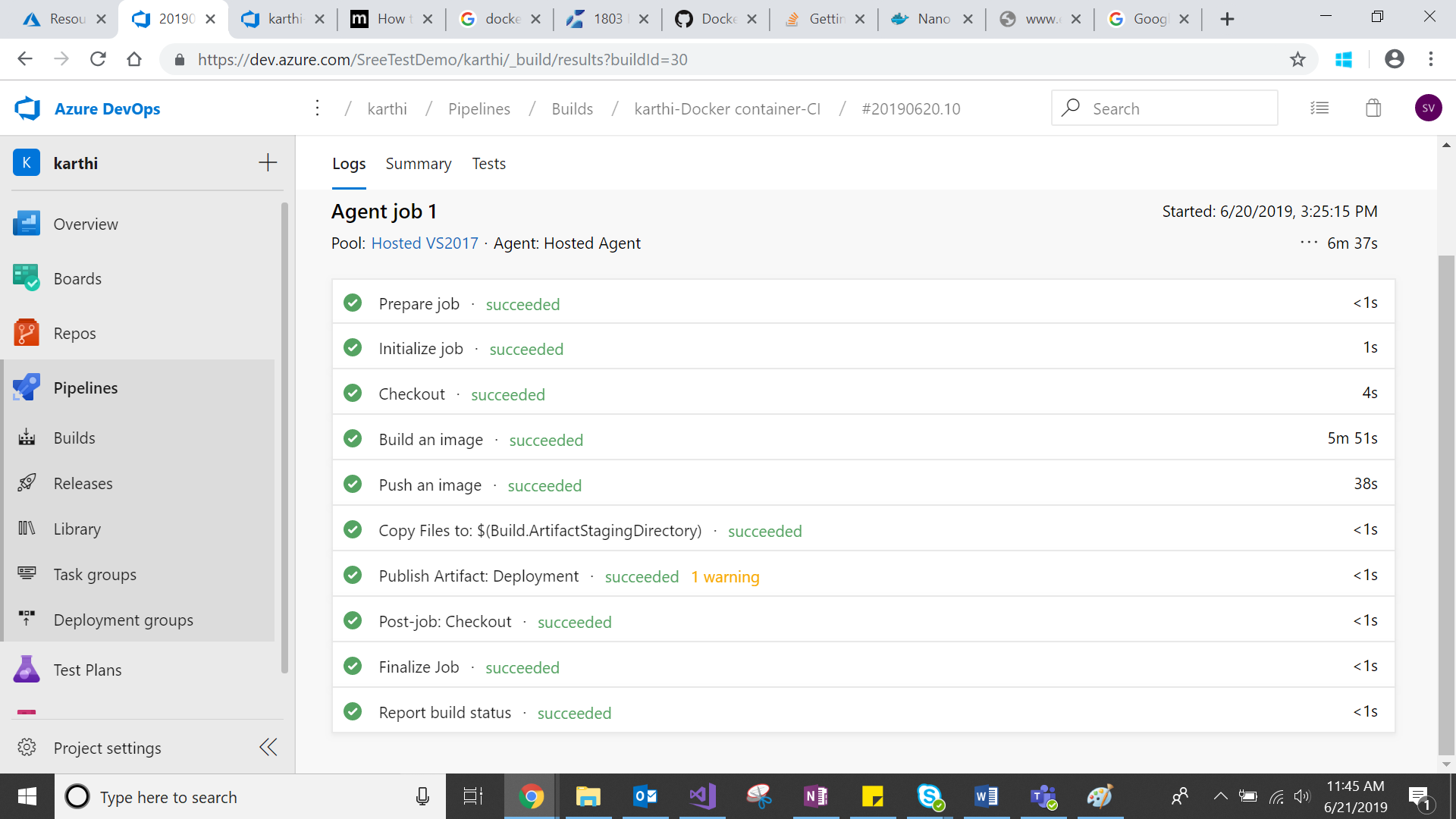


1. Add new task from Agent job & select public build artifacts





1. Click the save and queue, then you will get below screenshot



1. Go to created container registry in azure portal check the application created as image in repository

