

GettingStartedWithNetbeans

Java Server Faces inleiding

Java Server Faces is een relatief nieuwe Java technologie waarmee je op een gelijkaardige manier als bij ASP.NET een webpagina kan ontwerpen op een meer component gebaseerde manier. Java Server Faces maakt het ook eenvoudiger om events (bijvoorbeeld als een gebruiker op een knop klikt op de webpagina) op te vangen op de webserver.

Uiteraard bekijken we ook hoe je de gegevens uit een database kan koppelen aan een Java Server Faces component. Hiervoor kan je bijvoorbeeld zelf een dataaag schrijven of een bibliotheek zoals [hibernate](#) gebruiken die dit automatisch voor je regelt.

De IDE die we gaan gebruiken om Java Server Faces te ontwikkelen is Netbeans 5.0 (<http://www.netbeans.org>). Deze IDE ondersteunt nog geen drag en drop maar maakt toch de ontwikkeling van JSF pagina's heel eenvoudig. Wat je ook nodig hebt is een versie van de Java Development Kit. Dit jaar werken we met de versie 5 van deze development kit en deze kan je afhalen van de server of downloaden van <http://java.sun.com>.

We beginnen deze cursus met het herhalen van de belangrijkste elementen uit de Java syntax en een aantal nieuwe syntax elementen die mogelijk zijn in de JDK 1.5. Verder maken we je in dit labo vertrouwd met de Netbeans mogelijkheden in verband met website ontwikkeling.

Een nieuwe webtoepassing creëren

Netbeans bevat een ingebouwde webserver dus die moeten we niet apart gaan installeren (de gebruikte webserver is Tomcat). Het is wel noodzakelijk om de bibliotheek bestanden van Java Server Faces te installeren.

Eerst moeten we een nieuw webproject aanmaken, dit kan eenvoudigweg met behulp van File New Project ...

Volg dan de wizard om een nieuw web project aan te maken :

1. **Choose Project** : Kies de categorie "Web" en kies als project een gewone "Web Application". Klik op "**Next**".

2. **Name and Location** : Kies een map voor je project, en zorg er ook voor dat het "**Source Level**" juist ingesteld staat. Omdat we meestal wel de JDK 1.5 willen gebruiken kan je deze optie afvinken.

3. **Frameworks** : Kies als framework het JSF framework. "**De Servlet URL mapping**" stelt in dat alle JSF pagina's via een speciale URL worden aangesproken (bijvoorbeeld <http://localhost:8084/faces/eerstejsfpagina.jsp>).

Het project is nu aangemaakt en onder Configuration Files is een extra bestand faces-config.xml gecreëerd. Ook is er een voorbeeld JSF pagina genaamd welcomeJSF.jsp.

MyFaces bibliotheek aanmaken

Om alle componenten van de [MyFaces](#) bibliotheek te kunnen aanspreken moet je een aantal bibliotheken downloaden van het internet. [MyFaces](#) gebruikt nogal wat bibliotheken uit het Apache Commons project, en je kan van deze bibliotheken de laatste versie downloaden. De set van bibliotheken waarmee je alle [MyFaces](#) componenten zonder problemen kan gebruiken is de volgende :

commons-lang-2.1.jar
commons-beanutils.jar
commons-codec-1.3.jar
commons-collections-3.1.jar
commons-digester-1.7.jar
commons-el.jar
commons-fileupload-1.1.jar
commons-validator-1.2.0.jar
commons-io-1.1.jar
jakarta-oro-2.0.8.jar
myfaces-all.jar

Alle "**commons**" bibliotheken kunnen gedownload worden van de [commons download website](#). De Jakarta ORO bibliotheek kan je downloaden van de [ORO website](#).

Tenslotte kan je de myfaces-all bibliotheek downloaden van [de MyFaces website](#) zelf.

Via Tools -> Library Manager kan je in Netbeans een nieuwe bibliotheek aanmaken. Klik op "New Library" en geef als naam bijvoorbeeld [MyFaces](#). Voeg de elf jar bestanden toe die je gedownload hebt met behulp van de knop "Add JAR/Folder...".

Verdere Project configuratie

1. Open het context menu van het project en kies "Properties". Selecteer het "Libraries" item en hier kan je de bibliotheek "MyFaces" toevoegen die je net gecreëerd hebt. Omdat de [MyFaces](#) bibliotheek een volledige implementatie bevat moet je hier ook de JSF 1.1 bibliotheek afvinken. In principe zouden JSF 1.1 en tomahawk.jar moeten kunnen samenwerken (volgens de [MyFaces](#) documentatie) maar in de praktijk blijkt dit niet te werken.
2. Open het **web.xml** bestand (onder Configuration Files) en voeg er de volgende configuratie instellingen aan toe (Druk op de knop "XML" om de XML weergave van dit bestand te bekijken) :

```
<filter>
  <filter-name>MyFacesExtensionsFilter</filter-name>
  <filter-class>org.apache.myfaces.component.html.util.ExtensionsFilter</filter-class>
  <init-param>
    <param-name>maxFileSize</param-name>
    <param-value>20m</param-value>
  </init-param>
</filter>
<filter-mapping>
  <filter-name>MyFacesExtensionsFilter</filter-name>
  <url-pattern>/faces/*</url-pattern>
</filter-mapping>
```

Dit stukje xml kan je rechtstreeks onder het **web-app** element plaatsen.

Een eerste JSF pagina met een [MyFaces](#) component

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@taglib prefix="f" uri="http://java.sun.com/jsf/core"%>
<%@taglib prefix="h" uri="http://java.sun.com/jsf/html"%>
<%@taglib uri="http://myfaces.apache.org/tomahawk" prefix="t"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <f:view>
      <h:form>
        <t:outputText value="Hello MyFaces!" />
      </h:form>
    </f:view>
  </body>
</html>
```