JSF Application that uses managed-bean and ejb

Example jsf-managedBean-and-ejb can be browsed at https://github.com/apache/tomee/tree/master/examples/jsf-managedBean-and-ejb

This is a simple web-app showing how to use dependency injection in JSF managed beans using TomEE.

It contains a Local Stateless session bean `CalculatorImpl` which adds two numbers and returns the result. The application also contains a JSF managed bean `CalculatorBean`, which uses the EJB to add two numbers and display the results to the user. The EJB is injected in the managed bean using `@EJB` annotation.

A little note on the setup:

You could run this in the latest Apache TomEE [snapshot](https://repository.apache.org/content/repositories/snapshots/org/apache/openejb/apache-tomee/)

As for the libraries, myfaces-api and myfaces-impl are provided in tomee/lib and hence they should not be a part of the war. In maven terms, they would be with scope 'provided'

Also note that we use servlet 2.5 declaration in web.xml

```
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
version="2.5">
```

And we use 2.0 version of faces-config

```
<faces-config xmlns="http://java.sun.com/xml/ns/javaee"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
      http://java.sun.com/xml/ns/javaee/web-facesconfig_2_0.xsd"
              version="2.0">
```

The complete source code is provided below but let's break down to look at some smaller snippets and see how it works.

We'll first declare the `FacesServlet` in the `web.xml`

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

`FacesServlet` acts as the master controller.

We'll then create the `calculator.xhtml` file.

```
<h:outputText value='Enter first number'/>
<h:inputText value='#{calculatorBean.x}'/>
<h:outputText value='Enter second number'/>
<h:inputText value='#{calculatorBean.y}'/>
<h:commandButton action="#{calculatorBean.add}" value="Add"/>
```

Notice how we've use the bean here. By default it is the simple class name of the managed bean.

When a request comes in, the bean is instantiated and placed in the appropriate scope. By default, the bean is placed in the request scope.

```
<h:inputText value='#{calculatorBean.x}'/>
```

Here, getX() method of calculatorBean is invoked and the resulting value is displayed. x being a Double, we rightly should see 0.0 displayed.

When you change the value and submit the form, these entered values are bound using the setters in the bean and then the commandButton-action method is invoked.

In this case, `CalculatorBean#add()` is invoked.

`Calculator#add()` delegates the work to the ejb, gets the result, stores it and then instructs what view is to be rendered.

You're right. The return value "success" is checked up in faces-config navigation-rules and the respective page is rendered.

In our case, `result.xhtml` page is rendered.

The request scoped `calculatorBean` is available here, and we use EL to display the values.

Source

Calculator

```
package org.superbiz.jsf;

import javax.ejb.Local;

@Local
public interface Calculator {
    public double add(double x, double y);
}
```

CalculatorBean

```
package org.superbiz.jsf;

import javax.ejb.EJB;
import javax.faces.bean.ManagedBean;

@ManagedBean
public class CalculatorBean {
    @EJB
    Calculator calculator;
    private double x;
    private double y;
    private double result;

    public double getX() {
        return x;
    }

    public void setX(double x) {
        this.x = x;
    }

    public double getY() {
        return y;
    }

    public void setY(double y) {
        this.y = y;
    }

    public double getResult() {
        return result;
    }

    public void setResult(double result) {
        this.result = result;
    }

    public String add() {
        result = calculator.add(x, y);
        return "success";
    }
}
```

CalculatorImpl

```
package org.superbiz.jsf;

import javax.ejb.Stateless;

@Stateless
public class CalculatorImpl implements Calculator {

    public double add(double x, double y) {
        return x + y;
    }
}
```

web.xml

```
<?xml version="1.0"?>
```

```
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
version="2.5">
```

```
<description>MyProject web.xml</description>
```

```
<!-- 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>
```

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

```
<!-- Welcome files -->
<welcome-file-list>
   <welcome-file>index.jsp</welcome-file>
   <welcome-file>index.html</welcome-file>
</welcome-file-list>
</web-app>
```

Calculator.xhtml

```
<?xml version="1.0" encoding="UTF-8"?>
<!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:f="http://java.sun.com/jsf/core"
xmlns:h="http://java.sun.com/jsf/html">
```

```
<h:body bgcolor="white">
    <f:view>
        <h:form>
            <h:panelGrid columns="2">
            <h:outputText value='Enter first number'/>
           <h:inputText value='#{calculatorBean.x}'/>
            <h:outputText value='Enter second number'/>
            <h:inputText value='#{calculatorBean.y}'/>
           <h:commandButton action="#{calculatorBean.add}" value="Add"/>
            </h:panelGrid>
        </h:form>
   </f:view>
</h:body>
</html>
```

Result.xhtml

```
<?xml version="1.0" encoding="UTF-8"?>
<!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:f="http://java.sun.com/jsf/core"
xmlns:h="http://java.sun.com/jsf/html">
```

```
<h:body>
    <f:view>
        <h:form id="mainForm">
            <h2><h:outputText value="Result of adding #{calculatorBean.x} and
#{calculatorBean.y} is #{calculatorBean.result }"/></h2>
            <h:commandLink action="back">
            <h:outputText value="Home"/>
            </h:commandLink>
        </h:form>
    </f:view>
</h:body>
</html>
```

faces-config.xml

```
<?xml version="1.0"?>
<faces-config xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-facesconfig_2_0.xsd"
version="2.0">
```

```
<navigation-rule>
    <from-view-id>/calculator.xhtml</from-view-id>
    <navigation-case>
        <from-outcome>success</from-outcome>
        <to-view-id>/result.xhtml</to-view-id>
    </navigation-case>
</navigation-rule>
```

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