

# Spring Box Manager (V0.1)

## Start des Servers

Gestartet wird der Manager zum Beispiel durch

```
java -jar sbm.jar 136.199.51.196 1883 sudoku_db.json 1
```

Dabei sind die ersten beiden Argumente IP-Adresse und Portnummer des MQTT-Servers, das dritte Argument eine in JSON formatierte "Datenbank" mit Sudoku-Puzzeln und das letzte Argument die ID des Puzzles, das bei diesem Start an die Boxen verteilt werden soll.

An der angegebenen Adresse müsste - hoffentlich - **innerhalb der Uni** ein MQTT-Server zugreifbar sein.

Wer Docker installiert hat, kann sich seinen eigenen MQTT-Server ganz einfach selbst basteln:

```
docker run -d -p 1883:1883 --name mqtt eclipse-mosquitto:lates  
t
```

## Ausgaben

Der Manager ist aktuell recht geschwätzig. Er sendet alle 30 bzw. 60 Sekunden einen Heartbeat an den MQTT-Server. Gleichzeitig empfängt er alle an den MQTT-Server gesendeten Nachrichten einschließlich der Heartbeats (Topic-Wildcard #) und gibt sie auf der Console aus.

## REST API

Außerdem kann der Manager über das gemeinsam definierte API angesprochen werden:

- `http://127.0.0.1:4242/api/initialize` (Get)
- `http://127.0.0.1:4242/api/ready?box=sudoku%2Fbox_a1` (Get) (%2F ist das Escape für /)
- `http://127.0.0.1:4242/api/result` (Post)

Zusätzlich versteht er noch:

- `http://127.0.0.1:4242/api/reset` (Get), um ohne Neustart zurückgesetzt zu werden
- `http://127.0.0.1:4242/api/status` (Get), um seinen aktuellen Zustand auszugeben.

Letzteres ist aber noch rudimentär.

Die Portnummer 4242 ist aktuell noch eine Konstante im Sourcecode 😞.

## Stand der Dinge

Im Moment habe ich noch ein massives Problem, nach dem Empfang der 9 `Ready`-Nachrichten die entsprechende `Start`-Nachricht an den MQTT-Server zu senden. Ich möchte das in einer Bean, die den Zustand für das REST-API realisiert, mittels eines `ProducerTemplate` machen, das klappt aber irgendwie gar nicht. Ich denke gerade über einen alternativen Ansatz nach, wollte aber den Manager so schnell wie möglich für Ihre Experimente zur Verfügung stellen.