

Release Note

Programación OO con persistencia de entidades

Descripción

En la presente práctica se ha realizado un sistema para almacenar listas de correos de usuarios en una base de datos utilizando para ello JPA (Java Persistente API). A través de esta interfaz de programación se ha podido llevar a cabo el proceso de transformación de clase Java a Entidad para su posterior persistencia en disco.

La práctica estaba definida en dos partes; una parte de aplicación web y otra de aplicación escritorio. Para la parte de aplicación web se ha utilizado la tecnología de los **servlets**, y Apache Tomcat como contenedor de estos para poder redireccionar las peticiones del cliente web al servlet correspondiente que previamente estuviese registrado. Para la parte de la aplicación escritorio ha sido la librería Swing de Java la que se ha empleado en el desarrollo del mismo.

Como se ha podido comprobar durante todo el desarrollo, eran tres paquetes los que componían el sistema. El paquete modelo, el cual incluía la entidad que se almacenaba y la clase que hacía de enlace con la misma. El paquete interfaz, que contenía todo lo referido a la aplicación cliente de escritorio y que se comunicaba con el paquete modelo. Y el paquete comunicación, donde se desarrollaba el servlet que servía al cliente web las peticiones y que también se comunicaba con el paquete modelo.

Tecnologías y herramientas

Son varias las librerías y herramientas empleadas en el desarrollo de la práctica, comenzando por el IDE en el que se ha basado su desarrollo, **Eclipse**, que trae todo un entorno de desarrollo para aplicaciones JEE, Faces Projects, Web Dynamic Module, etc.

Apache Derby ha sido el sistema gestor de base de datos que se ha utilizado para tratar con la base de datos donde se almacenan las entidades persistentes.

EclipseLink es el ORM que implementa JPA utilizado para la creación de entidades a partir de clases Java junto con la **javax.persistence** en su versión 2.1.1.

El api **servlet** 3.0 ha sido el que se ha empleado para la implementación de estos módulos.

Para la parte del servidor web, **Apache Tomcat** en las versiones v8.0 y v8.5 han sido las utilizadas en el desarrollo y prueba del proyecto.

Modo de operación

Para poder probar esta práctica, es necesario tener instaladas todas las librerías y dependencias mencionadas en el apartado anterior. Una vez que se cumple con todos esos requisitos se ha de seguir los siguientes pasos para su ejecución:

1. Ejecutar en modo servidor (Run as Server) el fichero **ListaCorreosServlet.java**.
2. Cuando arranque el servlet se abriría una pagina del navegador web, la parte del cliente, que mostrará una lista vacía.
3. Lo siguiente será ejecutar la aplicación escritorio cliente, para ello es necesario ejecutar en modo aplicación Java (Java Application) el fichero **Cliente.java**.
4. Una vez ejecutado el cliente se podrá añadir, modificar y eliminar usuarios de la base de datos, esas transacciones se verán también reflejadas en el navegador web cuando se vaya refrescando.

Es importante indicar que la operaciones de insertar, modificar y eliminar sólo pueden realizarse desde la aplicación cliente de escritorio, quedando la el navegador web como un simple lector de datos.