

Teilnehmer: Denis Anderwald, Elias Kirchgässner

Inhaltsverzeichnis

- [1 Aufgabenbeschreibung](#)
- [2 Beschreibung der umgesetzten Funktionalität](#)
 - [2.1 Übersicht](#)
 - [2.2 Oberfläche](#)
 - [2.5 Datenbank](#)
 - [2.6 Server](#)
- [3 Analyse des Ergebnisses](#)

1 Aufgabenbeschreibung

Zur Anwendung der erlernten theoretischen Kenntnisse im Fach “Verteilte Systeme” soll ein klassischer Onlineshop mit Funktionalitäten erstellt werden. Dieser soll entsprechende Anforderungen erfüllen können:

- Onlineshop
 - Account & Login-System
 - Warenkorb, bestätigen und einsehen von Käufen
 - der Webshop soll verschiedene Anforderungen abbilden können
- CDN-System, welches die Ausfallsicherheit gewährleisten soll
 - Effiziente horizontale Skalierung Erklärung des Kunden: Website soll auch bei plötzlicher Lastzunahme stabil laufen können (zum Beispiel durch Referenzierung durch bekannte News-Seiten)
 - konstantes Repertoire an festen Server, schneller zuschalten zusätzlicher Server zum Ausgleich erhöhter Lasten
 - Ausfallfreiheit
 - Stabilität der Software auch bei fehlerhafter Nutzung
 - funktionierende Sessions

Abschluss der Arbeit durch eine Präsentation des Ergebnisses im Unterricht

Abschlusstermin: 14.04.2016

2 Beschreibung der umgesetzten Funktionalität

2.1 Übersicht

Für die Umsetzung des Verteilten Systeme-Projekts wurden verschiedene Technologien diskutiert.

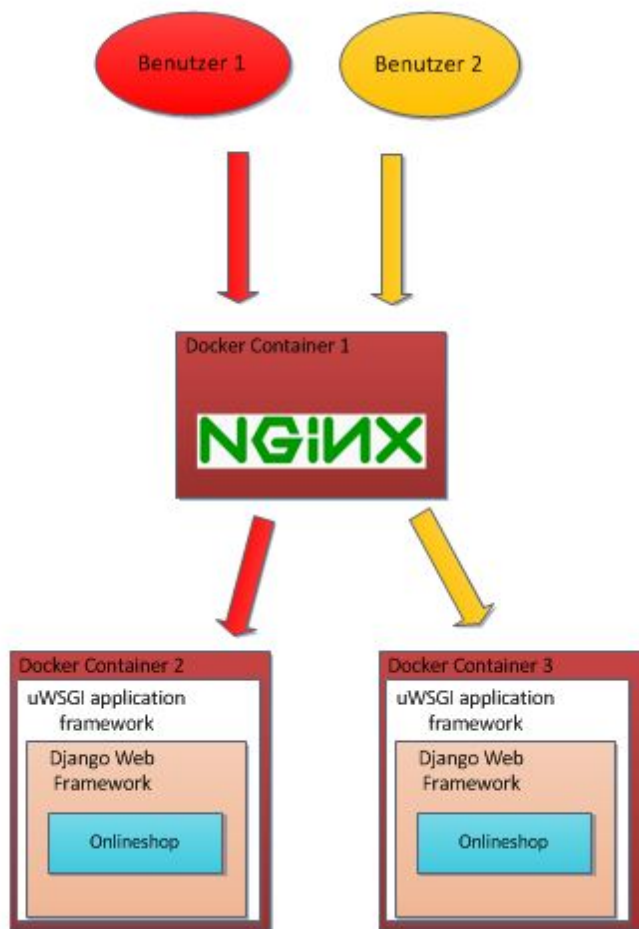


Abbildung 1: Übersicht über die Verteilung der Web-Applikationen mittels Load Balancer über Docker, Quelle: Eigene Abbildung

Verwendet wurden die folgenden Technologien, die später noch genauer dargestellt werden:

- Docker
- Django web framework
- Galera Percona Cluster
- uWSGI
- nginx load balancer

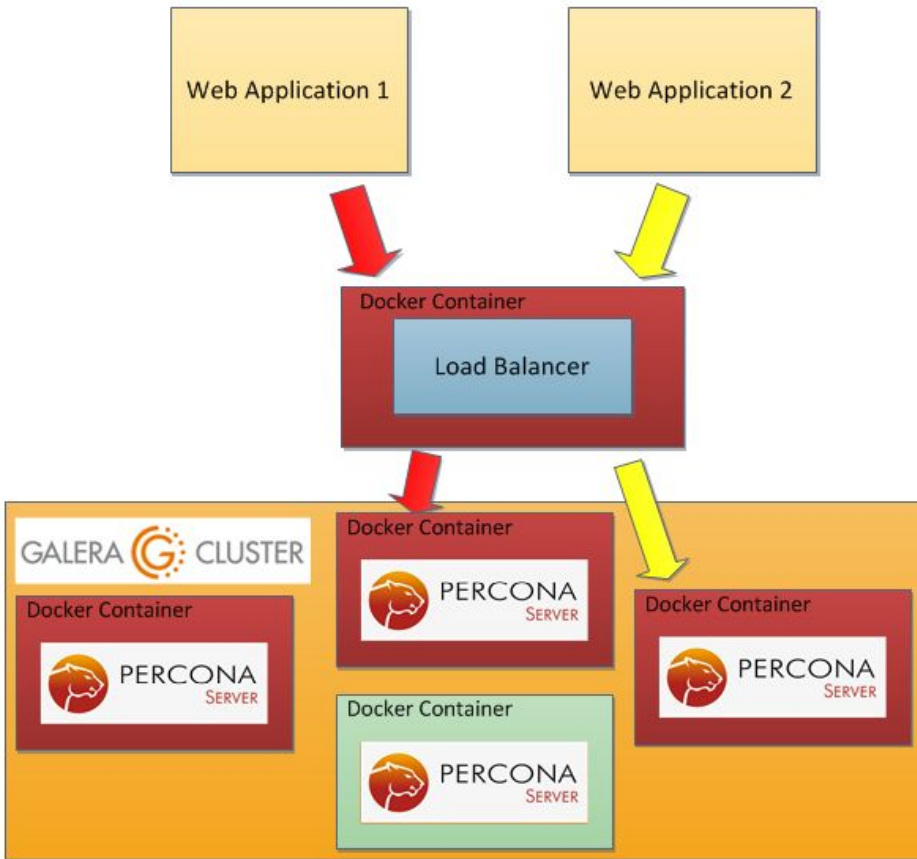


Abbildung 2: Datenbankanzbindung der Web-Applications über das Galera Cluster, Quelle: Eigene Abbildung

2.2 Onlineshop

Für die Erstellung des Onlineshops wurden die herkömmlichen HTML, CSS und JavaScript verwendet. Zur komfortablen und zeitoptimierten Durchführung der Implementierung wurde das Web Application Framework Django verwendet. Django basiert auf der Programmiersprache Python

Folgende Funktionalitäten werden im Onlineshop geboten:

- Darstellung unterschiedlichster Produkte
- Registrierung und Login mit Sessions durch gesetzte Cookies beim Nutzer
- einsehbarer und änderbarer Warenkorb, sowie Bestellungen

2.3 Datenhaltung

Zur Haltung und Bereitstellung der Daten wird ein Galera Percona Cluster verwendet. Hierbei sprechen die Webshop das Galera Cluster System an, welches anschließende Anfragen an die Percona Server weiterleitet. Diese sind modifizierte MySQL Datenbanken, welche

untereinander durch den Galera Cluster synchronisiert werden. Dieses System bietet eine automatische Lastverteilung und transparente Operationen von den Servern auf die Datenbanken. Außerdem sichert es den Ausfall des Systems ab, auch wenn einer oder mehrere Knoten nicht erreichbar sind. Die einzelnen Perconaserver können dabei alle als Master agieren und wählen einen, der als Hauptserver agiert. Fällt dieser aus kommt es zu einer neuen Wahlrunde.

2.4 Deployment

Für dieses Projekt wurden die Web-Server Software Nginx sowie Docker als Server-Komponenten verwendet. Nginx arbeitet dabei als Lastenverteiler, der mit Hilfe der Lastenverteilungs- sowie Proxy-Modulen die Benutzer auf einen entsprechenden Applikationsserver mit dem Onlineshop weiterleitet und verteilt. Nginx kennt dabei unterschiedliche Methoden, wie die Nutzer auf die bestehenden Server verteilt werden (siehe <https://www.nginx.com/resources/admin-guide/load-balancer/#method>) Dabei laufen Load Balancer Nginx, die Application Server mit dem Webshop sowie die Percona Datenbankserver und der Galera Cluster Load Balancer für Datenbanken jeweils auf Docker Containern.

3 Review

Unser System ist sehr gut skalierbar, da es über Docker leicht mit Docker Run erweitert werden kann. Zudem ist eine Zero Downtime gewährleistet. Unser System ist auch ausfallsicher, da es eine Vielzahl an Servern über Load Balancer nutzt. Bei Ausfall einer Systemkomponente werden die Benutzer automatisch umgeleitet. Über das Cluster, welches für die Datenbank verwendet wird, ist diese ebenfalls vor Überbelastungen geschützt.