

# Structuring Applications

## General part

Explain generally about:

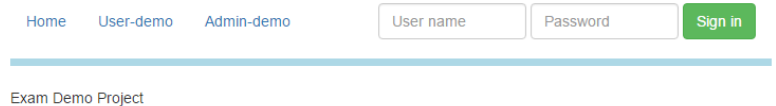
- Explain the purpose(s) of Java Packages and how even large programs can be made "understandable" by means of just package- and file-names.
- Explain the "architecture" of both sides of the seed, which you can clone from here (or better, use your own version with a database): <https://github.com/Lars-m/semesterseedfall2016.git>
- Explain your additions to the seed (in a previous exercise), to handle username and passwords

## Practical part

*The purpose of this exercise is to show that you hook into existing piece code and "twist" it to your needs.*

**Getting started:** Clone your own version of the Seed (the version with DB-support), and open the project in NetBeans.

1) Change the code to provide a UI as sketched in this figure. Remove all code that is no longer used (leave anything in the components +view2+3 folders, and don't touch the backend yet). For the Home Menu Entry, just show a simple message as sketched on the figure



2) Delete existing users from the DB, and add two new users:

- User-1: userName: **Peter**, password: **test**, role: **User**
- User-2: userName: **Anne**, password: **test**, role: **Admin**

Add the necessary changes to the index.html file so that:

- All Users (even when not signed in) can see (only) the Home menu-item.
- Users logged in with the User-role can see (also) the User-demo menu-item (Use the `$scope.isUser` property)
- Users logged in with the Admin-role can see the Admin-demo menu-item (Use the `$scope.isAdmin` property)

3) Change the User-demo page

- Change the backend REST-service to use the following URI: **api/footballclubs** and return the following hardcoded JSON-array<sup>1</sup>:  
[{"name": "Liverpool", "url": "http://www.liverpoolfc.com"}, {"name": "Manchester United", "url": "http://www.manutd.com/"}]
- Change the view to show an unordered list, with links for each club received via the call to: `api/footballclubs`
- Add a backend integration test to verify the behaviour of the REST API implemented above

4) Change the Admin-demo page

- Change the Backend REST-service to use the following path: **api/allusers** and return the following hard-coded JSON-array: [{"name": "Jan", "mail": "jan@a.dk"}, {"name": "Ann", "mail": "ann@a.dk"}, {"name": "ib", "mail": "ib@a.dk"}]
- Change the view to show a table with all the users received via the call to: `api/allusers`
- Add a backend integration test to verify the behaviour of the REST API implemented above

5) Upload your solution to DigitalOcean

---

<sup>1</sup> Hint: create an empty string like " " and paste the content from above into the string, NetBeans will escape all the "'s