

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

Practica 9

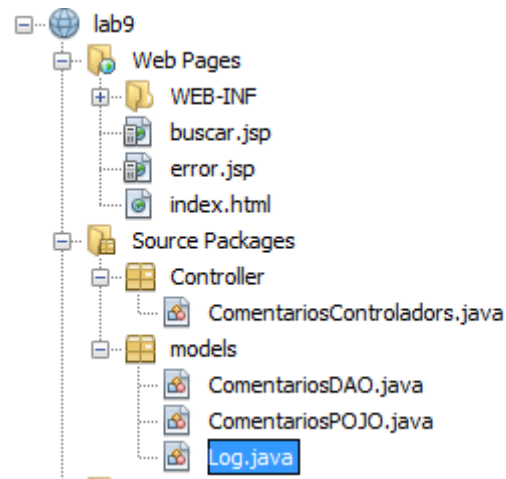
Estudiante: Jorge Eduardo Castro Tristan

Matricula: 1640167

Materia: LDOO

Fecha: 04/04/16

En esta práctica 9 lo que hicimos fue realizar el modelo MVC para crear una aplicación web que incluiría un jsp para buscar, otro para ingresar un comentario y adicional a esto debe crear un log que nos mostrara la fecha y la hora de la interacción.



Lo primero es crear dos archivos de JSP, uno para buscar los parámetros deseados ya sea por comentario o por usuario, y otro de error que nos ayudara a advertir al usuario que hubo un problema realizando la acción que se requería, además ahí mismo se agrega un link que nos llevara de vuelta al inicio, y el index que mostrara la pantalla inicial de la aplicación, adicional a esto tenemos un controlador y 3 clases de java que nos servirán como modelo.

```
<form action="ComentariosControladors" method="POST" >
  <h1>Buscar</h1>
  Nombre:<input type="text" value="" name="nombre"><br><br>
  Comentario:<br>
  <textarea rows="4" cols="15" name="comentario"></textarea><br><br>
  <input type="submit">
  <input type="hidden" name="action" value="buscar">
</form>
```

El formulario de nuestra pagina "buscar" nos permitirá ingresar el nombre y un comentario, además de un botón de tipo submit para poder hacer el envío de la información.

```

<% if(session != null){
    ArrayList comentarios = (ArrayList)session.getAttribute("comentarios");
    if(comentarios!=null){
%>
        <table border="1">
            <tr>
                <th>Nombre: </th>
                <th>Comentario: </th>
            </tr>

            <%
                for(Object o : comentarios){
                    ComentariosPOJO comentario = (ComentariosPOJO) o;
            %>
                <tr>
                    <td><%=comentario.getNombre() %></td>
                    <td><%=comentario.getComentario() %></td>
                </tr>
            <% } %>
        </table>
    <% } %>
} %>

```

En este JSP también de “buscar” lo que hara es crear un arreglo de comentarios que posteriormente nos los imprimirá según lo que el usuario haya buscado, se buscara por nombre y por comentario.

```

public ArrayList buscar(ComentariosPOJO pojo){
    ArrayList<ComentariosPOJO> comentarios = new ArrayList();
    try{
        abrirConexion();
        String sqlBuscar = "select * from COMENTARIOS where NOMBRE = ";
        Statement statement = conexion.createStatement();
        ResultSet result = statement.executeQuery(sqlBuscar);
        while(result.next()){
            String nombre = result.getString("Nombre");
            String comentario = result.getString("Comentario");
            ComentariosPOJO coment = new ComentariosPOJO();
            coment.setNombre(nombre);
            coment.setComentario(comentario);
            comentarios.add(coment);
        }
        cerrarConexion();
    }catch(Exception ex){
    }
    return comentarios;
}

```

Esto forma parte del modelo de ComentariosDAO, lo que hace básicamente es realizar la interaccion de manera directa con el servidor, es el que se encarga de buscar la información que se le va a mostrar al usuario, en este fragmento del código, lo que se hace es crear una conexión con el servidor en la que se va a obtener el valor especificado.

```
//Obtener Instancia y establecer ruta

Log L = Log.getInstance("C:\\Users\\Jorge\\Desktop\\Log.txt");
L.write(nombre);
if(action.equals("comentar")){

    cdao.insertar(pojo);

    response.sendRedirect("buscar.jsp");
}
if(action.equals("buscar")){

    ArrayList <ComentariosPOJO> comentarios = cdao.buscar(pojo);

    session.setAttribute("comentarios", comentarios);

    response.sendRedirect("buscar.jsp");
}
if(!"comentar".equals(action) & !"buscar".equals(action)){
    response.sendRedirect("error.jsp");
}
}
```

El modelo de java de Log es el que se encargara de llevar un registro de los sucesos importantes o de depuración de una aplicación Java. Con la finalidad de generar archivos de eventos para futuros diagnósticos de la aplicación. En este caso nos dara la hora, día, mes y año en el que el usuario realizo interaccion con la aplicación.

Preguntas

- ¿Qué ventajas identificas con el uso de un sistema de Logging de eventos?

Nos permite saber de manera mas especifica porque la aplicación tuvo un fallo, y en caso de que la seguridad de esta se vea vulnerada también saber que fue lo que se hizo para llegar a eso.

- ¿Qué ventajas tienes al utilizar una clase singleton?

Al proveer una única instancia, esto hace que la aplicación o el código con el que se este trabajando sea mas sencillo, se puedan evitar errores futuros y mejora el rendimiento teniendo menos líneas de código.

- ¿Qué "pros" y "contras" identificas al utilizar singleton vs clases estáticas?

Entre las ventajas contra las clases estáticas podría ser que nos da mas facilidad de crear clases al poder usar, herencia, polimorfismo e implementaciones, yo creo que uno de los casos en los que seria mas conveniente una clase estatica es cuando solo de acceso en general a todos sus métodos por la rapidez.