**Student ID__________________Student Name____________________________________**

## CS545 Web Applications Architecture and Frameworks

# DE Final Exam
# November 10, 2012

**PRIVATE AND CONFIDENTIAL**

- **Allotted exam duration is 2 hours.**
- **Closed book/notes.**
- **No personal items including electronic devices (cell phones, computers, calculators, PDAs).**
- **Cell phones must be turned in to your proctor before beginning exam.**
- **No additional papers are allowed. Sufficient blank paper is included in the exam packet.**
- **Exams are copyrighted and may not be copied or transferred.**
- **Restroom and other personal breaks are not permitted.**
- **Total exam including questions and scratch paper must be returned to the proctor.**

5 blank pages are provided for writing the solutions and/or scratch paper. All 5 pages must be handed in with the exam

**BE VERY CAREFUL WITH THE GIVEN 2 HOURS AND USE YOUR TIME WISELY. THE ALLOTED TIME IS GIVEN FOR EVERY QUESTION.**

**Write your name and student id at the top of this page.**

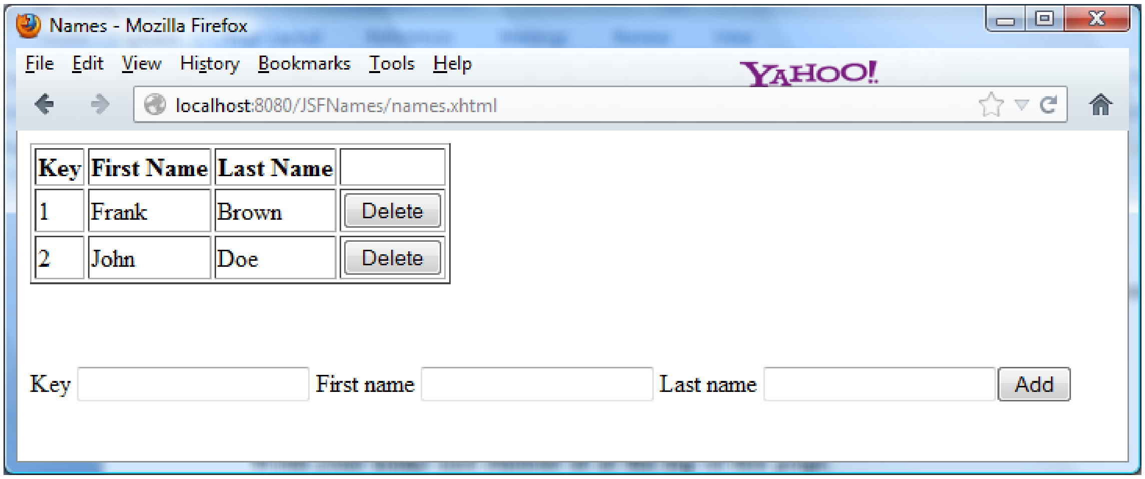
**Question 1 [ 10 points ] {10 minutes}**

Give the 6 phases of the JSF lifecycle and describe in one or two sentences what specific task is done in this phase

| **Phase** | **Specific task of this phase** |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |

**Question 2 [ 30 points ] {40 minutes}**

We need to write the following JSF application:


Names - Mozilla Firefox
File Edit View History Bookmarks Tools Help
YAHOO!
localhost:8080/JSFNames/names.xhtml
Key
First name
Last name
Add

| Key | First Name | Last Name | |
|---|---|---|---|
| 1 | Frank | Brown | Delete |
| 2 | John | Doe | Delete |

The application allows you to add persons to the list, and also remove them from the list.

Complete the partial given code. **<u>Do NOT write getter and setter methods!</u>**

```
public class Person {
  private String id;
  private String firstname;
  private String lastname;

    public Person(String id, String firstname, String lastname) {
        this.id = id;
        this.firstname = firstname;
        this.lastname = lastname;
    }
  //getters and setter methods are not shown here
}
```

## names.xhtml

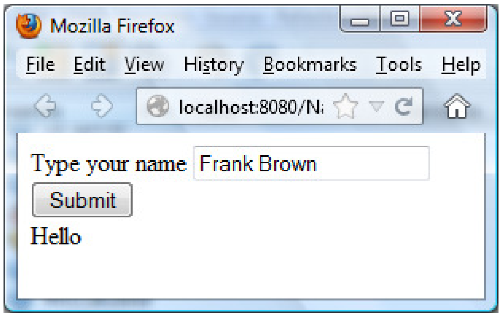
```
<?xml version='1.0' encoding='UTF-8' ?>
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:h="http://java.sun.com/jsf/html"
    xmlns:f="http://java.sun.com/jsf/core">
   <h:head><title>Names</title></h:head>
   <h:body>
      <h:form  >













         <p/>
         <p>
            Key
            <h:inputText value="#{names.key}" />
            First name
            <h:inputText value="#{names.fname}" />
            Last name
            <h:inputText value="#{names.lname}" />

            <h:commandButton value="Add"
                         action="#{names.add}"/>
         </p>

      </h:form>
   </h:body>
</html>
```

```
@ManagedBean
@RequestScoped
public class Names {
  private String key;
  private String fname;
  private String lname;
```

```
//getters and setter methods are not shown here
}
```

**Question 3 validation and conversion [ 20 points ] {20 minutes}**

We have to write the following application in JSF:



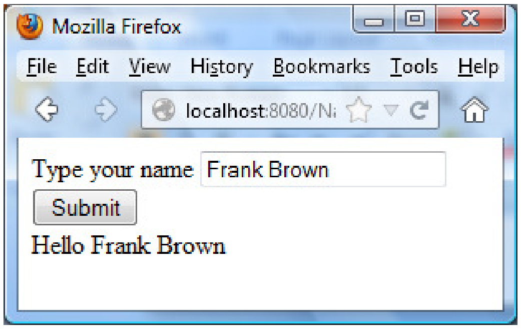
Type your name Frank Brown

Submit

Hello

If you fill in the name ‘Frank Brown’, it will be converted to a class Person with a firstName and lastName property:

```
public class Person {
   private String firstName;
   private String lastName;

//getter and setter methods are not shown
}
```

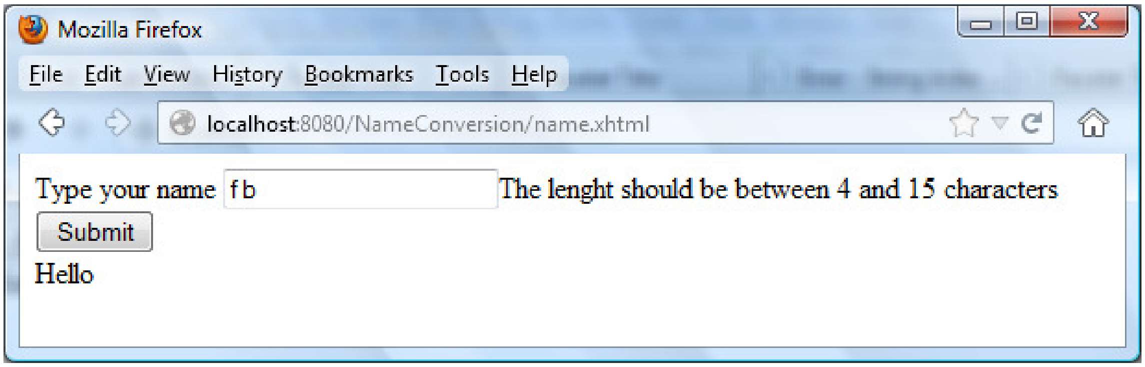


Type your name Frank Brown

Submit

Hello Frank Brown

When you click the Submit button, you write the text ‘Frank Brown’ in the given outputText control


Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:8080/NameConversion/name.xhtml

Type your name fb The lenght should be between 4 and 15 characters

Submit

Hello

We also want validation on the inputText control that the length of the name should be between 4 and 15 characters.

Complete the partial given code below. Make sure that

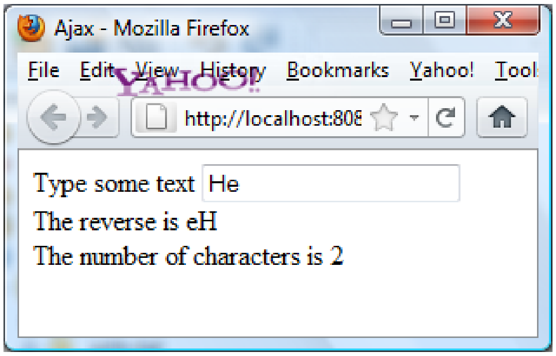
- You use a validator that checks if the length of the name is between 4 and 15 characters.
- You show the given message if validation fails
- You use a converter that converts a input String to a Person object and a Person object to an output String

**Do NOT write getter and setter methods!**

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

<h:body>
  <h:form>
    Type your name

  </h:form>
</h:body>
</html>
```

```
@ManagedBean
@RequestScoped
public class FormManagedBean {
  private Person person;

   public String sayHello(){
      return null;
   }
//getter and setter methods are not shown
}

public class Person {
   private String firstName;
   private String lastName;

//getter and setter methods are not shown
}



public class PersonConvertor



}
```

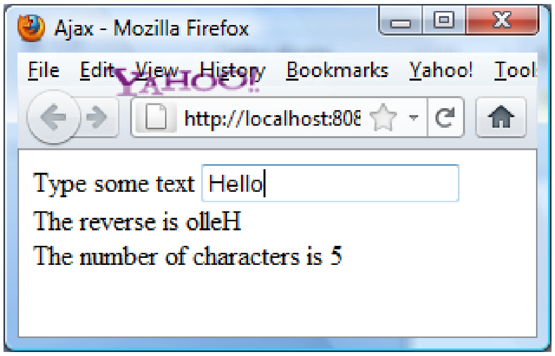
**Question 4 AJAX [ 15 points ] {20 minutes}**



Type some text He

The reverse is eH

The number of characters is 2



Type some text Hello

The reverse is olleH

The number of characters is 5

Write a JSF application with the following AJAX behavior:
Whenever you type something in the inputtext control, a request is send to the server, and the reverse of the entered text, and the number of characters entered is shown on the page. This behavior should be implemented with AJAX, such that only the reverse text and the number of characters are rendered.

You can get the reverse of a String with the following code:

***reverse= new StringBuffer(text).reverse().toString();***

Complete the partial given code such that the application works with the given behavior. You only have to complete the code for the xhtml file and the managed bean.
**IMPORTANT: do not write getter and setter methods!**

**ajax.xhtml**

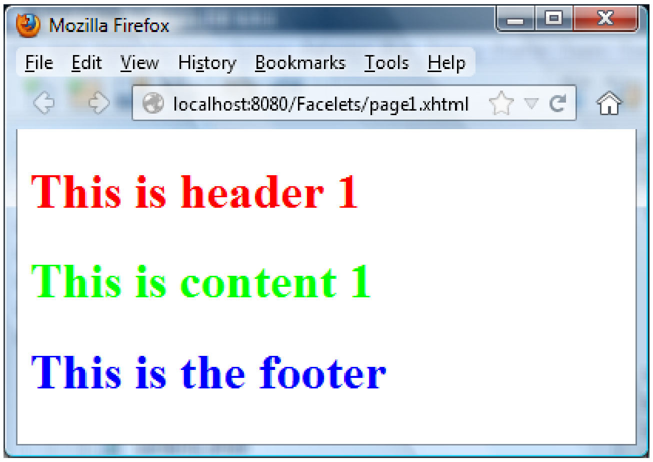
```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:f="http://java.sun.com/jsf/core">
   <h:head><title>Ajax</title></h:head>
   <h:body>
    <h:form>
         Type some text
         <h:inputText   ______________________________ >

            <f:ajax event="keyup" render="____________________________"

                                        listener="_____________________________"/>
         </h:inputText>
         <br />
         The reverse is <h:outputText id="reverse" _____________________________ />
         <br />
         The number of characters is <h:outputText id="count"

                                               ____________________________  />

      </h:form>
   </h:body>
</html>
```

```
@ManagedBean
@RequestScoped
public class Bean {

}
```

**Question 5 [ 20 points ] {20 minutes}**

We need to write a JSF application using facelets with the following 2 pages:

**<u>page1.xhtml</u>**

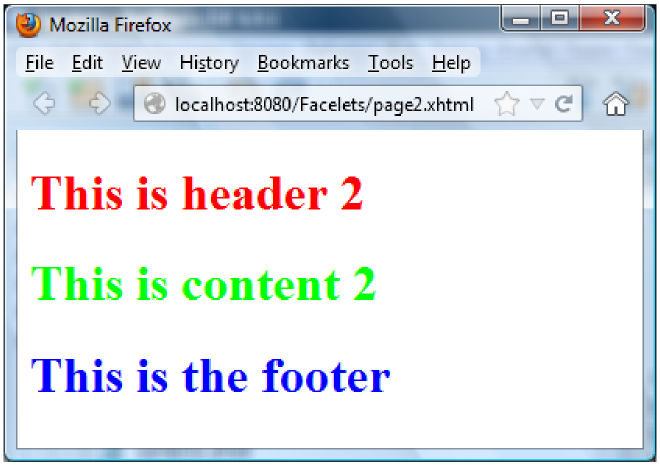

Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:8080/Facelets/page1.xhtml

This is header 1

This is content 1

This is the footer

**<u>page2.xhtml</u>**


Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:8080/Facelets/page2.xhtml

This is header 2

This is content 2

This is the footer

The following pages are given:

## header1.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:ui="http://java.sun.com/jsf/facelets">
  <body>
          <ui:composition>
                 <h1>This is header 1</h1>
          </ui:composition>
  </body>
</html>
```

## header2.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:ui="http://java.sun.com/jsf/facelets">
  <body>
          <ui:composition>
                 <h1>This is header 2</h1>
          </ui:composition>
  </body>
</html>
```

## content1.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:ui="http://java.sun.com/jsf/facelets">
  <body>
          <ui:composition>
                 <h1>This is content 1</h1>
          </ui:composition>
  </body>
</html>
```

## content2.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:ui="http://java.sun.com/jsf/facelets">
  <body>
          <ui:composition>
                 <h1>This is content 2</h1>
          </ui:composition>
  </body>
</html>
```

## footer.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://java.sun.com/jsf/html"
     xmlns:ui="http://java.sun.com/jsf/facelets">
   <body>
              <ui:composition>
                     <h1>This is the footer</h1>
              </ui:composition>
   </body>
</html>
```

Complete the partial given code of layout.xhtml which contains the layout of the template and page1.xhtml and page2.xhtml that will show as the pages given above.  You have to use Facelets for this application where you make use of the xhtml files given above. You don't get points if you create similar web pages without using facelets and without using the given xhtml pages given above.

## layout.xhtml

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

   <h:body>
           <div id="page">
              <div id="header" style="color:#FF0000">

                     <___________________________________ name="header" />
              </div>
              <div id="content" style="color:#00FF00">

                     <___________________________________ name="content" />
              </div>
         <div id="footer" style="color:#0000FF">

                     <___________________________________ name="footer" />
              </div>
      </div>
   </h:body>
</html>
```

## page1.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:ui="http://java.sun.com/jsf/facelets">
    <h:head>
       <title>Page 1</title>
    </h:head>
    <h:body>

    </h:body>
</html>
```

## page2.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:ui="http://java.sun.com/jsf/facelets">
    <h:head>
       <title>Page 2</title>
    </h:head>
    <h:body>

    </h:body>
</html>
```

**Question 6 SCI [ 5 points ] {10 minutes}**
Describe how we can relate JSF Facelets to the principles of SCI. Your answer should be about half a page, but should not exceed one page (handwritten). The number of points you get for this questions depend on how well you explain the relationship between JSF Facelets and the principles of SCI.