

## **Group 1 B-class: Ib Henrik Routhe, Martin Skou Nielsen and Simon Steinaa**

### **Overview: First Words from the group...**

We've focused mostly on making the software working and completing the tasks. The first two days went very well and we didn't fall behind...however the THIRD and FOURTH day we had problems which resulted in loss of efficiency and time thus limiting our completion of the last parts severely.

We chose to try solving our problems internally in the group before reaching our hands out for some teaching assistance.

As a result, We've learned A LOT in this CA – more than we could've learned from getting others to solve our problems.

### **What parts of the assignment we've implemented:**

Part 1:

- implemented all the parts – HOWEVER the openshift part for some reasons doesn't work now! (it worked friday 2 days ago – and we tested it there – everything, the database was working as well - but we now get a 404 error (and a weird MySQL exception when we try to restart the app from openshift web console – so NO openshift link)

Part 2:

- implemented all of the tasks

Part 3:

- implemented all of the tasks

Part 4:

- implemented all of the tasks

Part 5

- this project lacks lots of testing – we ran out of time and felt we need more knowledge on that – and we've decided only to do a few unit tests together with the postman api test.

Part 6:

- this project needs lots of more documentation - especially in all the areas of testing.

### **Testing REST API With postman**

In our CurrencyResource api we have two getters. One for returning all the daily currency rates in a Json string. The other returning a calculated result for the valuta converter function. Using postman, we can test these.

Typing in the link: <http://localhost:8080/AngSeedServer/api/currency/dailyrate>

We get following result  
---->

As one can see, we get an Json array filled with currency objects (in Json format).

GET <http://localhost:8080/AngSeedServer/api/currency/dailyrate>

Authorization Headers (2) Body Pre-request script

No Auth

Body Cookies Headers (4) Tests Status 200 OK Time 1037 ms

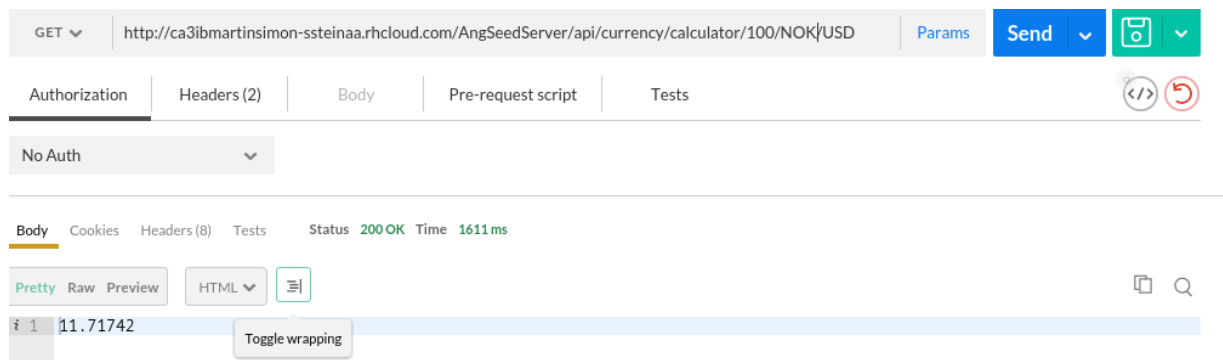
Pretty Raw Preview JSON

```
[
  {
    "Date": "2015/11/06",
    "Code": "ZAR",
    "Desc": "South African rand",
    "Value": "49.36"
  },
  {
    "Date": "2015/11/06",
    "Code": "SEK",
    "Desc": "Swedish kronor",
    "Value": "79.42"
  },
  {
    "Date": "2015/11/06",
    "Code": "USD",
    "Desc": "US dollars",
    "Value": "685.40"
  },
  {
    "Date": "2015/11/06",
    "Code": "CNY",
    "Desc": "Chinese yuan renminbi",
    "Value": "108.00"
  }
]
```

The next method takes in 3 parameters, which we type in the link:  
<http://localhost:8080/AngSeedServer/api/calculator/100/NOK/USD>

The first parameter is the amount that needs to be calculated. In this example its 100. The *from* currency is NOK and *to* currency is USD. The 3 inputs are gotten from the view, typed by the user.

The result we get in postman is this:



Here is a list of the rest api's we have:

User resources:	api/demouser	(POST)	note: creates new user on server.
Admin resources:	api/admin/{id}	(DELETE)	note: deletes user on the server.
Company resources:	api/company/	(POST)	note: gets company with inputs.
Currency resources:	api/currency/dailyrates	(GET)	note: gets all currency rates.
	calculator/{amount}/{fromCurrency}/{toCurrency}	(GET)	note: gets converted value.

## How to use the website

There are always 3 default users –

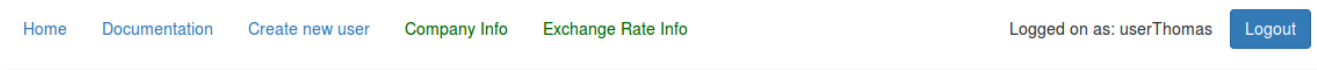
You can log in as a user or admin. Use one of these logins:

User:	“userThomas”	Password: “test”
Admin:	“adminLars”	Password: “test”
User and Admin:	“user_adminlb”	Password: “test”

even if you try to delete one of these they will be created again. So you can only delete other users in the system that you create. This makes sure, that all parts of the system can be used, even if you try to delete everything, which is good for safety.

### User login

When logging in as user, you have acces to 5 menu's: Home, Documentation, Create new user, company Info and Exchange Rate Info.



## Home

### Admin login

When logging in as user, you have access to 4 menu's. Home, Documentation, Create new user and All Users

[Home](#) [Documentation](#) [Create new user](#) [All Users](#)

Logged on as: adminLars

[Logout](#)

## Home

### Exchange Rate Info

Here we can convert one valuta to another. You type in the amount in the "from valuta" textfield. Choose the start currency code on the dropdown-list on the left, and the end currency code on the right. The result will be printed in the "to valuta" textfield, as soon as you type in a number in the "from value" textfield.

### Exchange Rate Info

#### Exchange Rate Converter

From Valuta   To Valuta

### Company Info

Here we can find companies, based on the inputs we choose.

#### Company Info

Search text

Options

Country

### Create new user

Here we can create a new user, that will be stored in the database.

### Create New User

Username:

Password:

Result:

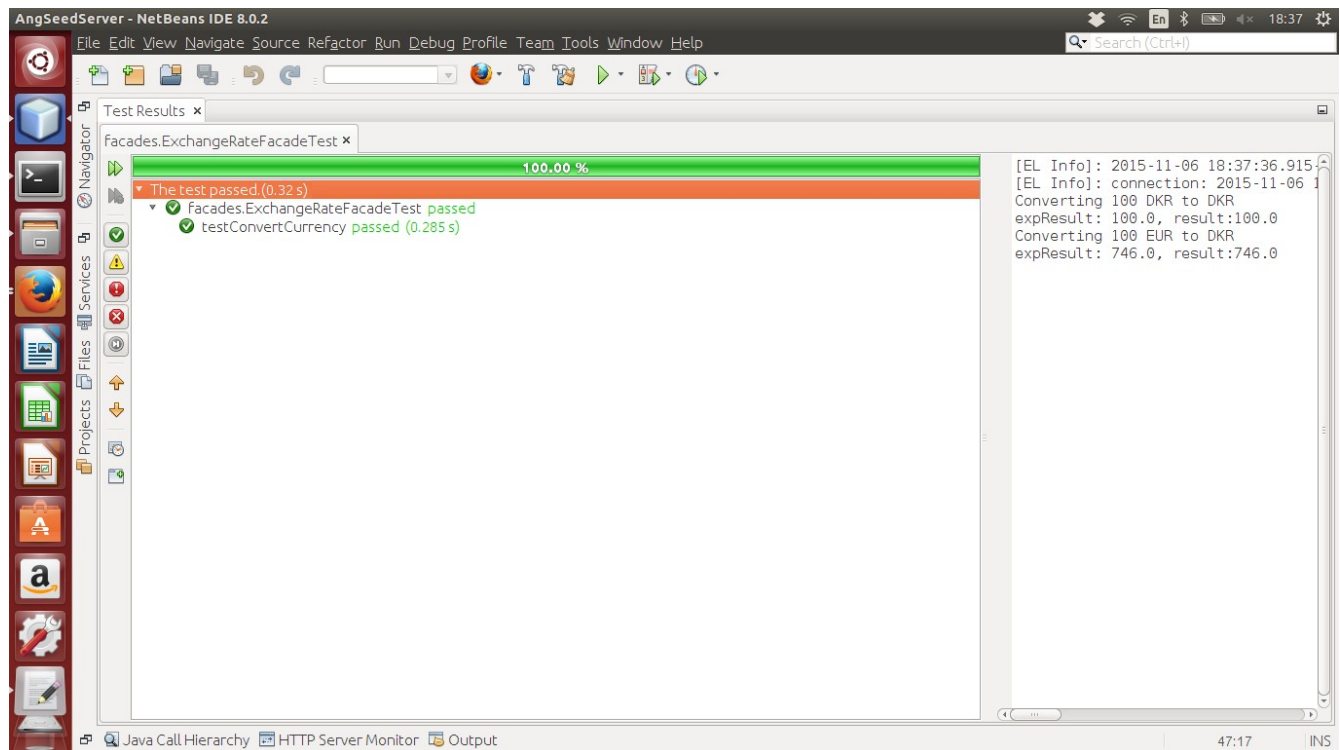
### All users

Here we can see all the users on the system in a list. And delete any of the user, by pressing the blue "delete" button.

### All Users

Username	Password	Roles	
adminLars	test	Admin	<a href="#">delete</a>
userThomas	test	User	<a href="#">delete</a>
user_adminlb	test	User, Admin	<a href="#">delete</a>

## Test of currency converter method in ExchangeRateFacade:



## Conclusion

We, as a dev-team, have done several CA's and projects together now. This leads to the team getting solid and unified, with high work ethics, and a deeper understanding of each others weaknesses and strengths. It shows as we can sit together, in complete silence and code for several hours, only interrupted by the odd question that will be answered quick and precise to the point. If one member has a problem, the two other members quickly pause their work, and as a unit we collect and use all our resources to solve the problem in the fastest and most efficient way.

The team also ensures that the member in question understands the solution, so we don't waste any resources on the same problem another time. We try and rotate the subjects among us everytime we get a new CA or project, so the team as whole will grow.

In this CA Frontend was mainly done by Simon, while Backend mainly by Martin and Ib. Just like the last CA, we feel that all members have collaborated and contributed equally, and therefore all members deserve full points.