AARHUS UNIVERSITET

Software Engineering and Architecture

Exercise in REST

AARHUS UNIVERSITET

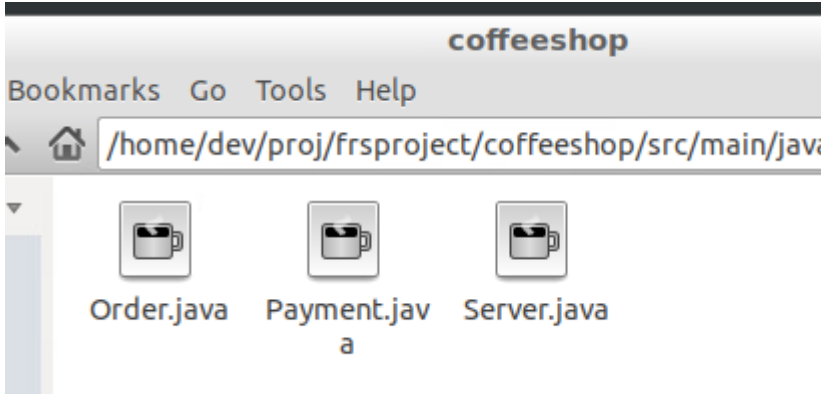
Level 2 REST

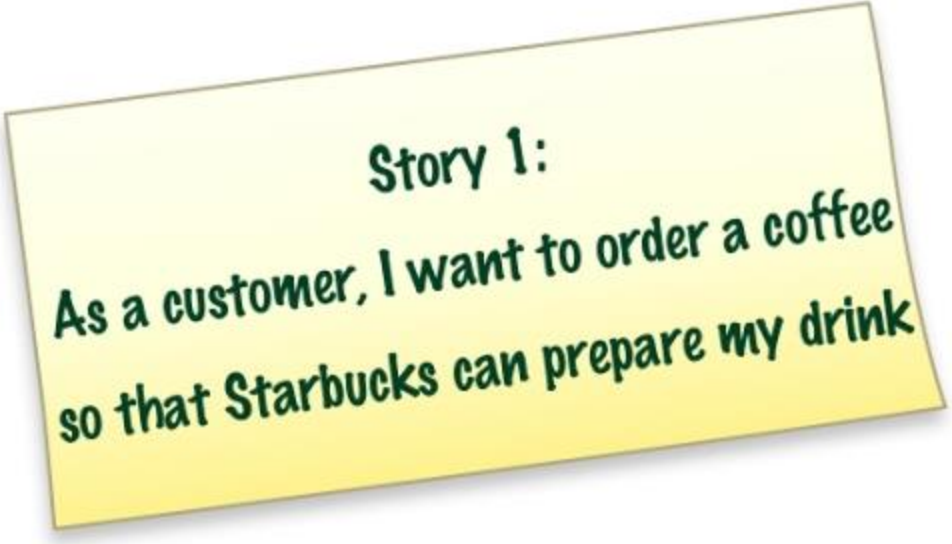
- The ‘CRUD on resources’ to model state transitions, I admit, takes a bit of time to get your head around ☺.
- We will try a small exercise, closely related to Webber et al.’s *How to GET a cup of Coffee* blog post
- Get hold of the ‘coffeeshop.zip’ from the weekplan.
  - Most info is also available in the README.md in the zip.

AARHUS UNIVERSITET

Story 1

- I have already implemented Story 1.





AARHUS UNIVERSITET

Story 1

- Our *highly user-friend client* is ‘curl’ as usual ☺
  - Sorry to folks that like clicking with a mouse…

```
dev@m1e18hbc:~/proj/frsproject/coffeeshop$ curl -i -X POST -d '{"drink":"espresso"}' localhost:4567/order
HTTP/1.1 201 Created
Date: Wed, 03 Oct 2018 11:42:30 GMT
Content-Type: application/json
Location: localhost:4567/order/100
Transfer-Encoding: chunked
Server: Jetty(9.4.6.v20170531)

{"paid":false,"drink":"espresso","cost":"3.00"}dev@m1e18hbc:~/proj/frsproject/coffeeshop$
dev@m1e18hbc:~/proj/frsproject/coffeeshop$ curl -i -X POST -d '{"drink":"americano"}' localhost:4567/order
HTTP/1.1 201 Created
Date: Wed, 03 Oct 2018 11:42:40 GMT
Content-Type: application/json
Location: localhost:4567/order/101
Transfer-Encoding: chunked
Server: Jetty(9.4.6.v20170531)

{"paid":false,"drink":"americano","cost":"3.00"}dev@m1e18hbc:~/proj/frsproject/coffeeshop$
dev@m1e18hbc:~/proj/frsproject/coffeeshop$
```

AARHUS UNIVERSITET

Story 1

- Review order by GET

```
dev@m1e18hbc:~/proj/frsproject/coffeeshop$ curl -i -X GET localhost:4567/order/100
HTTP/1.1 200 OK
Date: Wed, 03 Oct 2018 11:44:34 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Server: Jetty(9.4.6.v20170531)

{"paid":false,"drink":"espresso","cost":"3.00"}dev@m1e18hbc:~/proj/frsproject/coffeeshop$
dev@m1e18hbc:~/proj/frsproject/coffeeshop$ curl -i -X GET localhost:4567/order/101
HTTP/1.1 200 OK
Date: Wed, 03 Oct 2018 11:44:42 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Server: Jetty(9.4.6.v20170531)

{"paid":false,"drink":"americano","cost":"3.00"}dev@m1e18hbc:~/proj/frsproject/coffeeshop$
dev@m1e18hbc:~/proj/frsproject/coffeeshop$ curl -i -X GET localhost:4567/order/102
HTTP/1.1 404 Not Found
Date: Wed, 03 Oct 2018 11:44:49 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Server: Jetty(9.4.6.v20170531)

nulldev@m1e18hbc:~/proj/frsproject/coffeeshop$
```

AARHUS UNIVERSITET

Your Exercise

AARHUS UNIVERSITET

Implement Story 3

- Allow *paying* for the coffee
  - I.e. a state transition for the coffee order
  - HATEOAS: *You update a resource that represents the transition*
    - *Suggestion: payment/order/{id}*

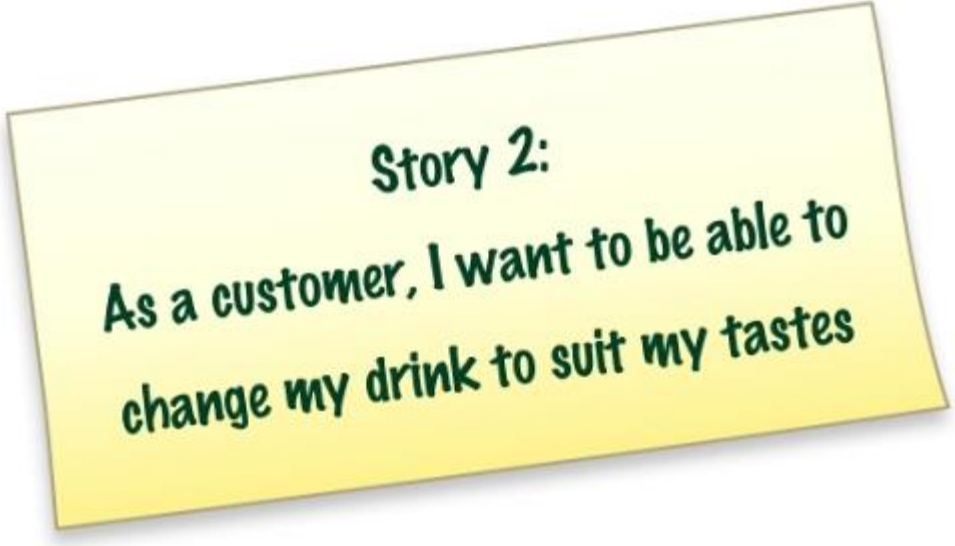
```
Story 3: Payment is not supported, but the Payment.java class template
is provided. As hint for the exercise, you should be able to PUT on the
payment resource along the lines of the following curl:

    curl -i -X PUT -d '{"cardno":"1234", "amount":"3.00"}' localhost:4567/payment/order/100

If there is an order with the given id, then the order is payed which
can be verified with another GET on the drink order. If not, appropriate
status code should be returned.
```

Note: Also change state of order to ‘paid’!

- Simplification:
  - No *next/links* section, we just assume URI /payment/ exists

Implement Story 2

- Implement the PUT, as outlined by Webber et al.



Add Stuff

- Validate proper payment, etc.
- Use correct status codes as return values