

Beleg 2: songsServlet

Servlet life cycle and thread safety

Fertigstellungstermin: Ihre Übung am Mi, 21. Nov., Do, 22. Nov. bzw. Mo, 26. Nov. 2018

User Story:

Als PO möchte ich einen Proof of Concept durchführen und einen einfachen Prototype für einen Webservice erstellen:

- Der Webservice soll die Servlet API nutzen.
- Der Webservice soll auf eine in-Memory Datensammlung mit Song-Einträgen zugreifen. Die Songsammlung befindet sich in der songs.json-Datei. Diese Datei soll beim Laden des Servlets gelesen und geladen werden.
- songs.json ist eine externe Datei und darf sich nicht im “resources”-Verzeichnis Ihres Maven-Projekts befinden.
- Wenn das Servlet vom Container entsorgt wird, soll der Inhalt Ihrer “Datenbank” in dieser Datei songs.json, die bei der Initialisierung des Servlets eingelesen wurde, in JSON-Format gespeichert werden.
- Der Webservice soll die HTTP-Methoden GET und POST unterstützen, und auf folgende Anfragen antworten:

GET `http://localhost:8080/songsServlet?all`
mit Accept-Header: * oder application/json oder ohne Accept-Header
soll alle gespeicherten Songs in JSON-Format zurücksenden

GET `http://localhost:8080/songsServlet?songId=6`
mit Accept-Header: * oder application/json oder ohne Accept-Header
soll den Song 6 in JSON-Format zurücksenden

GET-Requests mit Accept-Headern, deren Werte weder “*” noch “application/json” sind, sollen von ihrem Webservice mit dem entsprechenden HTTP-Statuscode abgewiesen werden

POST `http://localhost:8080/songsServlet` mit Payload

```
{
  "title" : "a new title",
  "artist" : "YOU",
  "album" : "First Album",
  "released" : 2017
}
```

Soll eine neue Id fuer den neuen Song generieren und den Song in der “DB” speichern. Die neue Id des Songs soll an den Client als Wert des “Location”-Header:

`http://localhost:8080/songsServlet?songId=newId` der Response zurückschicken. Der Response-Body ist leer. POST akzeptiert nur JSON payloads.

Die Servlet-Methoden, “init”, “doGet für einen Song”, “doPost” und deren Test-Cases müssen mit Unit-Tests abgedeckt sein. Die Unit-Tests werden bewertet.

Hinweis: **Test First!** Überlegen Sie sich erst alle Testfälle und schreiben Sie die entsprechenden Unit-Tests zuerst und dann den Servlet-Code.

Abnahme des Projektes: Ich werde ein Test-Script zusammenstellen, welches Anfragen an Ihren Webservice schickt und die Responses abspeichert. Wir werden uns Ihre Responses gemeinsam angucken und analysieren.