# **Plan to host the App**

1. The App will be hosted directly onto the Azure Web App service.
2. The Static files will be kept in a separate storage account.
3. The infrastructure will be created using Terraform.
4. The Azure DevOps will be used as Deployment tool for creating Release pipeline.
5. For SSL, self-signed certificate will be created and will be bound to app during release.
6. For Autoscaling, the inbuilt functionality of autoscaling the web app instances will be used.

# **Steps Taken to Achieve the Plan:**

Note:- There are screenshots and images in every folder that showcase the implementation of the task.

## **Infrastructure Code**

The terraform files you will find under Terraform Repo folder.

Contents of this folder are as below:

app.tf – This file consists of the code that will create WebApp, App Service Plan

backend.tf – This file has the code which tells the terraform where to keep the tfstate file safely.

main.tf – This has the code to connect to azure and create resource group

output.tf – This file has the details of the variable that we need from terraform as output.

storage.tf - This file consists of code regarding storage account creation.

variables.tf – All the variables used across the terraform project are declared here.

After writing these files, I had uploaded it to Azure DevOps repo.

## **Release Pipeline**

You will find the template of the Release pipeline created by the name SITReleasePipeline under Release Pipeline folder.

The steps involved in creating this Release Pipeline are:

## **Pipeline structure**

The pipeline stage structure you will find under the Release pipeline folder.

## **SSL implementation**

I had created the SSL certificates using the PowerShell commands in my local laptop. Th commands and screenshots are available under **Cert and SSL implementation** folder.

## **Autoscaling**

I have used the azure PowerShell module commands to set up the autoscaling profile and rules.

You can find the same under Autoscaling folder

## **Testcase**

I have written just one test case that checks whether an app is up and running after deployment or not. You will find it as the last steps of the release pipeline.