

Práctica 2

Autor: Juan Pablo González Casado

Descripción: Documentación de práctica 2 de DSS

Fecha: 15/Nov/2016

Introducción

En esta práctica se trata de realizar una aplicación con persistencia. Almacenando una lista de correos junto con el nombre y apellido del usuario.

Para ello se utiliza Apache Derby, Apache Tomcat y las librerías de persistencia de Java.

El proyecto se divide en tres paquetes:

Paquete Interfaz

En el se encuentre una interfaz basada en swing, en la que se puede interaccionar añadiendo usuarios, borrando y modificando sus valores de nombre y apellido. Cada vez que se realiza alguna de estas acciones, se utilizan llamadas POST con la acción requerida y los parámetros necesarios.

Este código ha sido proporcionado por el profesor.

Paquete Modelo

Se han implementado dos clases:

Usuario

Justo antes comenzar la clase, vemos una etiqueta *@Entity*, utilizada para indicar que se trata de una entidad de la que se guardará persistencia.

Por lo demás, es una clase básica con 3 atributos identificativos, sus gets/sets y un constructor de copia.

BDUsuario

Esta clase es la encargada de hacer las conexiones a Apache Derby

El flujo común de las funciones con base de datos es el siguiente:

1. Creamos el EntityManagerFactory

```
factoria = Persistence.createEntityManagerFactory(PERSISTENCE_UNIT_NAME); EntityManager
```

```
em = factoria.createEntityManager();
```

2. Iniciamos una transacción

```
em.getTransaction().begin();
```

3. Creamos la consulta 'SQL' y establecemos parámetros

```
Query q = em.createQuery("SELECT u FROM Usuario u WHERE u.email = :email");  
q.setParameter("email", usuario.getEmail());
```

4. Comprobamos la cantida de resultados devueltos

```
q.getResultList().size() == 0
```

5. Si todo sale correctamente guardamos y realizamos commit

```
em.persist(usuario); em.getTransaction().commit();
```

6. Una vez terminado, se cierra la transacción

```
em.close();
```

Paquete com

En este se encuentra lla clase **ListaCorreoServlet** que contiene el servlet que se encarga de realizar la conexión entre el cliente y el servicio.

Si no se indica acción, se consultará la lista de usuarios

```
if(accion == null)  
    //pintamos en html  
for (Usuario usuario: BDUsuario.listarUsuarios())  
    writer.println("<h3>"+usuario.getCadena()+"</h3>");
```

En el caso de indicar una acción, se detecta y se llamará

```
switch (accion) {  
    case "aniadirUsuario":  
        if (BDUsuario.select(email)==null) {  
            Usuario usuario = new Usuario();  
            usuario.setNombre(nombre);  
            usuario.setApellido(apellido);  
            usuario.setEmail(email);  
            BDUsuario.insert(usuario);  
            oos.writeInt(0);  
            oos.writeObject("Usuario aniadido  
correctamente.");  
        } else {  
            oos.writeInt(1);  
        }  
    }  
}
```

```
        oos.writeObject("Ya existe  
        un usuario con el email " + email + ".");  
    }  
    break;
```

Detalles a tener en cuenta

Es importante modificar las rutas en los archivos de configuración. Los archivos de configuración son: ***persistence.xml*** ***web.xml***

La clase **ListaCorreoServlet** debe de ser ejecutada con Apache Tomcat La clase **Cliente** debe de ser ejecutada como aplicación Java.