

Hochschule München
Fakultät für Mathematik und Informatik

Seminararbeit

Konzeption und Prototyping eines hochskalierbaren Onlineshop

Maximilian Spelsberg

Abgabe: 31 August 2016

betreut von: Martin Häusl *Fachhochschule München*

Prof. Dr. Peter Mandl *Fachhochschule
München*

Inhaltsverzeichnis

1	Einleitung	4
1.1	Motivation	4
1.2	Problemstellung	4
1.3	Zielsetzung	5
1.4	Aufbau der Arbeit	5
1.5	Zeitplan	5
2	Einführung in die Architektur hoch skalierbarer Systeme anhand des Fallbeispiels Shopsystem	7
2.1	Microservices und die Servicearchitektur	7
2.1.1	Domain Driven Design??	7
2.1.2	Aufbau einer Servicearchitektur mittels Microservices	7
2.1.3	Servicestruktur des Shopsystems (Komponenten-, Laufzeit- und Verteilungssicht)	7
2.2	Resource-oriented Client Architecture (ROCA)	7
2.3	Datenhaltung innerhalb der Servicearchitektur (Polyglot Persistence)	7
2.4	Service Discovery??	7
2.5	Application Resiliency??	7
3	Konzeption und Implementierung der Services	8
3.1	Funktionsinhalt und Aufbau eines Microservices	8
3.1.1	Entwicklungsunterstützung mittels Frameworks	8
3.1.2	Aufbau eines Microservices für das Shopsystem	8
3.2	Implementierung der Microservices im Kontext des Shopsystems	8
3.2.1	Technologie- und Frameworkwahl	8
3.2.2	Implementierung der Komponente Artikel	8
3.2.2.1	Zweck und Verantwortlichkeit	8
3.2.2.2	Schnittstellenbeschreibung	8
3.2.2.3	Persistenz	8
3.2.3	Implementierung der Komponente ShoppingCart	8
3.2.3.1	Zweck und Verantwortlichkeit	8
3.2.3.2	Schnittstellenbeschreibung	8
3.2.3.3	Persistenz	8
3.2.4	Implementierung der Komponente Shop (Zentraler Service - bspw CMS)	8
3.2.4.1	Zweck und Verantwortlichkeit	8
3.2.4.2	Schnittstellenbeschreibung	8
4	Bereitstellung der Services	9
4.1	Integration (CI)	9
4.1.1	Einsatz eines Serviceregisters	9
4.2	Delivery (CD)	9

4.2.1	Einsatz von Featuretoggles oder Staging	9
4.3	Deployment (CD)	9
4.3.1	Einsatz und Konfiguration von Docker	9
5	Qualitätssicherung	10
5.1	Qualitätsziele und -szenarien im Kontext des Shopsystems	10
5.1.1	Festlegung der Qualitätsziele	10
5.1.2	Herangezogene Messmetriken	10
5.1.3	Bewertung des Shopsystems	10
6	Fazit	11
7	Ausblick	12
8	Literatur	13

1 Einleitung

Heutzutage wird im Zuge der Entwicklung von Software immer häufiger die Anforderung „hoch skalierbar“ gestellt. Doch hierbei stellt sich initial die Frage, in welchem Kontext Software skalieren soll. Skalierbarkeit kann unter anderem die Performanz betreffen. Hierbei soll die Software mit der variablen Anzahl an Anfragen skalieren und entsprechend auf einen Anstieg an Systemanfragen reagieren. Gleichzeitig kann auch im Kontext des Bauens und Ausliefern von Software eine hohe Skalierbarkeit angestrebt werden. Um Trends wie *Continuous Delivery* und *Continuous Deployment* angemessen begegnen zu können, gilt es die Effizienz bei der Auslieferung von Software hoch zu halten [vgl. Seite 113 Thö15]. Zudem kann auch eine funktionale Skalierbarkeit gefordert sein. Diese bezeichnet eine erleichterte Erweiterbarkeit des Systems, so das je nach Anforderung gewisse Funktionalitäten hinzugefügt oder auch abgehängt werden können. Dies soll dabei häufig auch noch in produktiven Systemen ohne Ausfallzeiten geschehen. Letzten Endes wird heutzutage fast immer mindestens eine Form von Skalierbarkeit für moderne Webanwendungen gefordert. Plattformen wie Ebay oder Amazon haben in Vergangenheit bereits häufig Maßnahmen zur ständigen Optimierung nicht-funktionaler Anforderungen wie Performanz, Ausfallsicherheit und Erweiterbarkeit ergriffen, da diese Anforderungen fundamental wichtig in Shopsystemen sind.

1.1 Motivation

Diese zuvor in der Einleitung beschriebene Skalierbarkeit gilt es in modernen Webanwendungen wie einem Webshop aufzugreifen und neben möglichen anderen nicht-funktionalen Anforderungen umzusetzen. Das geplante Shopsystem, das explizit eine hohe Skalierbarkeit aufweisen soll, ist hierbei von Grund auf neu zu planen und zu entwickeln. Dies gibt die Möglichkeit, bereits innerhalb der Konzeption entsprechende Maßnahmen zu ergreifen, um dieser Anforderung bestmöglich nachzukommen. Hierfür kann ein sehr aktueller Architekturstil mit entsprechenden Konzepten, Tools und Frameworks zum Einsatz kommen, die zur Erreichung der Zielsetzung führen sollen.

1.2 Problemstellung

Für moderne Systeme, wie Webanwendungen, wird heutzutage häufiger die Forderung nach Skalierbarkeit erhoben. Es gilt hierbei meist die individuelle Analyse, welche Architekturentscheidungen und verfügbaren Mittel den Anforderungen am geeignetsten entgegen. Außerdem gilt hierbei, dass Skalierbarkeit die Komplexität des Systems in den meisten Fällen erhöht und Probleme entstehen können, die mit einer anderen Architektur ggf. nicht einher gehen. Die Kenntnis über entsprechende Probleme und mögliche Fallstricke fehlen häufig noch in den Unternehmen, um eine entsprechende Architektur zu konzipieren bzw. diese entsprechend umzusetzen. Es werden daher innerhalb dieser Arbeit im Zuge der technischen Konzeption mögliche Fallstricke aufgedeckt und entsprechende Maßnahmen ergriffen bzw. erläutert.

1.3 Zielsetzung

Im Fokus der Arbeit steht die technische Konzeption und die hierauf aufbauende Implementierung eines prototypischen, hoch skalierbaren Shopsystems. Es wird daher schrittweise ein technisches Konzept erarbeitet, in dem Architektur und entsprechende Entwurfsentscheidungen erläutert sind. Im Zuge der Implementierungen ist zunächst ein generischer Microservice zu entwickeln (bzw. eine beispielhafte Vorgehensbeschreibung zur Erstellung eines Microservices bereitzustellen?), der die Grundlage für die in den Prototypen verwendeten Services darstellt und darüber hinaus auch in Zukunft für die Erweiterung des Systems durch weitere Services verwendet werden kann.

1.4 Aufbau der Arbeit

Um die beschriebenen Ziele zu erreichen, wird zunächst in Abschnitt 2 auf die von der Technologie unabhängige, grundlegende Architektur eingegangen, die im Zuge der Konzeption gewählt wurden. Im Anschluss wird in Abschnitt 3 auf die Konzeption einzelner Services und deren Implementierung eingegangen. Hier stehen vor allem Technologiewahl, Designentscheidungen und deren Umsetzung im Fokus. Darauf folgend soll in Abschnitt 4 die Bereitstellung der Services für den Systemserviceverbund erläutert werden. Hierbei stehen Integration, Auslieferung und optimale Bedingungen für Deployments erläutert. Abschließend wird der beschriebene Prototyp eines Webshops noch auf die ausreichende Umsetzung der geforderten Qualitätsziele analysiert und bewertet.

1.5 Zeitplan

Für die Zielerreichung der zuvor beschriebenen Ziele liegt ein bestimmter Zeitplan zugrunde. Dieser ist in Abbildung 1 mittels eines Gantt-Diagramms dargestellt bzw. in der Tabelle 1 noch detaillierter ausgeführt.

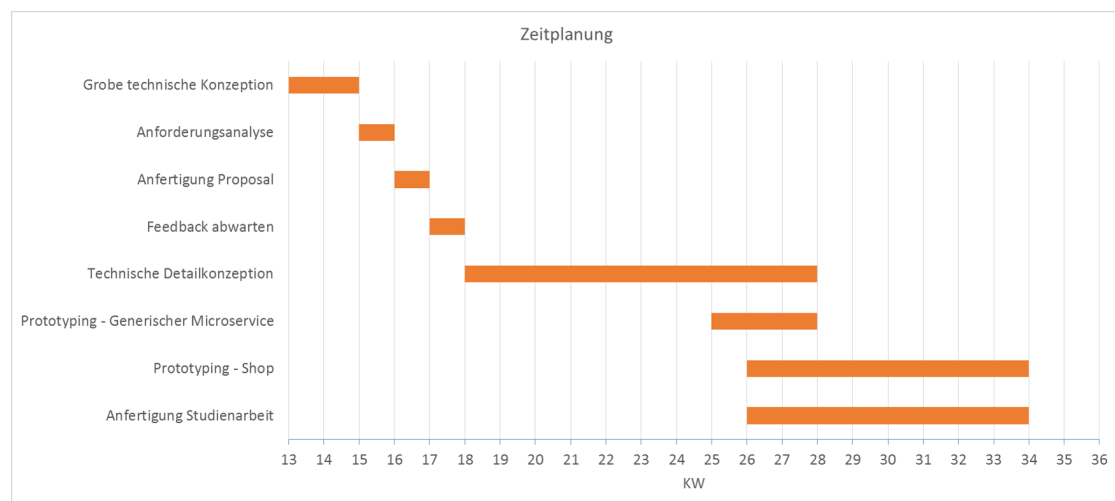


Abb. 1: Zeitplan für die Umsetzung der Zielsetzungen in Form eines Gantt-Diagramms

Aufgabe	Detailbeschreibung	Start KW	Ende KW
Feedback abwarten	Feedback und Rücksprache zu dem angefertigten Proposal abwarten und weiteres Vorgehen klären (Detailabsprache).	17	18
Technische Detailkonzeption	Anfertigung eines detaillierten technischen Konzepts auf Basis der fachlichen Anforderungen.	18	28
Prototyping - Generischer Microservice	Entwicklung und Test eines generischen Microservices in einer geplanten Testumgebung.	25	28
Entwicklung eines lauffähigen Shop-Prototypen	Entwicklung und Fertigstellung eines lauffähigen Prototypen, der grundlegende Funktionen eines Webshops besitzt und den geforderten Qualitätskriterien entspricht.	26	34
Anfertigung Studienarbeit	Finale Fertigstellung der Studienarbeit.	26	34

Tab. 1: Detailbeschreibung zu dem in Abbildung 1 dargestellten Gantt-Diagramm.

2 Einführung in die Architektur hoch skalierbarer Systeme anhand des Fallbeispiels Shopsystem

2.1 Microservices und die Servicearchitektur

2.1.1 Domain Driven Design??

2.1.2 Aufbau einer Servicearchitektur mittels Microservices

2.1.3 Servicestruktur des Shopsystems (Komponenten-, Laufzeit- und Verteilungssicht)

2.2 Resource-oriented Client Architecture (ROCA)

2.3 Datenhaltung innerhalb der Servicearchitektur (Polyglot Persistence)

2.4 Service Discovery??

2.5 Application Resiliency??

3 Konzeption und Implementierung der Services

3.1 Funktionsinhalt und Aufbau eines Microservices

3.1.1 Entwicklungsunterstützung mittels Frameworks

3.1.2 Aufbau eines Microservices für das Shopsystem

3.2 Implementierung der Microservices im Kontext des Shopsystems

3.2.1 Technologie- und Frameworkwahl

3.2.2 Implementierung der Komponente Artikel

3.2.2.1 Zweck und Verantwortlichkeit

3.2.2.2 Schnittstellenbeschreibung

3.2.2.3 Persistenz

3.2.3 Implementierung der Komponente ShoppingCard

3.2.3.1 Zweck und Verantwortlichkeit

3.2.3.2 Schnittstellenbeschreibung

3.2.3.3 Persistenz

3.2.4 Implementierung der Komponente Shop (Zentraler Service - bspw CMS)

3.2.4.1 Zweck und Verantwortlichkeit

3.2.4.2 Schnittstellenbeschreibung

4 Bereitstellung der Services

4.1 Integration (CI)

4.1.1 Einsatz eines Serviceregisters

4.2 Delivery (CD)

4.2.1 Einsatz von Featuretoggles oder Staging

4.3 Deployment (CD)

4.3.1 Einsatz und Konfiguration von Docker

5 Qualitätssicherung

5.1 Qualitätsziele und -szenarien im Kontext des Shopsystems

5.1.1 Festlegung der Qualitätsziele

5.1.2 Herangezogene Messmetriken

5.1.3 Bewertung des Shopsystems

6 Fazit

7 Ausblick

8 Literatur

- [Thö15] J. Thönes. “Microservices”. In: *IEEE Software* 32.1 (Jan. 2015), S. 116–116. ISSN: 0740-7459. DOI: 10.1109/MS.2015.11.