honebook-app.py 13.satır **>>** Bu sefer RDS Database benim yapacağım CloudFormation sonucu oluşacak. Yani endpoint de o zaman oluşacak. (Database’in endpoint’i)

Phonebook-app.py 15.satır **>>** Buraya bir tane password girmemiz lazım. Bunu da CloudFormation’da RDS oluştururken yapacağım şifre ile aynı şifre olmasına dikkat edeceğim.

Fonksiyonlar üstte tanımlanmış. Aşağıda da decorater’ların içinde kullanmışlar.

CloudFormation’da ne yapacağız? İşlemlerin sonunda 3 tane security group oluşturacağız. (ALBSecurityGroup, WebServerSecurityGroup (Instance’lar) ve myDBSecurityGroup)

Sonra bu yükü dağıtması için bir load balancer oluşturacağız. (Application Load Balancer)

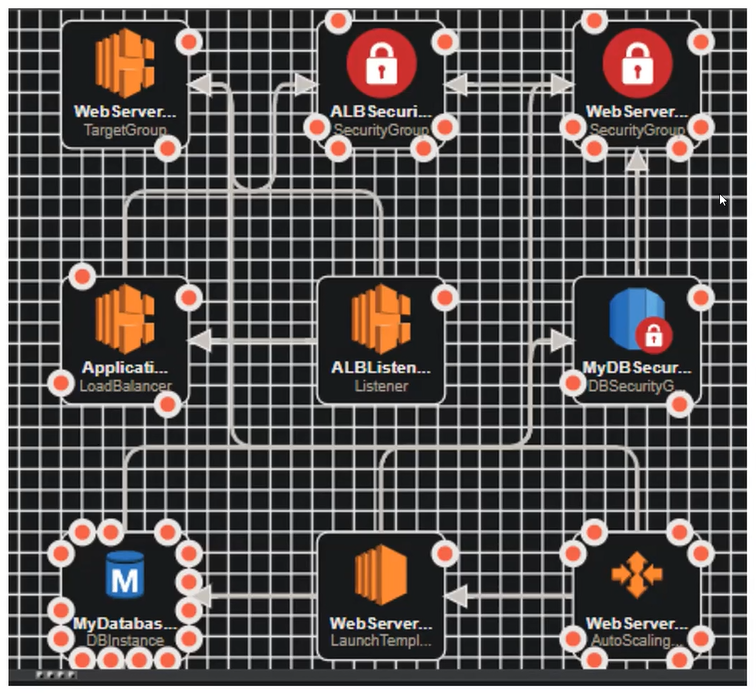
Bu load balancer’ın ne yapması lazım. Bir yeri dinlemesi lazım. ALB Listener oluşturacağız.

Sonra da bu dinlediği traifiği bir target group’a göndermesi lazım. Yani bu taget group’u da oluşturacağız. (WebServer Target Group)

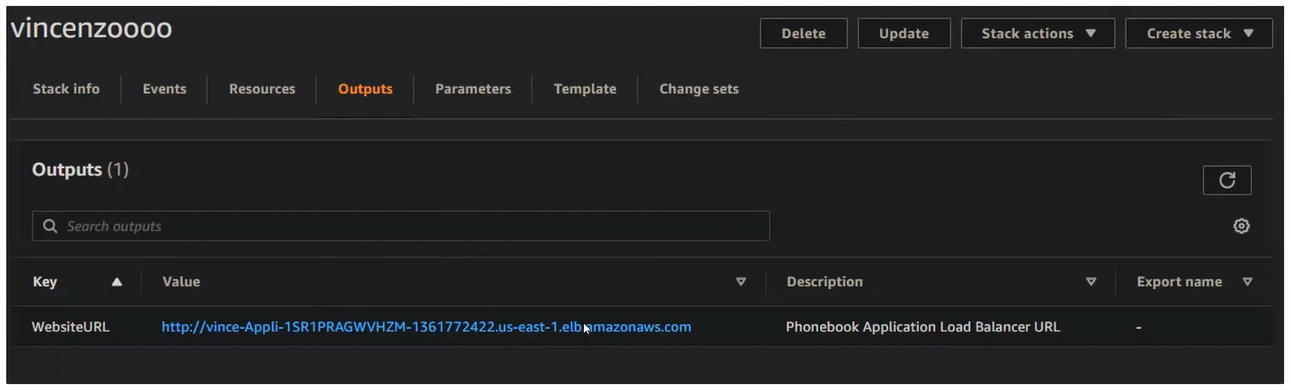
Bu target group’lar instance olacağı için bu instance’ları da ayağa kaldıracak Auto Scaling oluşturacağız.

Bu instance’lar da kafasına göre (gelişigüzel) ayağa kalkmayacak. Neye göre oluşacak? Yine bizim vereceğimiz talimatlara göre oluşacak Launch Template’lerden oluşacak.

Tabi ki bir de database oluşturacağız.



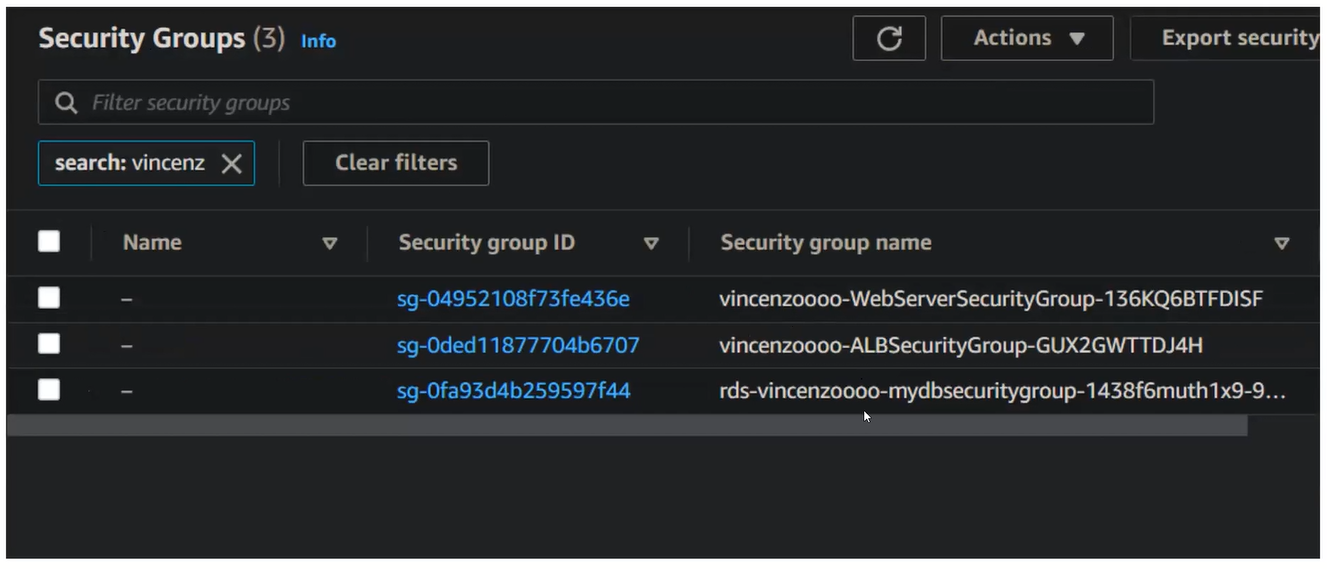
Resource’umuz oluştuğunda çıktısı load balancer’ın DNS name’i olacak. Outputs kısmına bunu yazacağız/ yazdıracağız. Git load balancer’ın DNS name’ini al diyeceğiz. Output’a tıkladığımız zaman telefon rehberi uygulaması karşımıza çıkacak.



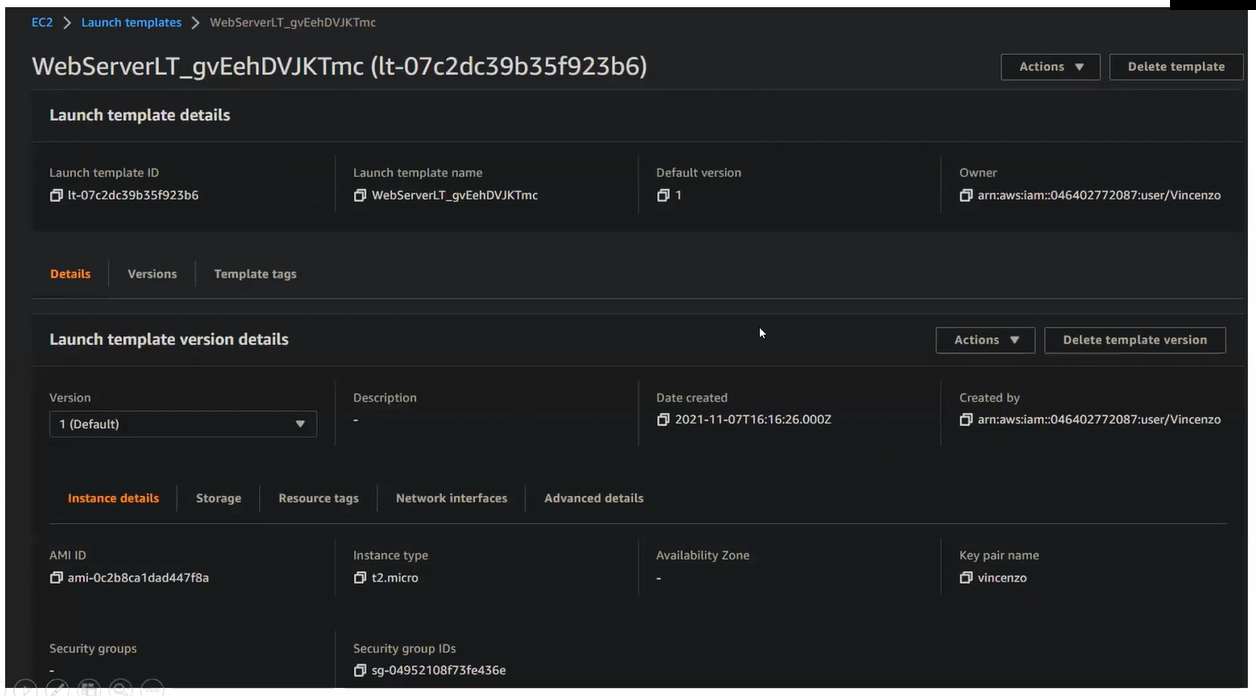
3 tane security group oluşturacağız. Birisi WebServerSecurityGroup (Instance’lar için)

Birisi ALBSecurityGroup (Load Balancer için)

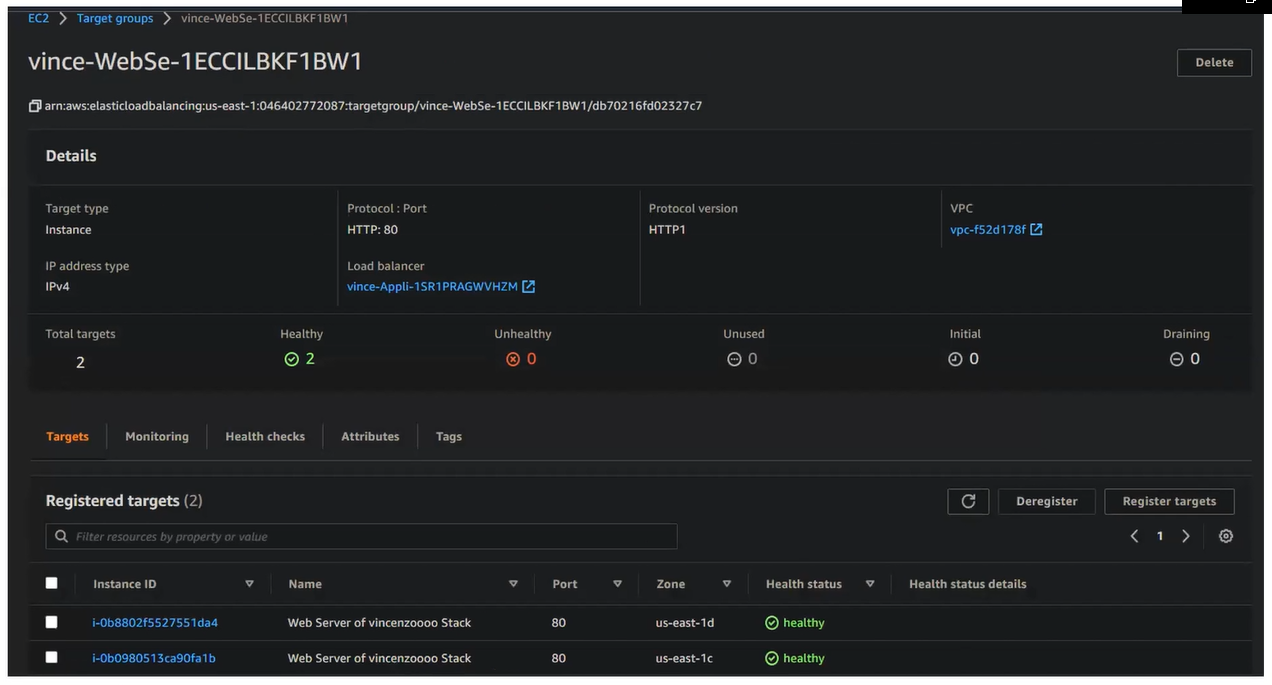
Bir diğeri de mydbsecuritygroup (RDS için)



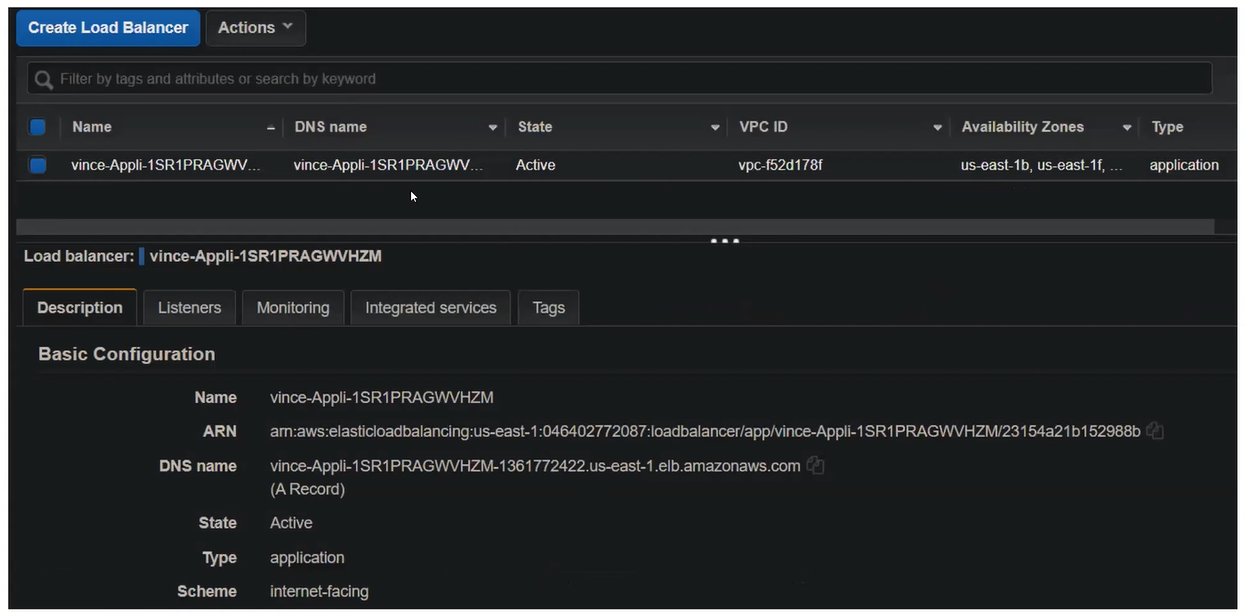
Launch Template ile Auto Scaling Group’a talimat vereceğiz. EC2 Instance’lar buna göre oluşturulacak. Oluşturulacak EC2’ların hangi AMI ID’sine sahip olacaklarını, hangi tipte olacaklarını (Ör: Instance type t2.micro), security group’larını, keypair’lerini launh template’de belirleyeceğiz. Ona göre Auto Scaling Group da EC2 Instance’ları oluşturacak.



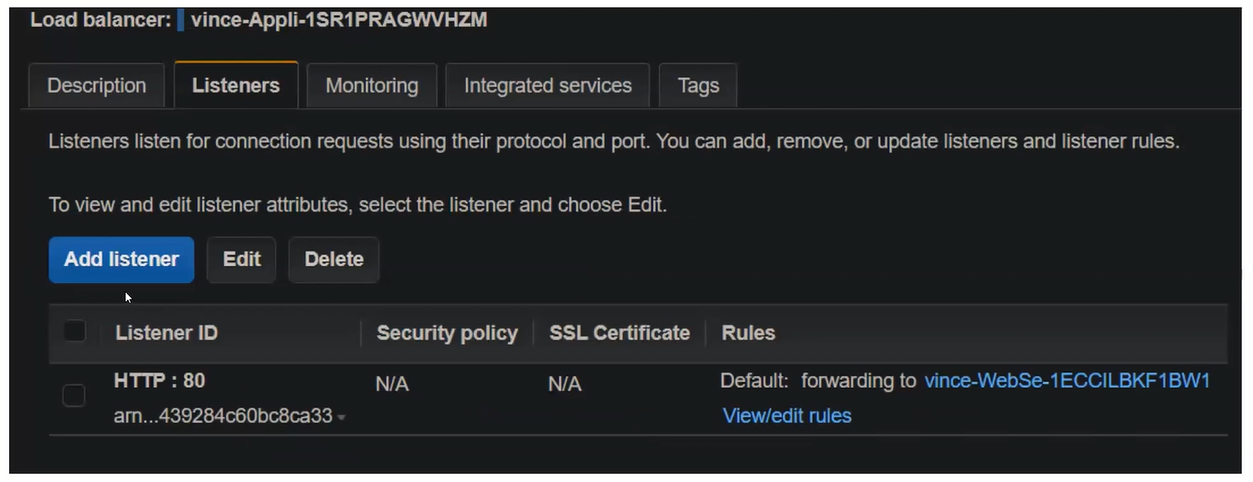
Sonra load balancer target group’lara trafiği yönlendirecek. Bu target group’ların target type’i instance olacak. Portu HTTP:80 olacak.



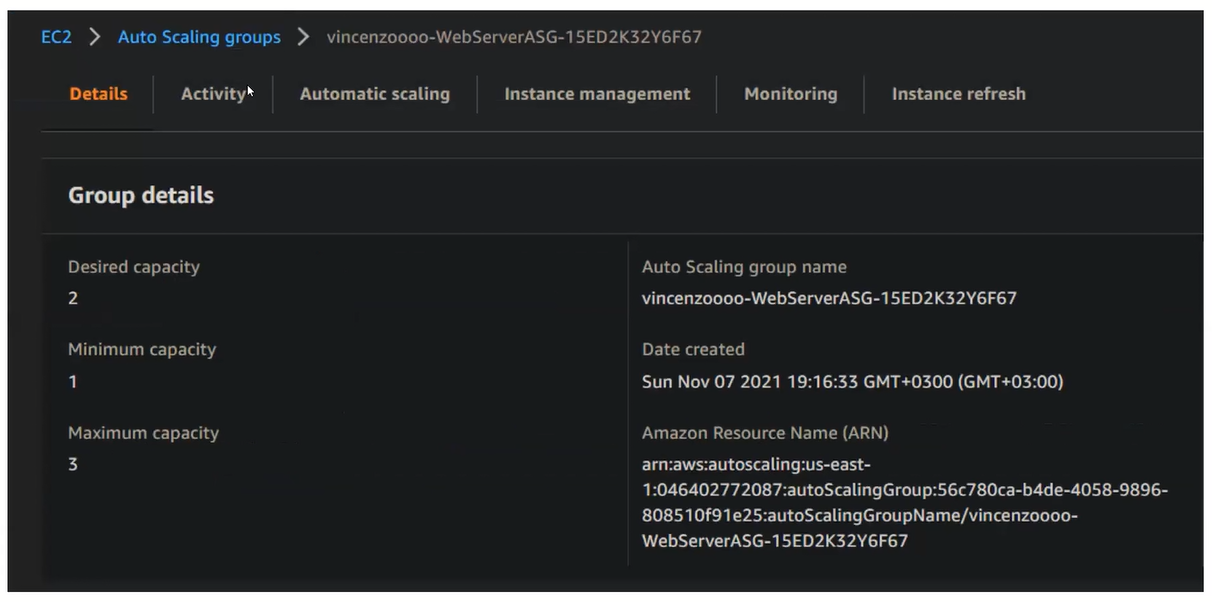
Load Balancer’ımız oluştuğu zaman aşağıdaki gibi bir DNS name’i verecek. Bunu outputs’a yazdıracağız.



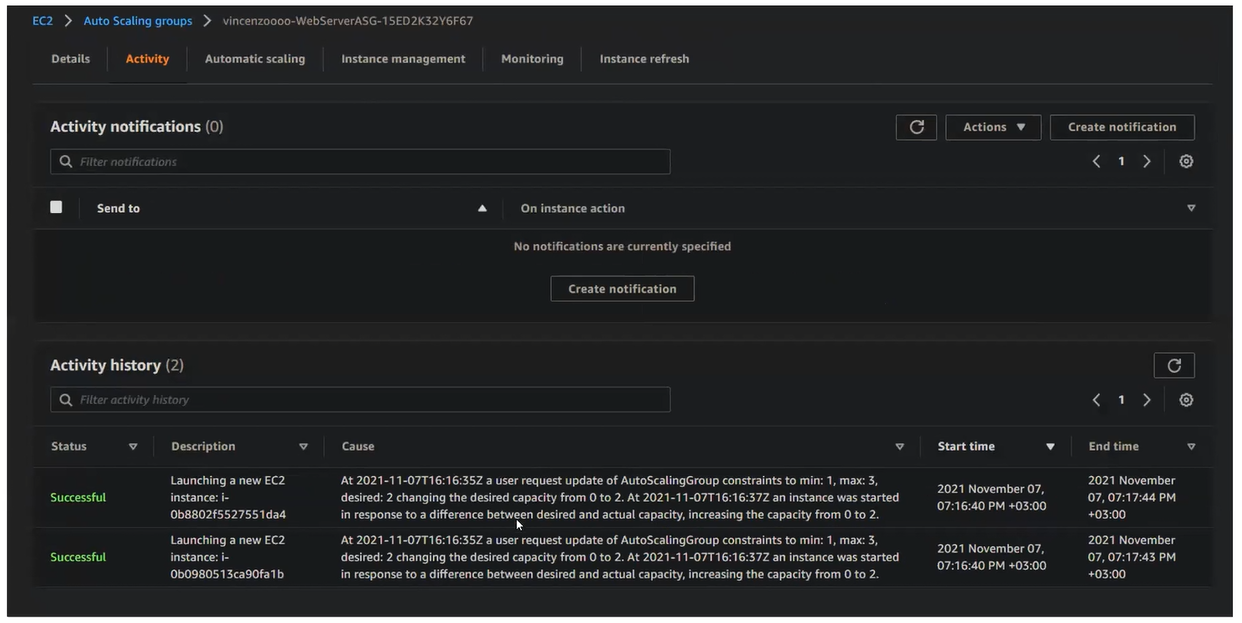
Load balancer’ın listener’ı http 80 portunu dinleyecek.



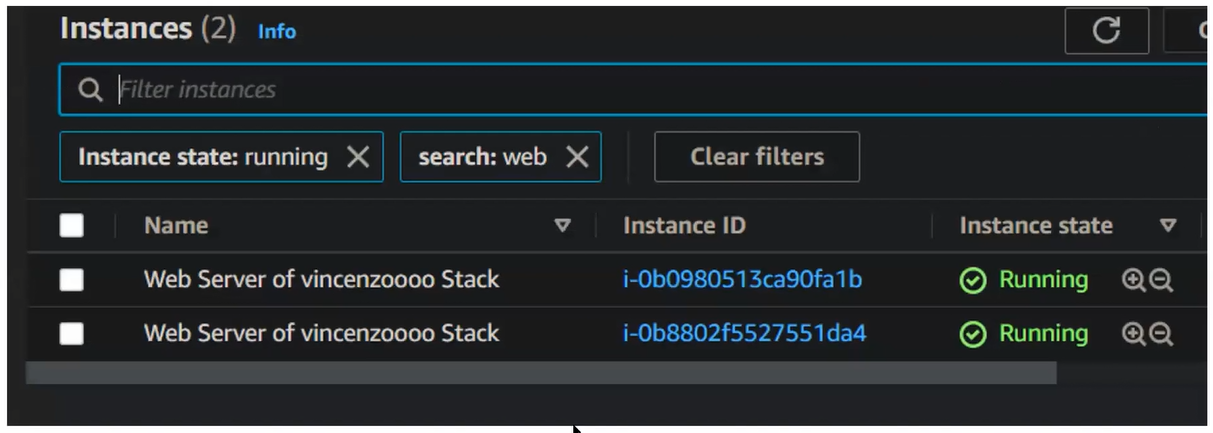
Auto Scaling Group oluşturacağız. (Desired, minimum ve maximum capacity’ye göre)



Sonra verdiğimiz talimatlara göre aşağıda görüleceği üzere 2 tane instance oluşacak.



Instanc’larımız stack’in ismini alacak. (Hiçbir şeyi manuel yapmıyoruz. Load Balancer ve Auto Scaling Group hepsini ayarlayacak.)



Database oluşturacağız. (Engine tipi: MySQL Community, DB Identifier: vincenzo-db-3, db’in instance tipi: db.t2.micro, security grubunu da seçeceğiz. Sonra da bize endpoint verecek. Bu endpoint’i user data kısmında çektireceğiz.)

