



## Universidad Autónoma de Nuevo León Facultad de Ciencias Físico Matemáticas

## Laboratorio 5

## **LDOO**

**Maestro: Miguel Salazar** 

**Alumno:** 

William Guillermo Villegas Rodríguez

**Matricula: 1660542** 

Crear un proyecto de tipo Java Web en NetBeans de nombre Laboratorio5.

- 2.- Elimina el archivo index.html que crea por defecto NetBeans y que se encuentra dentro de la carpeta Web Pages.
- 3.- Utiliza los archivos del Laboratorio4:

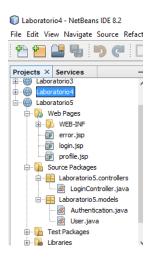
Copia y pega los archivos login.jsp y error.jsp del proyecto Laboratorio4.

Crea un nuevo servlet de nombre LoginController dentro del paquete Laboratorio5.controllers y agrega el código del controlador del Laboratorio4. En este punto, asegúrate de que se ha creado tu archivo web.xml dentro de la carpeta WEB-INF.

De la misma manera que el punto anterior, utiliza los modelos creados en el proyecto Laboratorio4. No olvides crear el paquete de los modelos.

4.- Agrega un nuevo archivo de nombre profile.jsp dentro de Web Pages.

Se deberían de ver de esta manera



El modelo User original contiene dos propiedades username y password. Deberás realizar los siguientes cambios:

- Agregar una nueva propiedad Nombre de tipo String y privada.
- Agregar una nueva propiedad Apellidos de tipo String y privada.
- Ambas propiedades tendrán información dummy como se creó en el Laboratorio4.
- Crear tres métodos: String getName(), String getLastName() y String getFullName(). Estos métodos deberán devolver el nombre, apellidos y el nombre completo (nombre + apellidos). El método getFullName() deberá hacer uso de los dos primeros métodos para su funcionamiento.

```
Source History | 😭 👺 + 🐺 + 💆 💆 😓 👺 📮 😭 - 😂 😂 😂 | 🚳 🔠 🏙 🚅
                                        public class User {
                                   12
  - Aboratorio 5
                               public class User (
private String usuario;
private String pass;
private String nombre;
private String apellido;
                                 public User(String usuario, String pass) {

this.usuario = usuario;

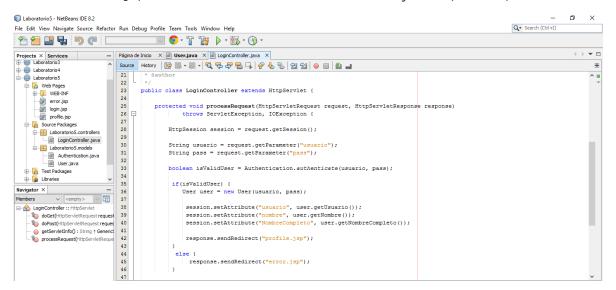
this.nasa = pass;

this.nombre = "William";
                                   this.nombre = "William";

this.apellido = "Villegas";

3
           ■ LoginController.java
      Laboratorio 5. models
         Authentication.java
User.java
                                   24
25 public String getUsuario() {
    return usuario;
     - Test Packages
   ⊕ 🔓 Libraries
           User - Navigator ×
                                   30 | }
31 | public String getApellido(){
32 | return apellido;
33 | }
      User(String usuario, String pass)
     getNombre(): String
                                            public String getNombreCompleto(){
      getNombreCompleto(): String
      getUsuario() : String
                                 34 = put
35 | 36 | }
                                                   return getNombre() + "" + getApellido();
```

El controlador LoginController original, en su método processRequest, realizaba una validación del usuario. De ser correcta, enviaba a la página success.jsp; de ser falsa, enviaba a error.jsp. Lo anterior se realizaba mediante el objeto RequestDispatcher



La página profile.jsp deberá aparentar el perfil del usuario que está iniciando sesión.

- Deberá mostrar un mensaje de bienvenida al usuario similar a la página success.jsp del Laboratorio4.
- Además, debemos agregar elementos de HTML para mostrar el nombre completo y el usuario.
- Validaremos la sesión del usuario al iniciar la página. Supongamos que alguien intenta tener acceso directamente a esta página tecleando en la url. Se deberá validar que puede entrar a la página si su sesión ya esta creada.

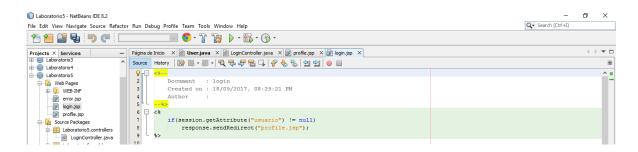
```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Projects X Services

| Radjon de Dico X | Buerjava X | Supricontroller/ywa X | profile part | page | profile | pro
```

Supongamos que alguien ya inició sesión, si entrara de nuevo a la página de login ¿Tiene sentido pedir nuevamente las credenciales? Of course, not my friend.

 Validar mediante la sesión si debe o no presentarse el formulario de login.jsp.



- Identificas alguna ventaja de usar sesiones en comparación al Laboratorio 4? ¿Cuáles?
  - R: Si tienes que hacer menos páginas y escribimos menos código
- Si requerimos más información del usuario, ¿Consideras que debe seguirse trabajando con sesiones?
  - R: A mi parecer si por que nos podría facilitar el trabajo
- ¿Qué ventajas tiene pasar información con sesión y no mediante valores en el request como en el Laboratorio 4?
  - R: Es más cómodo y más seguro para los datos