

songsMS

KBE-Abschlussbeleg SoSe2023

Harun Dastekin

28.09.23

Gliederung

1. Spring Cloud - Ein Überblick
2. Software Architektur Diagramm
 - ER-Modell
 - Schema der songsMS Datenbank
 - Sequenz-Diagramm
3. Zelkulon Microservice
4. Quellen

Was ist Spring Cloud?

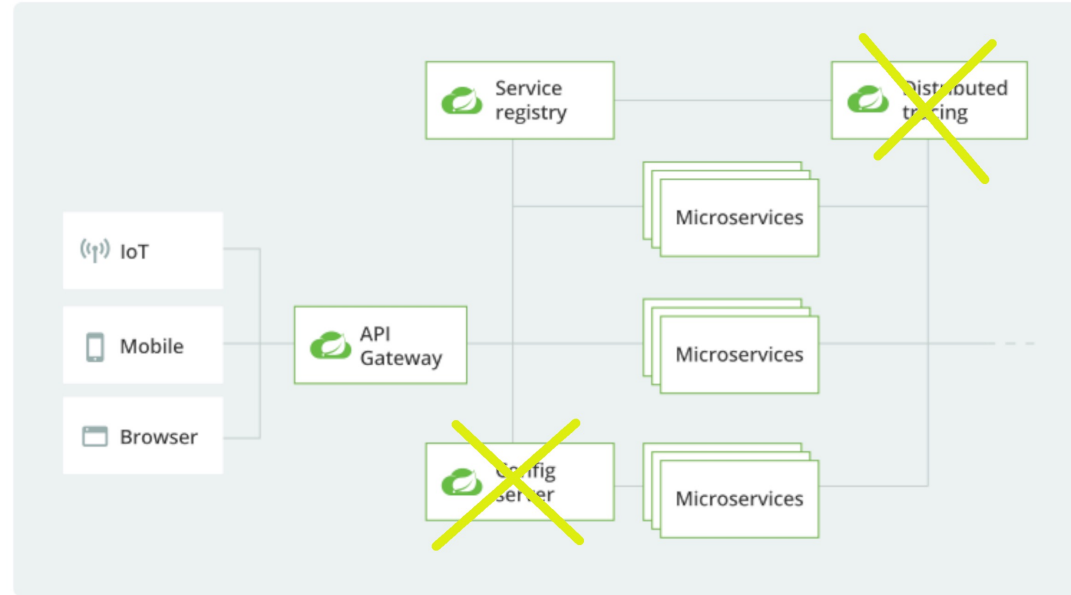
- Framework für Java-basierte Microservice-Anwendungen
- Baut auf Spring Boot auf

Hauptmerkmale

- **Service Discovery:** Integration mit z.B. Netflix Eureka.
- **Load Balancing:** Mit Ribbon.
- **Circuit Breaker:** Fehlertoleranz durch Hystrix.
- **Config Management:** Zentralisiert Verwaltung mit Spring Cloud Config

Warum verwenden?

- Schnelle Entwicklung und Skalierbarkeit.
- Unterstützt resiliente Microservice-Architekturen.

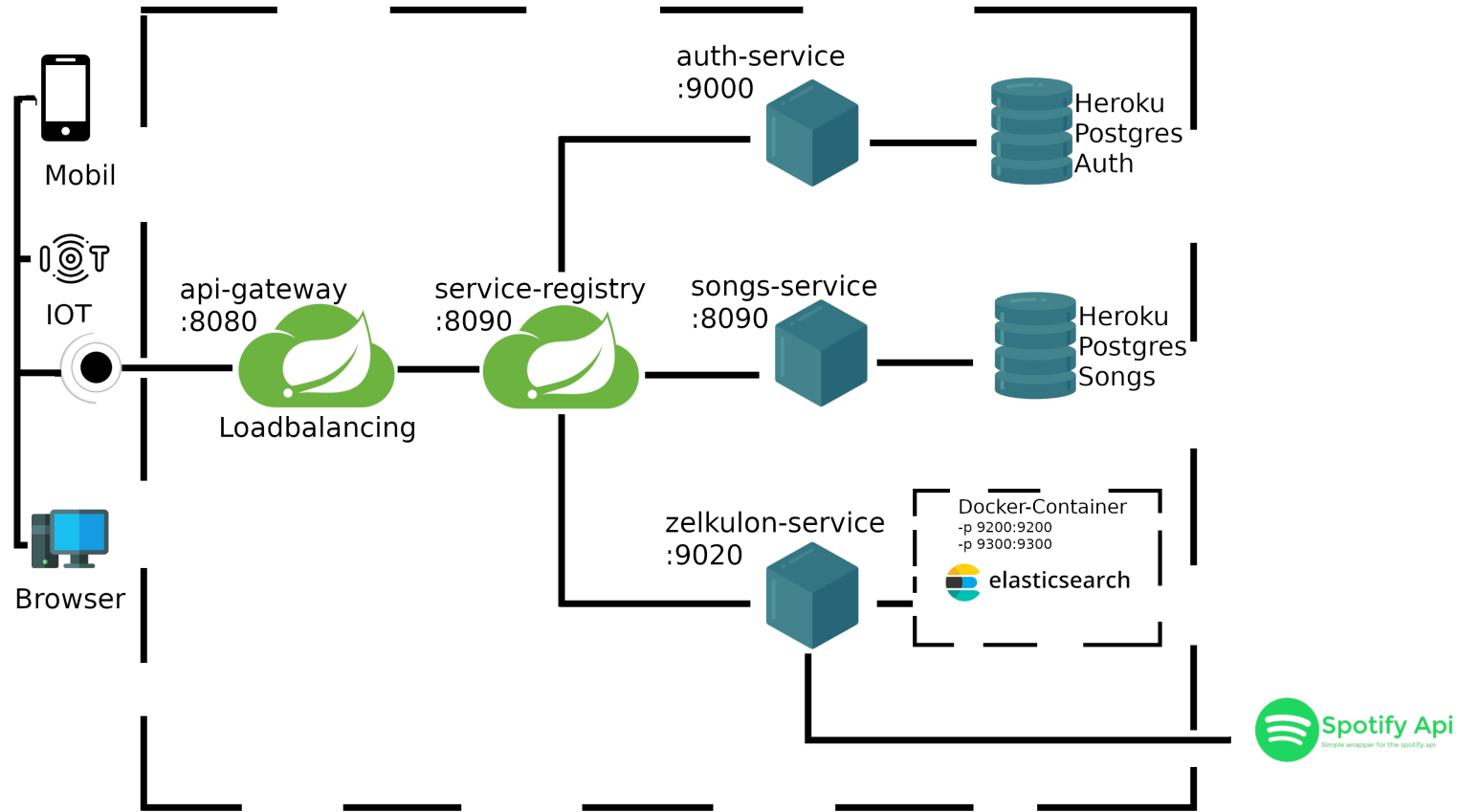


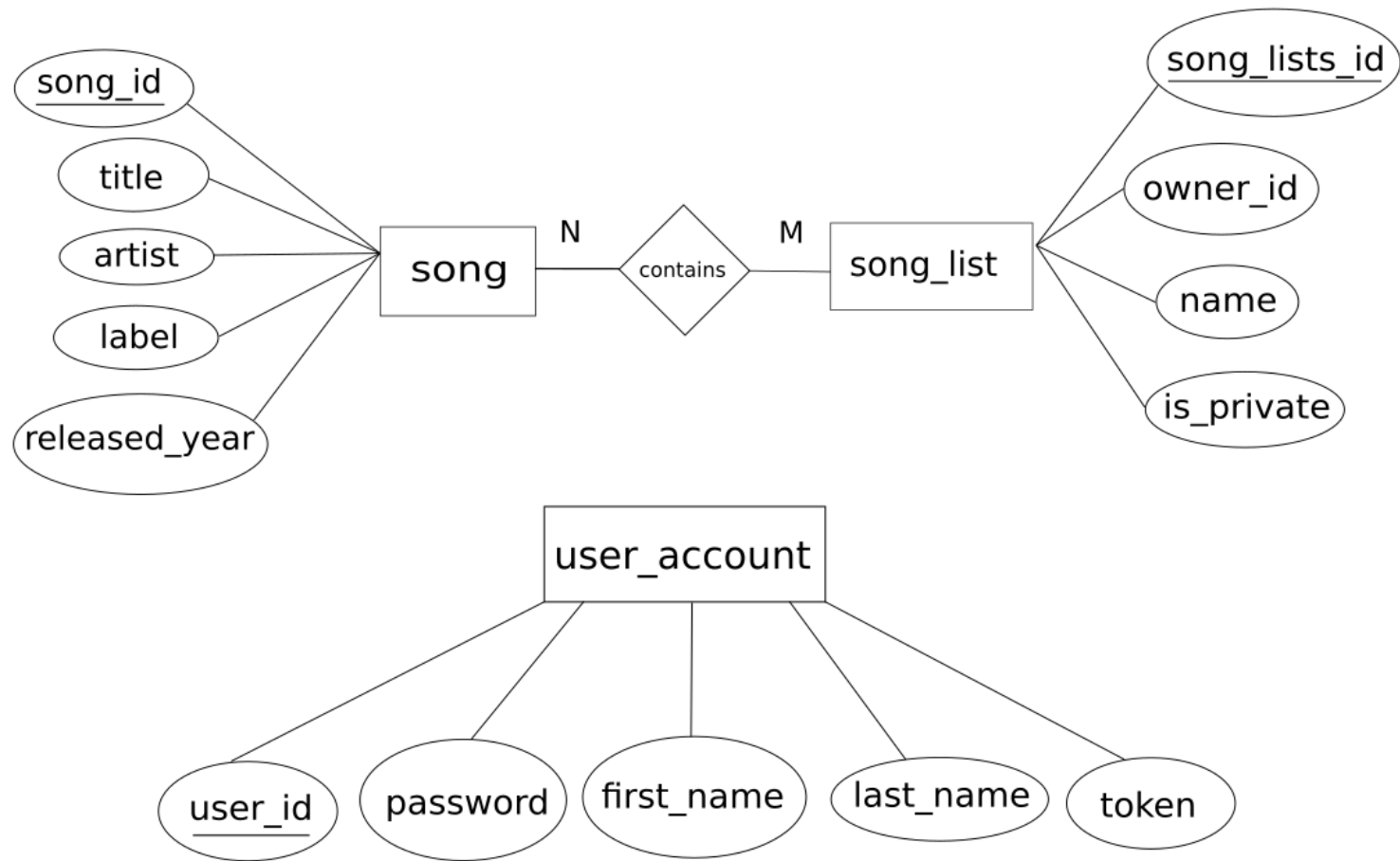
In SongMS:

service discovery:

spring-cloud-starter-netflix-eureka-server &
spring-cloud-starter-netflix-eureka-client

api gateway: spring-cloud-starter-gateway





Entitäten:

song: {[song_id:integer, title:string, artist:string, label:string, released:integer]}

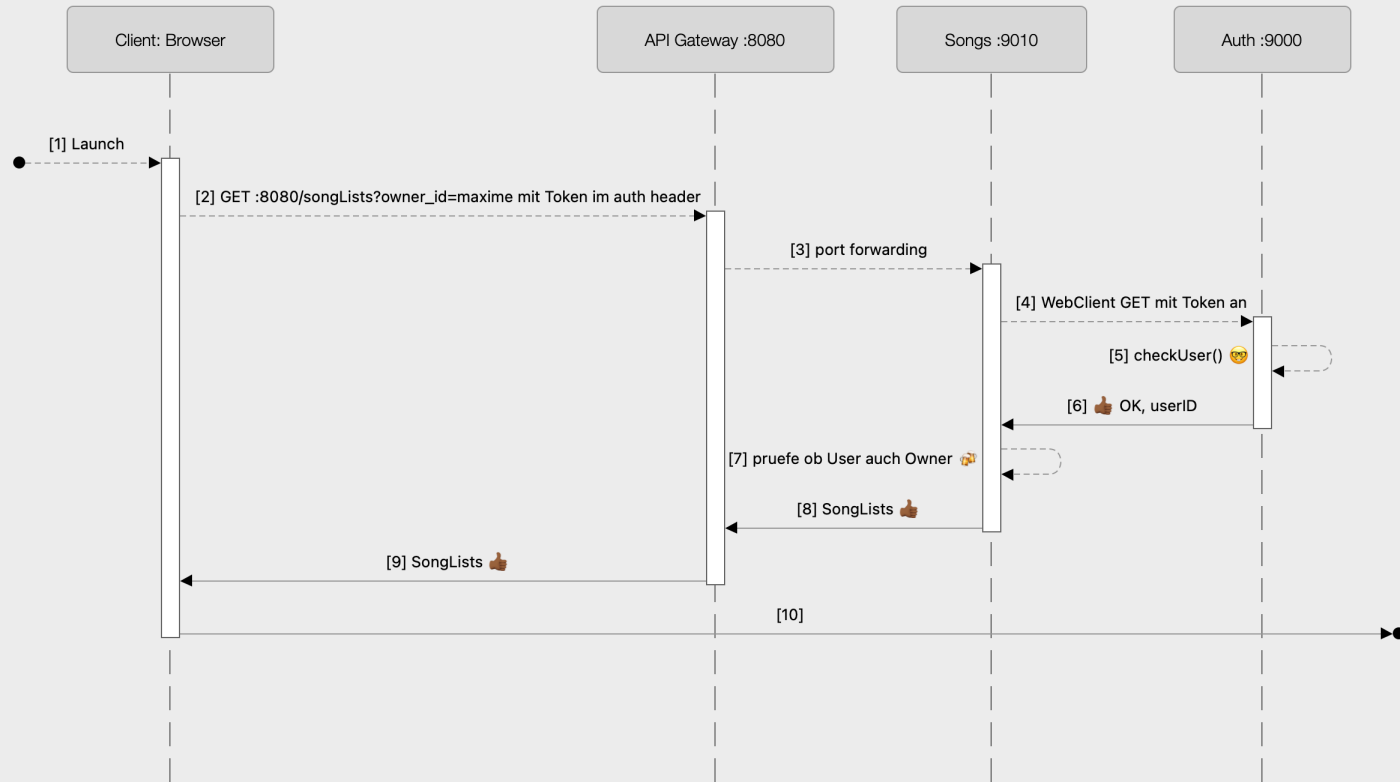
song_list: {[song_lists_id:integer, owner_id:string, name:string, is_private:boolean]}

user_account: {[user_id:string, password:string, first_name:string, last_name:string, token:string]}

Relationen:

contains: {[song_list_id:integer, song_id:integer]}

Maxime fragt ihre SongListen ab



Der Zelkulon Microservice: Live-Tracking und Archivierung Ihrer Spotify-Wiedergabe

Komponenten:

- Spring Cloud
- Spotify API
- Elasticsearch



Spotify Api
Simple wrapper for the spotify api



elasticsearch

Hauptfunktionen:

- **Live-Tracking:** Erfasst aktuelle Spotify-Wiedergabe alle 2,5 Minuten.
- **Archivierung:** Speichert jeden Song dauerhaft in Elasticsearch.
- **Künstlerstatistik:** Zeigt, wie oft ein Künstler gehört wurde.

Endpunkte:

- **Aktueller Song:** POST <http://localhost:8080/esong>
- **Alle Songs:** http://localhost:9200/songelastic/_search
- **Songanzahl eines Künstlers:** <http://localhost:8080/esong/count/{artist}>

Elasticsearch Dokumentstruktur:

```
@Document(indexName = "songelastic")
```

```
public class SongElastic {
```

```
    @Id
```

```
    private String id;
```

```
    private String title;
```

```
    private String artist;
```

```
    private String released;
```

```
}
```


1. Spotify Web API Java:

- Link: spotify-web-api-java.github.io

2. Elasticsearch:

- Hauptseite: elastic.co/de/elasticsearch
- Bildquelle: elastic.co/products/elasticsearch

3. Gateway:

- Video: [YouTube](#)

4. Spring Cloud:

- Hauptseite: spring.io/cloud
- Bildquelle: e4developer.com

5. Icons:

- Phone: Erstellt von [Freepik - Flaticon](#)
- Computer: Erstellt von [Freepik - Flaticon](#)
- lot: Erstellt von [Freepik - Flaticon](#)
- Cloud Gateway: clipartmax.com
- Module: Erstellt von [juicy_fish - Flaticon](#)
- Database: Erstellt von [juicy_fish - Flaticon](#)



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences

www.htw-berlin.de

