

Facelets

**JavaServer™ Faces View
Definition Framework**

by Ivana Vitlić



Uvod

Facelets

- open source web framework koji predstavlja alternativnu view handler tehnologiju za JavaServer Faces (JSF)
- template-ing jezik od početka razvijan da odgovara JSF

Glavne karakteristike

- templating
- višestruko korišćenje koda
- lak razvoj koda
- smanjena količina UI koda



Zašto koristiti Facelets?

Zašto koristiti Facelets umesto JSP za kreiranje JSF strane

- Facelets mogu da rade sa svakom implementacijom i verzijom JSF-a
- obezbeđuju rešenje za nekompatibilnost JSP-a i JSF-a
- obezbeđuju brže kompajliranje (u odnosu na JSP) jer se "iza scene" ne generiše i kompajlira Java bajtkod kad se stranica posećuje prvi put
- dozvoljava kreiranje komponenti čiji je razvoj trivijalan u poređenju sa "čistim" JSF komponentama. Na primer, nema potrebe kreirati tagove za UI komponente
- podržava Unified Expression Language (EL), uključujući podršku za EL funkcije i EL validaciju za vreme kompajliranja.



Zašto koristiti Facelets? (2)

- obezbeđuje precizan izveštaj kada dodje do greške (pokazuje liniju u kojoj se dogodio izuzetak)
- moguće je koristiti jsfc atribut da bi se omogućila integracija sa postojećim HTML editorima
- ne zahteva poseban render kit



Kreiranje aplikacije koristeći Facelets

- Facelets, kao i JSF, je baziran na standardima i ne zavisi od operativnog sistema ili proizvoda
- <http://facelets.dev.java.net/>

Koraci za kreiranje aplikacije "from scratch"

1. kreiranje strukture projekta
2. konfigurisanje web deskriptora (web.xml)
3. konfigurisanje faces deskriptora (faces-config.xml)
4. kreiranje web stranica



Kreiranje strukture projekta

- potrebno je obezbediti korektnu listu biblioteka koje se koriste
- biblioteke koje nedostaju, ili nekompatibilnost izmedju različitih verzija biblioteka mogu izazvati najčudnije i najneobičnije izuzetke
- dodatno otezava različitost web kontejnera, koji, svaki za sebe, imaju svoje biblioteke i posebnosti



Primer setupa projekta koji koristi Facelets i MyFaces

```
$PROJECT
+- /WEB-INF
  +- /lib
    +- /commons-beanutils.jar
    +- /commons-collections.jar
    +- /commons-digester.jar
    +- /commons-logging.jar
    +- /el-api
    +- /el-impl
    +- /jsf-facelets.jar
    +- /myfaces-api.jar
    +- /myfaces-impl.jar
  +- /web.xml
  +- /faces-config.xml
+- /[xhtml documents]
```

EL biblioteke (`el-api` i `el-impl`) treba izbaciti ukoliko je kontejner JSP 2.1 kompatibilan. JSP 2.1 sadrzi EL implementaciju i nastao bi konflikt pri pokretanju aplikacije.

Ako se koristi Reference Implementation (RI), `myfaces-api.jar` i `myfaces-impl.jar` se zamenjuju sa `jsf-ri.jar` i `jsf-api.jar`.



Konfigurisanje web deskriptora

- neophodno je u `web.xml` uključiti kontekst parametar `javax.faces.DEFAULT_SUFFIX` koji definiše sufiks dokumenta za pogled.
- po konvenciji, default ekstenzija za stranice urađene pomoću Facelets-a je `*.xhtml`
- važno je mapirati `FacesServlet`



Konfigurisanje web deskriptora (2)

Modifikacija web.xml fajla:

```
<web-app>
.....
<!-- Use Documents Saved as *.xhtml -->
  <context-param>
    <param-name>
      javax.faces.DEFAULT_SUFFIX
    </param-name>
    <param-value>.xhtml</param-value>
  </context-param>

  <!-- Faces Servlet -->
  <servlet>
    <servlet-name>Faces Servlet</servlet-name>
    <servlet-class>javax.faces.webapp.FacesServlet
    </servlet-class>
    <load-on-startup>1</load-on-startup>
  </servlet>
```



Modifikacija web.xml fajla (2)

```
<!-- Faces Servlet Mapping -->  
  <servlet-mapping>  
    <servlet-name>Faces Servlet</servlet-name>  
    <url-pattern>*.jsf</url-pattern>  
  </servlet-mapping>  
  
</web-app>
```

Mogu se dodati opcioni JSF-RI parametri koji pomažu pri debug-ovanju

```
<!-- Optional JSF-RI Parameters to Help Debug -->  
  <context-param>  
    <param-name>com.sun.faces.validateXml</param-name>  
    <param-value>true</param-value>  
  </context-param>  
  <context-param>  
    <param-name>com.sun.faces.verifyObjects</param-name>  
    <param-value>true</param-value>  
  </context-param>
```



Parametri za inicijalizaciju

- `facelets.BUFFER_SIZE`

Definiše veličinu bafera kada se generiše `ResponseWriter`. Predefinisana vrednost je `-1`, što znači da odgovoru neće biti dodeljena veličina bafera. Pri razvoju koda, ova vrednost može da se uveća da bi se sprečilo da se odgovor delimično renderuje kada dodje do greške.

- `facelets.DECORATORS`

Lista klasa odvojenih tačka-zarezom(`;`), koje implementiraju `com.sun.facelets.tag.TagDecorator` i imaju konstruktor bez argumenata. Ovi će se dekoratori učitati kada prvi zahtev pogodi `FaceletViewHandler` za kompilaciju stranice.

- `facelets.DEVELOPMENT`

Ako je ovaj atribut `true`, `FaceletViewHandler` ispisuje debug informacije kada dođe do greške prilikom renderovanja. Podrazumevana vrednost je `false`.



Parametri za inicijalizaciju(2)

- `facelets.LIBRARIES`

Lista putanja do Facelets tag biblioteka razdvojenih tačka-zarezom (;). Putanje su relativne u odnosu na root aplikacije. Ove biblioteke će se učitati kada prvi zahtev pogodi `FaceletViewHandler` za kompilaciju stranice.

- `facelets.REFRESH_PERIOD`

Period(u sekundama) posle kog kompajler proverava da li je došlo do promena nakon što je zatražena stranica. Korisno prilikom razvoja koda, jer je moguće menjati stranicu dok aplikacija radi, a stranica će biti iskompajlirana. Ukoliko se ne želi proveriti da li je došlo do promena parametar se postavlja na -1. Predefinisana vrednost je 2.

- `facelets.RESOURCE_RESOLVER`

Obezbeđuje drugi `ResourceResolver`, čime se dozvoljava implementiranje drugih načina za razlikovanje resursa resursi. Predefinisana vrednost je `com.sun.facelets.impl.DefaultResourceResolver`



Parametri za inicijalizaciju(3)

- `facelets.SKIP_COMMENTS`

Kada je true, kompajler preskače komentare na stranici, i ništa unutar komentara se neće renderovati u pogledu. Kada je false, iako se tagovi neće kompajlirati, EL izrazi će se parsirati kao da su inline. Podrazumevana vrednost je true.

- `facelets.VIEW_MAPPINGS`

*Lista resursa koja bi Facelets trebalo da koristi razdvojenih tačka-zarezom (;). Ukoliko putanje resursa nisu navedene, Facelets će opsluživati sve zahteve (DEFAULT). Ako je specificirana jedna ili više putanja, Facelets će koristiti samo one navedene, a u protivnom fall-back-ovati na roditeljski ili predefinisani ViewHandler (JSP). Ovo zahteva da se FacesServlet u `web.xml` mapira sa prefiksom za hvatanje više tipova fajlova (fajlova sa različitim ekstenzijama): `/faces/`**



Konfigurisanje Faces deskriptora

- Facelets zamenjuje predefinisani JavaServer Faces ViewHandler sa svojom implementacijom predstavljenom klasom `com.sun.facelets.FaceletViewHandler`
- Potrebno je specificirati `<view-handler>` element u `faces-config.xml` fajlu:

```
<faces-config>
  <application>
    <view-handler>
      com.sun.facelets.FaceletViewHandler
    </view-handler>
  </application>
</faces-config>
```



Konfigurisanje Faces deskriptora (2)

- Moguće je mapirati JavaBeans u JavaServer Faces aplikaciju pomoću `faces-config.xml`
- JavaBeans koja se koriste mogu biti povezana preko bean property-a i može im se dodeliti opseg: jedan zahtev, cela sesija, za sve u aplikaciji ili ni za koga (ukoliko ne želite da se smeštaju direktno)
- navigacija među stranicama se određuje tagovima u `faces-config.xml` fajlu



Kreiranje web stranica

Posmatrajmo ceo problem na primeru:

- Kreiramo leksikon ptica - za svaku pticu kreiramo stranu (npr. za papagaja i orla)
- sve strane imaju sličan layout (sadržaj i navigaciju)- moguće je kreirati template.

Sa ovim pretpostavkama struktura projekta izgleda:

```
$PROJECT
+- /WEB-INF
    +- /lib [with dependencies]
    +- /web.xml
    +- /faces-config.xml
+- template.xhtml
+- index.xhtml
+- parrot.xhtml
+- eagle.xhtml
+- menu.xhtml
```

Kreiramo 5 dokumenata (sa ekstenzijom *.xhtml) - template (template.xhtml) koga koriste ostale 3 strane (index.xhtml, parrot.xhtml i eagle.xhtml), i meni (menu.xhtml), koju koristimo za navigaciju



Kreiranje web stranica (2)

1. Template

- definišemo osnovni layout stranice sa Facelets UI tagom `ui:insert`, koji se koristi da se definišu delovi strane koji mogu biti zamenjeni.
- za Facelets tagove koristi se `http://java.sun.com/jsf/facelets` namespace.
- `ui` je prefiks za tagove iz UI Facelets biblioteke
- za sve tagove koji se koriste u dokumentu mora se deklarirati namespace. Ako je namespace invalidan, prijavljuje se greška pri kompilaciji
- Primer koda za `template.xhtml` stranicu:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:h="http://java.sun.com/jsf/html">
<head>
  <title>The Happy Birds Directory</title>
  <style type="text/css">
```



Primer koda za template.xhtml

```
<!--
    .box {
        float: right;
        width: 50%;
        border: black dotted 1px;
        padding: 5px
    }
-->
</style>
</head>
<body>
    <h:form>
        <h1>The Happy Birds Directory</h1>
        <div class="box">
            <ui:insert name="navigation"/>
        </div>
        <ui:insert name="main">
            Welcome to the nest!
        </ui:insert>
    </h:form>
</body>
</html>
```



Template stranica (2)

- plutajući `div` element se koristi za sadržaj, a glavna sekcija za opis ptica
- najvažniji elementi su dva `ui:insert` tag-a (nazvani `navigation` i `main`)
- deklariramo `navigation` oblast unutar `div` elementa i otvorenu `main` sekciju.

Možemo da damo predefinisane vrednosti ovim oblastima tako što ćemo taj sadržaj uneti između `ui:insert` tag-ova. Ako ne budemo želeli da prepíšemo glavnu sekciju ispisaće se fraza “Welcome to the nest!”



Kreiranje web stranica (3)

2. Naslovna strana (*index.xhtml*)

- primer koda za *index.xhtml* stranicu:

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
<body>
This and everything before will be ignored
  <ui:composition template="template.xhtml">
    <ui:define name="navigation">
      <ui:include src="menu.xhtml"/>
    </ui:define>
  </ui:composition>
This and everything after will be ignored
</body>
</html>
```



Naslovna strana `index.xhtml`

- koristimo 3 nova Facelets tag-a:

- `ui:composition`

Pomoću njegovog `template` atributa referenciramo `template` koji želimo da koristimo na stranici (`template.xhtml` dokument). Sve što se nalazi unutar tag-a izračunava Facelets pri kompajliranju. Svi elementi van ovog taga se ignorišu

- `ui:define`

tag se koristi da se definiše sadržaj koji ide u oblasti deklarisanе u `template-u`. Njegov `name` atribut mora da se slaže sa onim u `ui:insert` tag-u u `template-u`. Za `index.xhtml` pogled definišemo samo `navigation` oblast, a koristićemo predefinisanu `main` oblast.

- `ui:include`

dozvoljava da uključimo sadržaj drugog dokumenta - u našem slučaju je to fajl `menu.xhtml`



Kreiranje web stranica (4)

3. Meni za navigaciju (menu.xhtml)

- primer koda za menu.xhtml stranicu

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html    xmlns="http://www.w3.org/1999/xhtml"
        xmlns:ui="http://java.sun.com/jsf/facelets"
        xmlns:h="http://java.sun.com/jsf/html">
    <body>
        This and everything before will be ignored
    <ui:composition>
        <h3>Contents table</h3>
        <hr/>
        <h:panelGrid columns="1">
            <h:commandLink value="Home" action="home" />
            <h:commandLink value="Parrot" action="parrot" />
            <h:commandLink value="Eagle" action="eagle" />
        </h:panelGrid>
    </ui:composition>
    This and everything after will be ignored
</body>
</html>
```



Strana za navigaciju (menu.xhtml)

- koristi se `ui:composition` tag, pa će se sve van njega ukloniti kada se meni uključi u stranu.
- koristi se nekoliko link elemenata (`h:commandLink`) sa `action` atributima definisanim da trigeruju navigaciju u aplikaciji
- Pravila za navigaciju se dodaju u `faces-config.xml` fajl



Primer koda za faces-config.xml

- Faces-config.xml fajl sa dodatim pravilima za navigaciju:

```
<!DOCTYPE faces-config PUBLIC
"-//Sun Microsystems, Inc.//DTD JavaServer Faces Config 1.0//EN"
"http://java.sun.com/dtd/web-facesconfig_1_1.dtd" >

<faces-config>
  <application>
    <view-handler>
      com.sun.facelets.FaceletViewHandler
    </view-handler>
  </application>
  <navigation-rule>
    <navigation-case>
      <from-outcome>home</from-outcome>
      <to-view-id>/index.xhtml</to-view-id>
    </navigation-case>
  </navigation-rule>
  <navigation-rule>
    <navigation-case>
      <from-outcome>parrot</from-outcome>
      <to-view-id>/parrot.xhtml</to-view-id>
    </navigation-case>
  </navigation-rule>
</faces-config>
```



Primer koda za faces-config.xml (2)

```
<navigation-rule>
  <navigation-case>
    <from-outcome>eagle</from-outcome>
    <to-view-id>/eagle.xhtml</to-view-id>
  </navigation-case>
</navigation-rule>
</faces-config>
```

- ovime smo postigli da svaki od linkova vodi korisnika na neku od strana



Kreiranje web stranica (5)

4. Stranice sa konkretnim sadržajem (*parrot.xhtml*, *eagle.xhtml*)

- skoro identične sa `index.xhtml` - koriste template i specificiraju sadržaj koji će se pokazati u main i navigation sekcijama
- primer `parrot.xhtml` strane:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
  <body>
    This and everything before will be ignored
    <ui:composition template="template.xhtml">
      <ui:define name="navigation">
        <ui:include src="menu.xhtml"/>
      </ui:define>
```



Primer koda za parrots.xhtml (2)

```
<ui:define name="main">
    <h1>Parrot</h1>
    <p>
        Parrots are interesting birds...
    </p>
</ui:define>
</ui:composition>
    This and everything after will be ignored
</body>
</html>
```

- opis papagaja stavljamo u main sekciju koristeći ui:define tag, vodeći računa da se njegovo ime poklapa sa onim u ui:insert tag-u template-a.
- navigation sekcija sadrži samo meni kao na index strani



Zaštita pristupa

- ograničavanje pristupa korisnicima je zajednička stvar za sve aplikacione servere i spada u specifikaciju Servleta
- ukoliko imamo neki Facelet XHTML dokument koji želimo da posmatramo u browser-u tokom razvoja koda, ali ne želimo da mu pristupi bilo ko drugi ukoliko nije serviran kroz JavaServer Faces Servlet, dodajemo sledeći blok u web.xml fajl:

```
<web-app>
.....
<security-constraint>
  <display-name>Restrict XHTML Documents</display-name>
  <web-resource-collection>
    <web-resource-name>XHTML</web-resource-name>
    <url-pattern>*.xhtml</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <description>
      Only Let 'developer's access XHTML pages
    </description>
  </auth-constraint>
</security-constraint>
</web-app>
```



Zaštita pristupa (2)

- Više o zaštiti prava pristupa može se naći na <http://java.sun.com/webservices/docs/1.3/tutorial/doc/Security2.html>



Korišćenje Tag-ova

1. Tag namespace-ovi

- Facelets koristi validan XML sa namespace podrškom za kompilaciju.
- Da bi se koristila biblioteka tagova potrebno ju je uključiti tako što će se deklarirati namespace
- Deklarisanje namespace-a uključuje korišćenje URI-a ili URL-a biblioteke i njegovo mapiranje u prefiks

Posmatrajmo primer:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html">
<body>
    <h:inputText value="#{hello.world}"/>
</body>
</html>
```



Tag namespace-ovi (2)

U gornjem primeru deklarirali smo dva namespace-a

- `xmlns="http://www.w3.org/1999/xhtml"`,
koji označava da se svaki tag bez prefiksa (npr. `<body>`) tumači kao deo XHTML
- `xmlns:h="http://java.sun.com/jsf/html"`,
koji mapira prefiks `h:` u JSF HTML biblioteku



Korišćenje Tag-ova (2)

2.Korišćenje 'jsfc'

- Facelets omogućava da se jedan XML element na strani konvertuje u drugi u vreme kompajliranja tako što se specificira jsfc atribut.

Posmatrajmo primer:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html">
<body>
    <input type="text" jsfc="h:inputText" value="#{hello.world}"/>
</body>
</html>
```



Korišćenje 'jsfc' (2)

- U gornjem primeru `<input>` tag se prikazuje u browser-ima i visual alatima kao klasično tekst polje (text field). Bilo bi zgodno koristiti to isto tekst polje u JavaServer Faces, pa se obezbeđuje deklaracija `jsfc="h:inputText"`. Ova deklaracija govori kompajleru da resolve-uje `<input>` tag kao `<h:inputText>`
- Napomenimo da ista pravila za namespace-ove važe i za vrednost `jsfc` atributa



Korišćenje Tag-ova (3)

3. Nerazrešeni namespace-ovi

- Ukoliko u dokumentu postoji tag pod namespace-om koji nije naveden u Tag biblioteci, Facelets će ga tretirati kao prost teks fragmet i optimizovaće ga za kreiranje stabla
- Ako je namespace poznat, a naziv elementa nije u biblioteci, baciće se izuzetak koji će obavestiti u kom je fajlu i liniji unutar njega došlo do problema
- Ako se tag ili komponenta ne pojavljuju na stranici posle renderovanja, potrebno je proveriti source strane, da bi se videlo da li je tag izignorisan. Ukoliko je to slučaj, to uglavnom znači da se za konkretnu biblioteku koristi pogrešan namespace



Korišćenje Tag-ova (4)

4. Ponuđene biblioteke

- Facelets dolazi sa podrškom za sve tagove (komponente) koji su uključeni u JavaServer Faces specifikaciju. Velika pažnja je posvećena imitiranju JSP tag paralela da bi se iskoristilo postojeće znanje i dokumentacija



Biblioteke tag-ova

<i>Namespace</i>	<i>Prefiks</i>	<i>Biblioteka</i>
http://www.w3.org/1999/xhtml	--	HTML
http://java.sun.com/jsf/facelets	ui	Faces
http://java.sun.com/jsf/core	f	JSF Core
http://java.sun.com/jsf/html	h	JSF HTML
http://myfaces.apache.org/tomahawk	t	MyFaces Tomahawk
http://java.sun.com/jstl/core	c	JSTL Core
http://java.sun.com/jsp/jstl/fmt	fn	JSTL Functions



Biblioteke tag-ova (2)

Pojedine biblioteke su već uključene u Facelets

- *Templating biblioteke*
sadrže tagove koji se koriste za templating
- *JSF biblioteke*
sadrže neophodne informacije da bi se omogućilo korišćenje JSF tagova u dokumentima na isti način kao u JSP-u
- *JSTL*
delimična podrška za JSP Standard Tag Library (JSTL) tag-ove (potpuna za Functions i delimična za Core). Core tag-ovi koji su podržani su: `c:if`, `c:forEach`, `c:catch` i `c:set`



Korišćenje Tag-ova (5)

5. Učitavanje biblioteka

Postoji više načina:

- Facelets će učitati sve biblioteke iz classpath-a i one specificirane u `web.xml`
- Dok Facelets pretražuje classpath, pokušaće da učitava svaki fajl sa ekstenzijom `*.taglib.xml` iz `/meta-inf` direktorijuma unutar Jar-ova
- Facelets će proveriti inicijalizacioni parametar u `web.xml` i učitati sve biblioteke specificirane tamo (relativno u odnosu na root aplikacije). Ovo je idealan način da se deklariraju specijalizovani tagovi za konkretnu aplikaciju, koji ne moraju da budu distribuirani unutar Jar-ova



Korišćenje Tag-ova (6)

Lista Facelets tag-ova

1. <ui:component/>

- uključuje novu instancu UIComponent u JavaServer Faces stablo kao koren svih komponenata koje sadrži

- atributi:

- id (nije obavezan): ukoliko ne postoji, Facelets sam kreira id prateći pravila iz JavaServer specifikacije

- binding (nije obavezan): koristi se za referenciranje instance UIComponent

- sve van ovog taga kompajler ignoriše - neće ga biti posle renderovanja



Lista Facelets tag-ova (2)

2. `<ui:composition/>`

- enkapsulira sadržaj koji može da se uključi u druge Facelet strane
- atributi:
 - `template` (nije obavezan): putanja do template-a koji će biti populisan sadržajem koji se nalazi unutar ovog tag-a
- UIComponent stablo može da se formira od delova opisanih na različitim stranama unutar aplikacije
- sve van ovog tag-a će biti uklonjeno
- ne kreira komponentu u stablu
- može da se koristi za popunjavanje template-a



Lista Facelets tag-ova (3)

3. `<ui:debug/>`

- pokreće se tasterima Ctrl + Shift + `<hot key>` (D, predefinisano), i aktivira pop-up prozor koji pokazuje stablo i opseg promenljivih
- atributi:
 - `hotkey` (nije obavezan): pritiskom na Ctrl + Shift + `<hot key>` pokazaće se Facelets debug prozor
 - `rendered` (nije obavezan): mora da ima Boolean vrednost. Ako je false, skript potreban za pokretanje debug prozora neće biti prisutan na stranici
- po konvenciji nalazi se na kraju strane, ali može biti bilo gde

4. `<ui:decorate/>`

- sličan sa `ui:composition` tag-om, s tim što `decorate` ne uklanja sve van taga
- za dodavanje sadržaja oko 'dekorisane' sekcije koristeći template
- atributi:
 - `template` (obavezan): putanja do template-a koji će biti populisan sadržajem između `composition` tag-a



Lista Facelets tag-ova (4)

5. `<ui:define/>`

- koristi se za umetanje imenovanog sadržaja u template
- koristi se unutar tagova koji dozvoljavaju templating, kao što su `ui:composition` i `ui:decorate` tag-ovi.
- nazivi koji se koriste u `define` tag-u moraju da se podudaraju sa imenima korišćenim u `ui:insert` tag-u ciljnog template-a.
- atributi:
 - name (obavezan): specificira ime koje mora da se poklapa sa nazivom `ui:insert` tag-u ciljnog template-a
- Facelets kompajler će ignorisati sadržaj van ovog tag-a



Lista Facelets tag-ova (5)

6. <ui:fragment/>

- sličan `ui:component` tag-u, s tim što se sadržaj ovog taga ne odseca
- atributi:

`id` (nije obavezan): moguće je obezbediti `id`, a ukoliko ga nema Facelets će koristiti `id` prateći pravila JavaServer specifikacije

`binding` (nije obavezan): za referenciranje instance `UIComponent`

ubacuje novu instancu `UIComponent` u stablo komponenti. Sve druge komponente i sadržaj van tag-a će i dalje postojati za vreme kompajliranja



Lista Facelets tag-ova (6)

7. `<ui:include/>`

- služi da uključi u tekući dokument bilo koji drugi Facelets fajl koji sadrži `ui:component` ili `ui:composition` tag ili samo deo XHTML ili XML

- atributi:

- `src` (obavezan): vrednost ili EL izraz koji deklariše fajl koji treba uključiti. Putanja može biti apsolutna ili relativna (u odnosu na originalni Facelets koji je zatražen)

8. `<ui:insert/>`

- za specificiranje onih delova template-a koji mogu biti zamenjeni sa `ui:define` tag-ovima deklarisanim u klijentskom template-u

- atributi:

- `name` (nije obavezan): koristi se za poklapanje imena insert tag-a sa istim imenom u klijentu. Ukoliko ne postoji, ceo klijentski template će biti ubačen



Lista Facelets tag-ova (7)

može da ima ugnježdjeni sadržaj. Ako je to slučaj, a nije specificiran define tag u klijentu čije se ime poklapa sa imenom insert tag-a, biće umetnut ugnježdjeni sadržaj (koristi se za umetanje predefinisanog sadržaja, kada nije specificiran define tag)

9. <ui:param/>

- za prosledjivanje objekata(kao varijabli) izmedju Facelets-a

- atributi:

 - name (obavezan): ime varijable koja će se proslediti

 - value (obavezan): vrednost ili EL izraz koja će biti dodeljena imenovanoj promenljivoj

- name atribut ui:param tag-a mora da se poklapa sa imenom ui:define tag-a sadržanom u template-u definisanom u ui:composition ili ui:decorate tag-u



Lista Facelets tag-ova (8)

10. `<ui:remove/>`

- za uklanjanje delova koda u vreme kompajliranja
- nema attribute
- način da se, kada aplikacija ode na produkcijski server, uklone oni delovi koda koji imaju smisla samo pri razvoju ili testiranju, a da kod ipak ostane za naredni razvoj ili ispravljanje bug-ova

11. `<ui:repeat/>`

- za iteraciju kroz listu objekata
- atributi:
 - value (obavezan): EL izraz koji se izračunava u listu objekata nad kojom će se iterirati
 - var (obavezan): ime varijable koja će se koristiti za iteraciju



Migriranje postojeće aplikacije sa JSP-a

- postupak je prilično jednostavan

1. uključiti Facelets JAR i njegove zavisnosti

2. promeniti deklaraciju strane svih stranica sa

```
<%@ taglib uri=http://java.sun.com/jsf/html prefix="h"%>  
<%@ taglib uri=http://java.sun.com/jsf/core prefix="f"%>
```

na

```
<html xmlns="http://www.w3.org/1999/xhtml"  
xmlns:h="http://java.sun.com/jsf/html"  
xmlns:f="http://java.sun.com/jsf/core">
```

3. obezbediti da su svi dokumenti po XHTML standardu
(da su svi tag-ovi propisno zatvoreni)



Literatura

- "Facelets Essentials Guide to JavaServer™ Faces View Definition Framework", Bruno Aranda and Zubin Wadia
- <https://facelets.dev.java.net/>
- Google

