

Asterix dokumentaatio

Table of Contents

JSP sivujen rakenne ja toiminta.....	1
Nykyinen rakenne.....	1
Yleiset tiedostot.....	1
header.jsp.....	1
navi.jsp.....	1
tools.jsp.....	1
Uuden sivun lisääminen.....	2
Uuden viewin luominen.....	2
1) Luo uusi JSP sivu.....	2
2) Lisää toiminto controller luokkaan.....	2
3) Muokkaa navigointia.....	3
4) Käännä ja testaa.....	3
Toiminnallisuuden lisääminen viewiin.....	3

JSP sivujen rakenne ja toiminta

Nykyinen rakenne

Hakemisto `src/main/webapp/WEB-INF/views/secure` alihakemistoiin sisältää kaikki suojatut sivut, jaoteltuna roolien mukaan.

Yleiset tiedostot

Seuraavat tiedostot löytyvät kaikkien roolien view kansioista.

header.jsp

Tämä tiedosto sisältää kaikille rooleille yhteiset ulkoasu määritelmät, ja täältä sisällytetään navi.jsp.

navi.jsp

Tämä tiedosto sisältää pelkän navigoinnin HTML koodin. Navigointi on toteutettu unordered list HTML elementeillä (sekä). Navigointi on erotettu header.jsp:stä erilliseen tiedostoon selkeyden vuoksi, ja jotta minimoidaan riski että navigointia muokatessa rikotaan headerin HTML koodi.

tools.jsp

Kyseisen roolin etusivu.

Uuden sivun lisääminen

Uuden sivun luominen olemassa olevalle roolille edellyttää seuraavia toimenpiteitä. Tässä on esimerkkinä käytetty adminille adduser sivun luomista.

Uuden viewin luominen

1) Luo uusi JSP sivu

Luo uusi JSP sivu roolin kansioon. Tiedoston nimen tulee vastata controllerin @RequestMapping value kenttää. Sivun sisältö on esimerkiksi seuraava:

```
<%@ include file="header.jsp" %>
<h1>Add user</h1>

<p>This page is a placeholder for add user functionality.</p>

<%@ include file="../../footer.jsp" %>
```

<%@ include ... %> riveillä lisätään header ja footer osat muodostettavaan HTML sivuun.

2) Lisää toiminto controller luokkaan

Lisää roolin controlleriin uusi handler sivun näyttämistä varten. Muokattava controller sijaitsee hakemistossa src/main/java/fi/asterix/controller/. Controllerin `@RequestMapping` value arvon tulee täsmätä kohdan 1) tiedoston nimen kanssa, ilman `.jsp` päätettä. Alla AdminController.java adduser toiminnon sivun lisäämisen jälkeen, muutokset koodissa lihavoituna.

```
package fi.asterix.controller;

import java.util.List;

import javax.inject.Inject;

import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

import fi.asterix.DAO.AuthorityDAO;
import fi.asterix.JPA.Authority;

@Controller
@RequestMapping(value="/secure/admin")
public class AdminController {

    @Inject
    private AuthorityDAO authDAO;

    @RequestMapping(value = "/tools", method = RequestMethod.GET)
    public String paasivu(Model model) {

        List<Authority> roolit = authDAO.findAll();

        model.addAttribute("roolit", roolit);
        return "secure/admin/tools";
    }
    @RequestMapping(value = "/adduser", method = RequestMethod.GET)
    public String adduser(Model model) {
        return "secure/admin/adduser";
    }
}
```

`RequestMapping` määrittää että mikä URL ajaa tämän metodin. Metodin return lauseessa määrätään että mikä view ladataan metodin ajamisen päätteeksi.

3) Muokkaa navigointia

Lisää linkki uuteen sivuun navi.jsp tiedostoon. Muutokset lihavoitu. Alla navi.jsp adduser sivun lisäämisen jälkeen.

```
<ul>
  <li><a href="tools">Admin frontpage</a></li>
  <li><a href="adduser">Add user</a></li>
  <li><a href="#">List users</a></li>
  <li><a href="#">My profile</a></li>
  <li><a href="../../j_spring_security_logout" > Log out</a></li>
</ul>
```

Linkit joissa on href="#" ovat vaan muistuttamassa siitä että mitä sivuja tarvittaisiin, ja niissä # vaihdetaan korrektiin controllerin requestmapping arvoon. Href kentän sisältö vastaa controllerin rivin @RequestMapping value tietoa, ilman kauttaviivaa.

4) Käännä ja testaa

Aja eclipsessä Run as: maven clean, Run as: maven install, ja Run as: Run on server ja testaa että uusi sivu toimii.

Toiminnallisuuden lisääminen viewiin

Minulla (Jukka) ei ole juuri nyt mitään muistikuvaa miten se kannattaa tehdä. Modelissa se tapahtuisi kai korrektisti, mutta itse lisäisin ehkä jopa controlleriin osan toiminnoista.