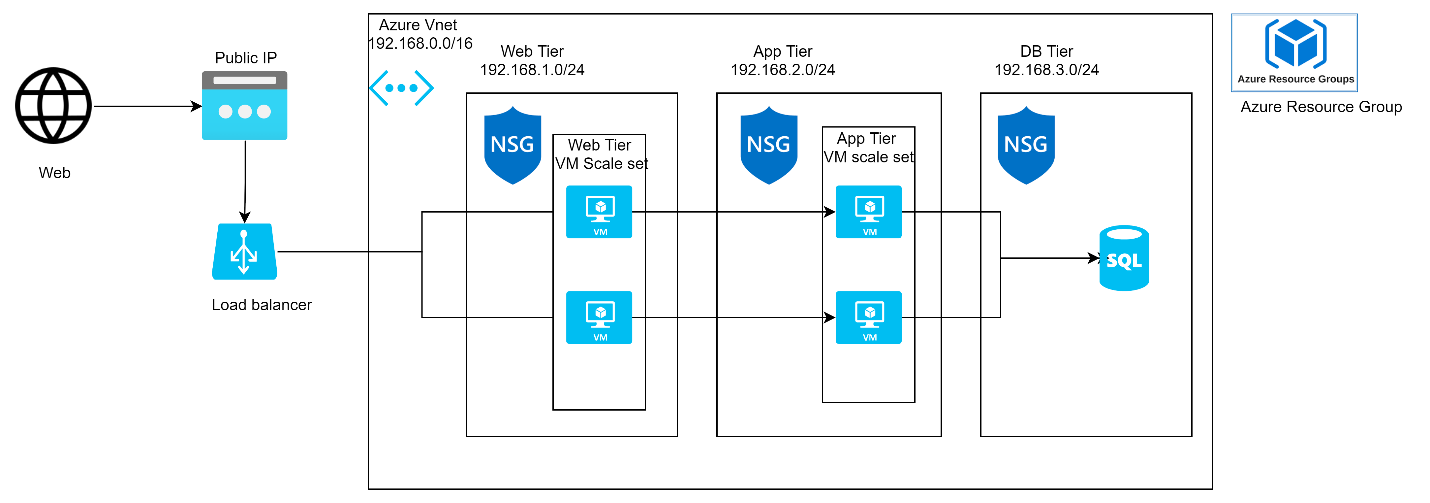
**Challenge #1 Question:**

A 3-tier environment is a common setup. Use a tool of your choosing/familiarity create these resources. Please remember we will not be judged on the outcome but more focusing on the approach, style and reproducibility.

**Solution:**

**Architecture design**



Architectural design includes a virtual network with three subnets for each Business Tier, Azure VM Scale set for High Availability created in Web & App Tiers.

Three Network security group created and attached in the subnet level, where rules configured are

1. Web NSG: allow inbound traffic from internet only
2. App NSG: Allow traffic from first virtual machine only and can reply the same virtual machine again.
3. DB NSG: App VM can connect to database and database can connect to App VM but database cannot connect to web VM

**Resource Modules**:

Terraform Modules are used for code reusability and five modules are created and with a combination of services. Below are the module details

1. Resourcegroup – Creates resource group
2. Networking – Creates Vnet & Subnets
3. SecurityGroup – Creates NSG groups & required rules and associate with the Subnets
4. Compute – Create Public IP, Load Balancer, LB Backendpool, LB probes, VMSS (Virtual Machine Scale set) for App & Web
5. Database – Creates database server & database

All the resources are placed in the module folders and variables are stored in terraform.tfvars

I have tested by Applying the terraform code and all the resources are deployed as per architecture design and I have attached the statefile with the resources detail.