



#### KN05: Skalierung

- A) Installation App
- B) Vertikale Skalierung
- C) Horizontale Skalierung
- D) Auto Scaling

## KN05: Skalierung

Beachten Sie die [allgemeinen Informationen zu den Abgaben](#).

In dieser Kompetenz werden Sie eine Instanzen horizontal, vertikal und automatisiert skalieren.

[Lesen Sie sich in die Theorie ein, auch in die Dokumente in der Quelle.](#)

### A) Installation App

Im Ordner für [KN05](#) finden Sie Cloud-init Dateien. Diese Dateien installieren einen Datenbank- und einen Webserver für entweder eine .NET oder eine Java Applikation. Speziell dabei ist, dass sowohl die .NET Applikation als auch die Java Applikation mit eigenem Webserver kommen, die auf Port 5000, resp. 5001 hören. Bisher hatten wir Apache als Web-Server verwendet. Nun werden wir Nginx einsetzen. Sie möchten ja nicht, dass ihre Applikation nur über den Port 5000, resp 5001 aufgerufen werden kann, sondern über den normalen Port 80. Nginx wird daher als [Reverse Proxy](#) eingesetzt, welcher die Anfragen weiterleitet.

#### 1. Installieren Sie den Datenbankserver:

1. Verwenden Sie Ubuntu mit t2.micro und 8GB Harddisk.
2. Verwenden Sie das in **KN05** erstellte Netzwerk-Interface
3. Verwenden Sie die vorgegebene Cloud-Init Konfiguration. **Achtung:** Sie müssen gewisse Platzhalter ersetzen.

#### 2. Installieren Sie den Webserver

1. Verwenden Sie Ubuntu mit t2.micro und 8GB Harddisk.
2. Verwenden Sie das in KN05 erstellte Netzwerk-Interface mit der öffentlichen IP.
3. Entscheiden Sie sich für die .NET oder Java Applikation. Es handelt sich bei beiden Applikationen um ein Web-API mit Swagger aktiviert.
4. Verwenden Sie die vorgegebene Cloud-Init Konfiguration. **Achtung:** Sie müssen gewisse Platzhalter ersetzen.

Es kann sein, dass Sie ein paar Minuten warten müssen bis alles installiert ist.

#### .NET Applikation

- Infos zum Nachlesen: <https://learn.microsoft.com/en-us/aspnet/core/host-and-deploy/linux-nginx?view=aspnetcore-7.0&tabs=linux-ubuntu>
- Swagger-URL: <http://ihre-ip/swagger/index.html>

## Java Applikation

- Infos zum Nachlesen: <https://www.linode.com/docs/guides/how-to-deploy-spring-boot-applications-nginx-ubuntu-22-04/>
- Swagger-URL: <http://ihre-ip/swagger-ui/index.html>

### Abgaben:

- Screenshot der zeigt, dass Sie Swagger aufrufen können.
- Screenshot der zeigt, dass sie mindestens eines der API-Endpoints (mit DB-Zugriff) aufrufen können (via Swagger).

## B) Vertikale Skalierung

Skalieren Sie nun ihren Datenbankserver und erweitern Sie die Disk auf **20GB**. Dies machen Sie im laufenden Betrieb.

Skalieren Sie nun auch ihren Webserver und Verwenden Sie die den Instanztyp **t2.medium**.

### Abgaben:

- Vorher-Nachher Screenshots der Instanz-Ressourcen

## C) Horizontale Skalierung

Erstellen Sie einen Load Balancer und fügen Sie diesem zwei Instanzen des Web Servers hinzu. Suchen Sie **selbstständig** nach Ressourcen. Überdenken Sie ihre anderen Objekte wie *statischen IPs, Sicherheitsgruppen*, etc. Was macht nun Sinn in ihrer neuen Umgebung?

### Abgaben:

- Sie zeigen ihre laufende Umgebung der Lehrperson.
- Seien Sie bereit Auskunft zu erteilen über ihr Vorgehen und die erstellten Objekte wie *Load Balancer, Target Group, Health Check, IPs, Sicherheitsgruppen, Listener u. a.*

## D) Auto Scaling

Erstellen Sie ein Auto-Scaling, welches mit ihrem Load Balancer zusammenarbeitet. Als Ziel sollen jeweils 2 Instanzen verfügbar sein, maximal aber 5.

### Abgaben:

- Zeigen Sie ihre laufende Umgebung der Lehrperson. Sie dürfen die Umgebung aus C) hier erweitern, ohne, dass Sie vorher abgeben.
- Seien Sie bereit Auskunft zu erteilen über ihr Vorgehen und die erstellten Objekte wie *Template, Health Checks, u.a.*