FRONTEND

01

JSF (Java Server Faces)

WEB BROWSER

Muy utilizado a nivel empresarial

EMPEZAR

Framework de Java

ÍNDICE

- DEFINICIÓN
- VENTAJAS
- CONFIGURACIÓN
- USO

Patrón de diseño

¿QUÉ ES JSF?

Framework basado en MVC

Proporciona un conjunto de componentes en forma de etiquetas definidas en páginas XHTML.

Antes de la especificación actual se utilizaba JSP para la creación de nuestras pantallas.

REGRESAR

VENTAJAS

¿POR QUÉ UTILIZARLO?

- PATRÓN DE DISEÑO MVC
- FRONTEND - BACKEND
- GRAN CANTIDAD DE COMPONENTES
- CURVA DE APRENDIZAJE MÍNIMA

REGRESAR

¿CÓMO FUNCIONA?

test-jsf/pom.xml | *login.xhtml | LoginController.java

```
1  <html xmlns="http://www.w3.org/1999/xhtml"
2        xmlns:f="http://java.sun.com/jsf/core"
3        xmlns:h="http://java.sun.com/jsf/html">
4
5      <h:head>
6          <title>Login DevPredator</title>
7      </h:head>
8      <h:body>
9          <h:form>
10
11             <h:outputLabel>Login - DEVPREDATOR</h:outputLabel>
12
13             <h:panelGrid columns="2" border="1">
14                 <h:outputText value="Usuario: "></h:outputText>
15                 <h:inputText value="#{loginController.username}"></h:inputText>
16
17                 <h:outputText value="Contraseña: "></h:outputText>
18                 <h:inputSecret value="#{loginController.password}"></h:inputSecret>
19
20                 <h:outputLabel></h:outputLabel>
21                 <h:commandButton value="Entrar" action="#{loginController.ingresar}" style="width: 10
22             </h:panelGrid>
23         </h:form>
24     </h:body>
25 </html>
```

XHTML

PASO 1

Se debe de crear un archivo con extensión xhtml como el que se encuentra en la imagen.

¿CÓMO FUNCIONA?

test-jsf/pom.xml | login.xhtml | LoginController.java

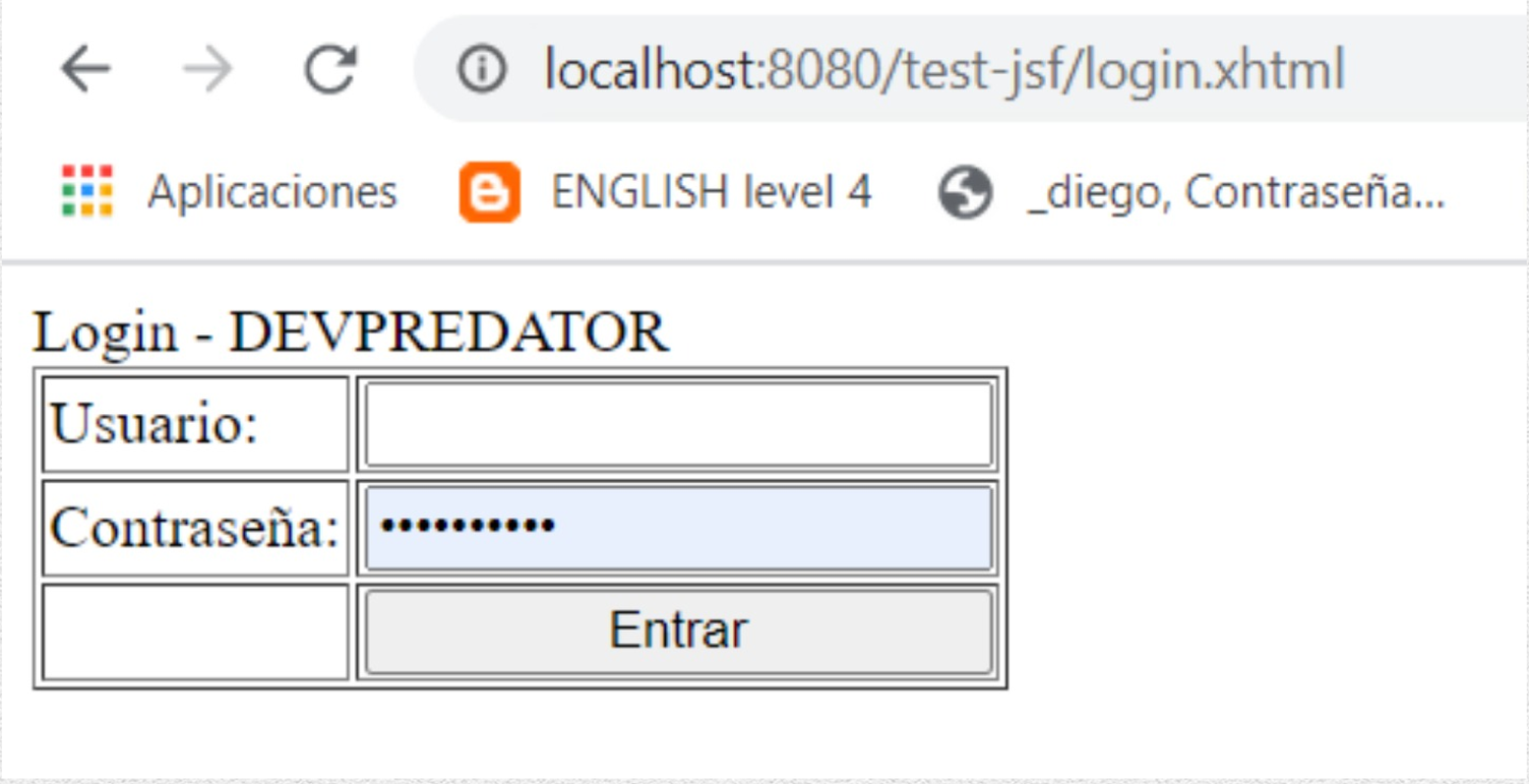
```java
 * @author DevPredator
 * Clase para comunicarse con la pantalla de login.
 */
@ManagedBean
@ViewScoped
public class LoginController {

    /**
     * Nickname del usuario.
     */
    private String username;
    /**
     * Contraseña del usuario.
     */
    private String password;
    /**
     * Metodo que permite ingresar al home del usuario.
     */
    public void ingresar() {
        System.out.println("Ingresando al home...");
    }

    /**
     * @return the username
     */
    public String getUsername() {
        return username;
    }
```

CONTROLLER

PASO 2

El controlador se encargará de comunicar las propiedades y métodos con la pantalla xhtml.

¿CÓMO FUNCIONA?



NAVEGADOR

PASO 3

Como se debe de ver en Chrome, Safari, Explorer, Firefox, etc...

CONFIGURACIÓN

Solo hay que agregar las dependencias en el archivo pom.xml

test-jsf/pom.xml | login.xhtml | LoginController.java

```
20          <maven.compiler.target>14</maven.compiler.target>
21      </properties>
22
23      <dependencies>
24          <dependency>
25              <groupId>junit</groupId>
26              <artifactId>junit</artifactId>
27              <version>4.11</version>
28              <scope>test</scope>
29          </dependency>
30
31          <!-- https://mvnrepository.com/artifact/com.sun.faces/jsf-api -->
32          <dependency>
33              <groupId>com.sun.faces</groupId>
34              <artifactId>jsf-api</artifactId>
35              <version>2.2.20</version>
36          </dependency>
37
38          <dependency>
39              <groupId>com.sun.faces</groupId>
40              <artifactId>jsf-impl</artifactId>
41              <version>2.2.20</version>
42          </dependency>
43
44      </dependencies>
```

CONFIGURACIÓN

Agregar en el archivo web.xml el mapeo de las páginas de las pantallas.

test-jsf/pom.xml | login.xhtml | LoginController.java | web.xml

```xml
    http://java.sun.com/xml/ns/javaee/web-app_3_1.xsd"
    id="WebApp_ID" version="3.1">

  <display-name>Archetype Created Web Application</display-name>

    <!-- Change to "Production" when you are ready to deploy -->
    <context-param>
        <param-name>javax.faces.PROJECT_STAGE</param-name>
        <param-value>Development</param-value>
    </context-param>

    <!-- Welcome page -->
    <welcome-file-list>
        <welcome-file>login.xhtml</welcome-file>
    </welcome-file-list>

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

    <!-- Map these files with JSF -->
    <servlet-mapping>
        <servlet-name>Faces Servlet</servlet-name>
        <url-pattern>*.xhtml</url-pattern>
    </servlet-mapping>

</web-app>
```

REGRESAR