**204\_Phonebook.tf :** Phonebook application’ı beş docker makinasına (1 x leader manager, 2 x manager, 2 x worker) deploy eder.

Leader manager, manager ve worker sac ayakları üzerinde üç template’ten oluşan bir terraform file’dır. İnfrustructure’a Docker, python, git yükler.

* Leader manager tarafından Github’dan Docker compose ve docker file’ın çekilmesi ve docker file’ın build edilip ECR repoya gönderilmesi
* Diğer makinalar tarafından imajın ECR repodan çeiklmesi,
* Database’in dockerhub’dan çekilmesi,
* App server’ın ECR repo’dan çekilmesi,
* Gerekli role’lerin tanımlanması

**205\_install-jenkins.tf :** t3medium makine ayağa kaldırır, Jenkins server kurar.

Amazon’da çalışıyorsanız ve credential tanımladıysanız provider bloğunda bulunan profile (3.satır) satırını siliniz. Access key ve secret key giriniz, ya da AWS configure yapınız.

**207.main.tf**

4 makine ayağa kaldırır. Birincisine (Control node’a) ansible, ikincisine nodejs ve docker, üçüncüsüne react ve docker, dördüncüsüne postgresql ve docker yükler.

**Main.tf:** Launch template, load blancer, listener, target group, autoscaling group db instance file’larını oluşturacak şekilde hazırlanmış bir terraform dosyası.

**Provider.tf:** AWS ve Github’ı kullanabilmek için hazırlanmış bir provider dosyası.

**maven-java-terraform-template:** İçinde Java-11 and Maven yüklenmiş olan bir EC-2 instance oluşturur

**sec-gr.tf:** üç tane security grup oluşturur, severlar, db instance ve load balancer için.

**bookstore.tf:** Providerlar AWS ve Github. Bir tane EC2 instance ayağa kaldırır, bir github repo ve github files oluşturur. Uygulama dosyalarını önce github’da oluşturduğu repoya yükler sonra oluşan EC2’ya (curl komutu ile) çeker. EC2’ya python, flask ve docker yükler. Önce docker build ile imaj oluşturur, sonra docker compose up ile iki container ayağa kaldırır (database ve webserver) ve app çalışır.