

Architektur einer Microservice-Anwendung und Deployment in OpenShift am Beispiel einer Partnerdatenbank

Inhaltsverzeichnis

zur

Masterarbeit

zur Erlangung des akademischen Grades
Master of Science in Engineering

Eingereicht von

Christoph Ruhsam, BSc

Betreuer: DI (FH) Thomas Reidinger, 3 Banken IT GmbH
Begutachter: FH-Prof. DI Dr. Herwig Mayr

November, 2019

Inhaltsverzeichnis

1	Einleitung ... 5.5 Seiten	1
1.1	Motivation zur Architektur von Microservices ... 1.5 Seiten	1
1.2	Motivation zum Einsatz von Cloudtechnologien ... 1.5 Seiten	1
1.3	Zielsetzung der Implementierung der Partnerdatenbank ... 1 Seite	1
1.4	Ziel des Deployment der Partnerdatenbank in OpenShift ... 1 Seite	1
1.5	Leitfaden und Gliederung der Schrift ... 0.5 Seiten	1
2	Serviceorientierte Architektur und Microservices ... 12.5 Seiten	2
2.1	Definition und Abgrenzung ... 4 Seiten	2
2.2	Vergleich zu monolithischer Architektur ... 2 Seiten	2
2.3	Charakteristiken ... 3 Seiten	2
2.4	Varianten ... 1.5 Seiten	2
2.5	Vor- und Nachteile ... 2 Seiten	2
3	Containerisierung mit Docker ... 6 Seiten	3
3.1	Docker ... 4 Seiten	3
3.2	Notwendigkeit von Containerisierung ... 2 Seiten	3
4	OpenShift ... 10.5 Seiten	4
4.1	Beschreibung von OpenShift ... 2 Seiten	4
4.2	Komponenten von Kubernetes ... 3.5 Seiten	4
4.3	Die OpenShift-Umgebung ... 3 Seiten	4
4.4	Fabric8 ... 2 Seiten	4
5	Partnerdatenbank ... 14.5 Seiten	5
5.1	Grundaufbau und Ziel der Partnerdatenbank ... 2.5 Seiten	5
5.2	Backend-Beschreibung ... 4 Seiten	5
5.3	Beschreibung der einzelnen Services ... 4 Seiten	5
5.4	Frontend-Beschreibung ... 4 Seiten	5
6	Design der Partnerdatenbank ... 11 Seiten	6
6.1	Microservice-Architektur ... 3 Seiten	6
6.2	Beschreibung der verwendeten Microservice-Technologien ... 6 Seiten	6
6.3	Design in OpenShift .. 2 Seiten	6
7	Implementierung der Partnerdatenbank ... 17 Seiten	7
7.1	Microservice-Architektur ... 2 Seiten	7
7.2	Automatisierte Test-, Build- und Deployment-Pipelines mit Jenkins ... 3 Seiten	7
7.3	Fehlerbehandlung mit Microprofile ... 2 Seiten	7
7.4	REST-Schnittstellenbeschreibung mit Swagger ... 2 Seiten	7
7.5	Tracing mit Jaeger ... 1 Seite	7

7.6	Einsatz von Docker zur Containerisierung der Anwendung ... 1 Seite	7
7.7	Konfiguration und Deployment-Deskriptoren von OpenShift ... 4 Seiten	7
7.8	Deployment in OpenShift mit Fabric8 ... 2 Seiten	7
8	Evaluierung der Anwendung ... 5.5 Seiten	8
8.1	Evaluierung des Frontend ... 1.5 Seiten	8
8.2	Unittests ... 1 Seite	8
8.3	Integrationstests ... 1 Seite	8
8.4	Architekturevaluierung ... 2 Seiten	8
9	Zusammenfassung und Ausblick ... 2-3 Seiten	9
10	Quellenverzeichnis	10

Kapitel 1

Einleitung ... 5.5 Seiten

- 1.1 Motivation zur Architektur von Microservices ... 1.5 Seiten
- 1.2 Motivation zum Einsatz von Cloudtechnologien ... 1.5 Seiten
- 1.3 Zielsetzung der Implementierung der Partnerdatenbank ...
1 Seite
- 1.4 Ziel des Deployment der Partnerdatenbank in OpenShift ...
1 Seite
- 1.5 Leitfaden und Gliederung der Schrift ... 0.5 Seiten

Kapitel 2

Serviceorientierte Architektur und Microservices ... 12.5 Seiten

2.1 Definition und Abgrenzung ... 4 Seiten

2.2 Vergleich zu monolithischer Architektur ... 2 Seiten

2.3 Charakteristiken ... 3 Seiten

2.4 Varianten ... 1.5 Seiten

2.5 Vor- und Nachteile ... 2 Seiten

Kapitel 3

Containerisierung mit Docker ... 6 Seiten

3.1 Docker ... 4 Seiten

3.2 Notwendigkeit von Containerisierung ... 2 Seiten

Kapitel 4

OpenShift ... 10.5 Seiten

4.1 Beschreibung von OpenShift ... 2 Seiten

4.2 Komponenten von Kubernetes ... 3.5 Seiten

4.3 Die OpenShift-Umgebung ... 3 Seiten

4.4 Fabric8 ... 2 Seiten

Kapitel 5

Partnerdatenbank ... 14.5 Seiten

5.1 Grundaufbau und Ziel der Partnerdatenbank ... 2.5 Seiten

5.2 Backend-Beschreibung ... 4 Seiten

5.3 Beschreibung der einzelnen Services ... 4 Seiten

5.4 Frontend-Beschreibung ... 4 Seiten

Kapitel 6

Design der Partnerdatenbank ... 11 Seiten

6.1 Microservice-Architektur ... 3 Seiten

6.2 Beschreibung der verwendeten Microservice-Technologien ... 6 Seiten

6.3 Design in OpenShift .. 2 Seiten

Kapitel 7

Implementierung der Partnerdatenbank ... 17 Seiten

- 7.1 Microservice-Architektur ... 2 Seiten
- 7.2 Automatisierte Test-, Build- und Deployment-Pipelines mit Jenkins ... 3 Seiten
- 7.3 Fehlerbehandlung mit Microprofile ... 2 Seiten
- 7.4 REST-Schnittstellenbeschreibung mit Swagger ... 2 Seiten
- 7.5 Tracing mit Jaeger ... 1 Seite
- 7.6 Einsatz von Docker zur Containerisierung der Anwendung ... 1 Seite
- 7.7 Konfiguration und Deployment-Deskriptoren von OpenShift ... 4 Seiten
- 7.8 Deployment in OpenShift mit Fabric8 ... 2 Seiten

Kapitel 8

Evaluierung der Anwendung ... 5.5 Seiten

8.1 Evaluierung des Frontend ... 1.5 Seiten

8.2 Unittests ... 1 Seite

8.3 Integrationstests ... 1 Seite

8.4 Architekturevaluierung ... 2 Seiten

Kapitel 9

Zusammenfassung und Ausblick ... 2-3 Seiten

Kapitel 10

Quellenverzeichnis