

# Workshop:

# Spring REST + Architecture

## Exercises

Add On: Spring REST + Architecture

### Exercise 1: Spring REST Intro

15 min

Reproduce the example under <https://spring.io/guides/gs/rest-service/>.

### Exercise 2: Product Client

25 min

In the presentation we have seen a product server. Now we need to develop an associated REST client to query and create products:

- Create queries as SpringBootApplication and as StandAloneApplication.
- Add validation in the original application. What would have to be changed?

### Exercise 3: Highscore REST Server

45 min

Create a highscore server to manage and query highscore data. Store the highscores in an in-memory DB. Provide the following endpoints:

- **POST „Highscore“**
- **GET /highscores**
- **GET /highscores/top5**
- **GET /highscores/byPerson/<name>**
- **GET /highscores/byLocalDate/<localdate>**
- **DELETE /highscores/<n>**
- **DELETE /highscores/of/<name>**
- **DELETE /highscores/on/< localdate >**

**Exercise 4: Create a User REST Server****45 min**

Consider a REST server for managing and querying user data that works exclusively with GET. The users are stored in an in-memory DB and provided by the following endpoints:

- `/create?email=[email]&name=[name]`: create a new user with an auto-generated id and email and name as passed values.
- `/delete?id=[id]`: delete the user with the passed id.
- `/get-by-email?email=[email]`: retrieve the id for the user with the given email address.
- `/update?id=[id]&email=[email]&name=[name]`: update the email and the name for the user identified by the given id.

Improve the application design by

- a) using matching status codes and HTTP verbs
- b) integration of a service
- c) and by using error handling
- d) Validation

Check validation and other actions with CURL, such as

```
curl -X POST -H "Content-Type: application/json"  
http://localhost:8080/users/ -d '{ "email" : "E-MAIL", "name" :  
"Micha" }'
```

And with a valid email:

```
curl -X POST -H "Content-Type: application/json"  
http://localhost:8080/users/ -d '{ "email" : "mail@mike.de", "name"  
: "Micha" }'
```