Министерство образования и науки Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение науки и высшего образования**

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

**Институт информационных технологий и радиотехники**

**(ИИТР)**

Кафедра информационных систем и программной инженерии

**Лабораторная работа № 06**

**по дисциплине**

**«Основы разработки веб-приложений с использованием каркаса JSF 2.0»**

Выполнил:

ст. гр. ПРИ-117

Хлызова В.Г.

Принял:

Трифонов Д.А.

Владимир, 2019

ЦЕЛЬ РАБОТЫ

Изучить особенности каркаса JavaServer Faces 2.0 для разработки веб-приложений (Facelets, аннотации, поддержка AJAX) и получить навыки реализации веб-приложений с использованием данного каркаса.

ВЫПОЛНЕНИЕ РАБОТЫ

В данной лабораторной работе была доработана программа из предыдущей лабораторной. Создан шаблон Facelets с шапкой и подвалом. Новые jsf страницы были созданы с использованием данного шаблона. Так же вместо обновления страницы при вызове методов обработки даты и числовых данных были реализованы ajax-запросы. Создан новый компонент интерфейса и применён к jsf странице.

Страницы jsf с использованием шаблона:

* Страница входа

*<?***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:ui="http://xmlns.jcp.org/jsf/facelets"  
 xmlns:h="http://xmlns.jcp.org/jsf/html"  
 xmlns:f="http://xmlns.jcp.org/jsf/core"**>  
  
 <**body**>  
 // компонент на использование шаблона  
 <**ui:composition template="./NewTemplate.xhtml"**>  
 // шапка страницы с выводом кол-ва посещений  
 <**ui:define name="top"**>  
 <**h:form**>   
 <**h:panelGrid columns="1" style="font-size**: 20**px**;**"**>  
 <**h:outputText value="Кол-во обращений к странице: #{sessionBean.numb}"** />   
 </**h:panelGrid**>  
 </**h:form**>   
 </**ui:define**>  
 //основной контент и вводом имени и проверкой пароля  
 <**ui:define name="content"**>  
 <**h:form**>  
 <**h:panelGrid columns="2" style="font-size**: 20**px**;**"**>  
 <**h:outputText value="Введите логин"**/>  
 <**h:inputText value="#{sessionBean.login}" validatorMessage="Введите логин на английском языке   
 с большой буквы, в имени не должны присутствовать символы цифр,   
 длина текста от 3 до девяти символов"**>  
 <**f:validateLength minimum="3" maximum="9"** />  
 <**f:validateRegex pattern='[A-Z]{1}[a-z]\*'** />  
 </**h:inputText**>  
 <**h:outputText value="Введте пароль"** />  
 <**h:inputSecret value="#{client.password}" validatorMessage="длина пароля максимум три символа"**>  
 <**f:validateLength maximum="3"** />  
 </**h:inputSecret**>  
 <**h:commandButton value="GO!" style="font-size**: 15**px**;**" action="#{client.checkLogin()}"**/>  
 </**h:panelGrid**>  
 </**h:form**>   
 </**ui:define**>  
 //подвал страницы с выводом дополнительной информацией  
 <**ui:define name="bottom"**>  
 <**h:form**>   
 <**h:panelGrid columns="1" style="font-size**: 20**px**;**"**>  
 <**h:outputText value="Лабораторная работа"**/>  
 <**h:outputText value="Основы разработки веб-приложений с использованием каркаса JSF 2.0"** />   
 <**h:outputText value="Выполнил студент групы ПРИ-117"**/>  
 <**h:outputText value="Хлызова В.Г."**/>  
 </**h:panelGrid**>  
 </**h:form**>   
 </**ui:define**>  
  
 </**ui:composition**>  
  
 </**body**>  
</**html**>

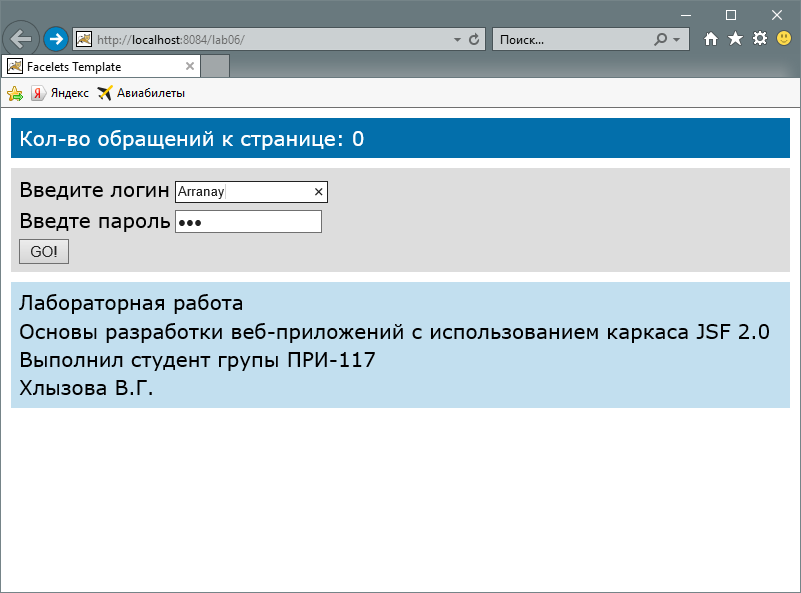


Рисунок 1. Пример выполнения программы.

* Страница после успешной обработки пароля

*<?***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:ui="http://xmlns.jcp.org/jsf/facelets"  
 xmlns:h="http://xmlns.jcp.org/jsf/html"  
 xmlns:f="http://xmlns.jcp.org/jsf/core"  
 xmlns:mc="http://xmlns.jcp.org/jsf/composite/ezcomp"**>  
  
 <**body**>  
  
 <**ui:composition template="./newTemplate.xhtml"**>  
 <**mc:NewComponent buttonGo="#{client.length()}"   
 nDay="#{client.numbDay}"  
 sDay="#{client.significantDay}"**/>  
   
 <**ui:define name="top"**>  
   
 <**h:form**>   
 <**h:panelGrid columns="1" style="font-size**: 20**px**;**"**>  
 <**h:outputText value="#{sessionBean.login}, добро пожаловать"** />   
 </**h:panelGrid**>  
 </**h:form**>   
 </**ui:define**>  
  
 <**ui:define name="content"**>  
 //применение созданного компонента   
 <**mc:NewComponent buttonGo="#{client.length()}"   
 nDay="#{client.numbDay}"  
 sDay="#{client.significantDay}"**/>   
  
 <**h:form id="form1"**>  
 <**h:panelGrid columns="8" style="font-size**: 20**px**;**"**>  
 <**h:inputText id="slag1" value="#{client.slag1}" validatorMessage=" Первое слагаемое должно иметь вид числа в диапазоне от 1 до 100"**>  
 <**f:validateLongRange minimum="1" maximum="100"** />  
 </**h:inputText**>  
 <**h:commandButton value="+" action="#{client.sum(client.slag1, client.slag2)}"**>  
 <**f:ajax execute = "@all" render = "value1"**/>  
 </**h:commandButton**>  
 <**h:commandButton value="-" action="#{client.dif(client.slag1, client.slag2)}"**>

//пример ajax-запроса. В атрибуте execute обрабатываются все компоненты формы. В атрибуте render отображается компонент с id value1. Далее все ajax-запросы строятся по такому же принципу. Запрос срабатывает при нажатие на кнопку.  
 <**f:ajax execute = "@all" render = "value1"**/>  
 </**h:commandButton**>  
 <**h:commandButton value="\*" action="#{client.mult(client.slag1, client.slag2)}"**>  
 <**f:ajax execute = "@all" render = "value1"**/>  
 </**h:commandButton**>  
 <**h:commandButton value="/" action="#{client.sub(client.slag1, client.slag2)}"**>  
 <**f:ajax execute = "@all" render = "value1"**/>  
 </**h:commandButton**>  
 <**h:inputText id="slag2" value="#{client.slag2}" validatorMessage=" Второе слагаеое должно иметь вид числа в диапазоне от 1 до 100"**>  
 <**f:validateLongRange minimum="1" maximum="100"** />  
 </**h:inputText**>  
 <**h:outputText value="="**/>  
 <**h:outputText id="value1" value="#{client.result}"** />   
 </**h:panelGrid**>  
 </**h:form**>  
 </**ui:define**>  
  
 <**ui:define name="bottom"**>  
 <**h:panelGrid columns="2" style="font-size**: 20**px**;**"**>  
 <**h:link value="Вернутся к проверке пароля" style="color**: **black" outcome="indexForTemplate"** />  
 </**h:panelGrid**>  
 </**ui:define**>  
  
 </**ui:composition**>  
  
 </**body**>  
</**html**>

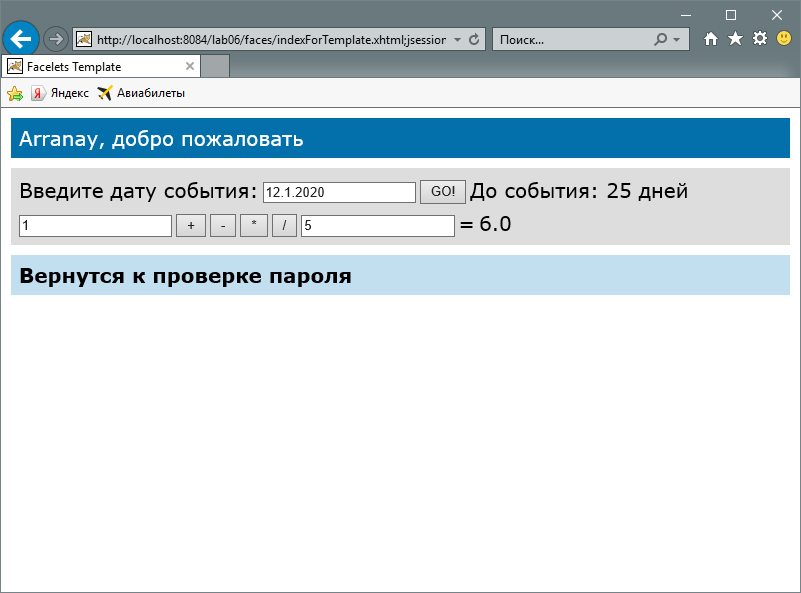


Рисунок 2. Пример выполнения программы.

Управляемые бины:

**package** client;  
  
**import** java.util.Date;  
**public class** Client { **private** String password;   
 **private** Date significantDay;  
 **private static** String numbDay = **"(￢\_￢)"**;  
   
 **public** Date getSignificantDay() {  
 **return** significantDay;  
 }   
 **public void** setSignificantDay(Date significantDay) {  
 **this**.significantDay = significantDay;  
 }  
 **public** String getNumbDay() {  
 **return** numbDay;  
 }   
   
   
 **public** String getPassword() {  
 **return** password;  
 }   
 **public void** setPassword(String password) {  
 **this**.password = password;  
 }  
   
   
   
 **public** String checkLogin(){  
 **if** (password.equalsIgnoreCase(**"111"**)){  
 **return "success"**;  
 } **else** {  
 **return "failed"**;  
 }  
 }  
   
 **public void** length(){  
 Date date = **new** Date();  
 **boolean** flag = **true**;  
 **if** (significantDay.after(date)){  
 **long** time = significantDay.getTime()- date.getTime();  
 time = time/(24\*60\*60\*1000)+1;  
 **if** (time==1) {numbDay = **"До события: "**+time+**" день"**; }  
 **if** ((time>1)&&(time<5)) {numbDay = **"До события: "**+time+**" дня"**;}   
 **if** (time>4) {numbDay = **"До события: "**+time+**" дней"**;}   
 flag=**false**;  
 }  
 **if**(flag) numbDay = **"Событие уже прошло"**;  
 }  
   
 **private int** slag1;  
 **private int** slag2;  
 **private static double** result;  
   
 **public void** setSlag1(**int** slag1) {  
 **this**.slag1 = slag1;  
 }   
 **public void** setSlag2(**int** slag2) {  
 **this**.slag2 = slag2;  
 }   
 **public double** getResult() {  
 **return** result;  
 }  
 **public int** getSlag1() {  
 **return** slag1;  
 }  
 **public int** getSlag2() {  
 **return** slag2;  
 }  
   
 **public void** sum(**int** slag1, **int** slag2){  
 result=slag1+slag2;  
 }  
   
 **public void** dif(**int** slag1, **int** slag2){  
 result = slag1-slag2;  
 }  
   
 **public void** mult(**int** slag1, **int** slag2){  
 result = slag1\*slag2;  
 }  
   
 **public void** sub(**int** slag1, **int** slag2){  
 result = (**double**)slag1/(**double**)slag2;  
 }  
}

**package** session;  
**public class** sessionBean {  
   
 **private static int** numb;  
 **private** String login;  
   
 **public** String getLogin() {  
 **return** login;  
 }  
 **public void** setLogin(String login) {  
 **this**.login = login;  
 }  
   
 **public int** getNumb(){  
 **return** numb++;  
 }   
   
}

Новый компонент интерфейса:

*<?***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:cc="http://xmlns.jcp.org/jsf/composite"  
 xmlns:h="http://xmlns.jcp.org/jsf/html"  
 xmlns:f="http://xmlns.jcp.org/jsf/core"**>  
  
 *<!-- INTERFACE -->* <**cc:interface**>  
 <**cc:attribute name="sDay"**/>  
 <**cc:attribute name="nDay"**/>  
 <**cc:attribute name="buttonGo" method-signature="java.lang.String action()"**/>  
 </**cc:interface**>  
  
 *<!-- IMPLEMENTATION -->* <**cc:implementation**>  
 <**h:form**>  
 <**h:panelGrid columns="4" style="font-size**: 20**px**;**"**>   
 <**h:outputText value="Введите дату события:"** />  
 <**h:inputText id="inputDate" value="#{cc.attrs.sDay}"**>  
 <**f:convertDateTime dateStyle="short"** />  
 </**h:inputText**>  
 <**h:commandButton value="GO!" action="#{cc.attrs.buttonGo}"**>  
 <**f:ajax execute = "inputDate" render = "outputDate"**/>  
 </**h:commandButton**>  
 <**h:outputText id="outputDate" value="#{cc.attrs.nDay}"** />  
 </**h:panelGrid**>  
 </**h:form**>  
 </**cc:implementation**>  
</**html**>

ВЫВОД

В ходе лабораторной работы были изучены особенности каркаса JavaServer Faces 2.0 для разработки веб-приложений (Facelets, аннотации, поддержка AJAX) и получены навыки реализации веб-приложений с использованием данного каркаса.