



## PRÁCTICA 4A: PRUEBAS DE SOFTWARE

### 1. Objetivos

- Aprender a diseñar casos de prueba unitarios aplicando técnicas de caja negra y caja blanca para un sistema software orientado a objetos.
- Aprender a ejecutar pruebas unitarias utilizando un framework de pruebas, en este caso formado por las herramientas *JUnit* y *EclEmma*.
- Aprender a ejecutar pruebas de integración en aplicaciones basadas en interfaces gráficas SWING usando la librería *AssertJ Swing*.

### 2. Actividades

El alumno deberá llevar a cabo el proceso (parcial) de diseño y ejecución de pruebas unitarias y de integración para la aplicación que se desarrolló en la práctica 3. Para las pruebas de caja negra se debe aplicar las técnicas de partición equivalente y AVL y para las pruebas de caja blanca se requiere cobertura de sentencias y de decisión/condición.

NOTA: Para el desarrollo de esta práctica se deberá realizar una copia de los proyectos desarrollados en la Práctica 3 en la carpeta correspondiente a la Práctica 4 y trabajar sobre ellos.

Los pasos a seguir para llevar a cabo el proceso de prueba son los siguientes:

1. Pruebas unitarias de la clase `Empleado`.
  - i. Diseñar casos de prueba para la clase `Empleado`.
  - ii. Escribir el código de la clase de prueba correspondiente utilizando *JUnit*.
  - iii. Ejecutar la clase de prueba y corregir defectos.
  - iv. Cuando las pruebas de caja negra sean correctas, comprobar la cobertura.
  - v. Diseñar nuevos casos de prueba para alcanzar la cobertura deseada en caso de no haberse alcanzado con las pruebas de caja negra.

NOTA: Dentro de la fase de pruebas unitarias de la aplicación, deberían probarse el resto de clases de la capa de dominio (Tienda) y las clases de las capas dao, negocio y presentación. Sin embargo, no se aborda su desarrollo en esta práctica.

Al tratarse de pruebas de integración, para distinguirlas de las pruebas unitarias, el nombre de la clase de prueba deberá incorporar la cadena "ITest".

2. Pruebas de integración de la capa de presentación con la capa de negocio y la capa DAO. Diseñar y aplicar pruebas de integración para comprobar el funcionamiento conjunto de la clase `VistaGerente` con el resto de capas de la aplicación.
  - i. Diseñar casos de prueba para la funcionalidad implementada hasta el momento en la clase (correspondiente al caso de uso Consulta Tienda).
  - ii. Implementar en *JUnit* los casos de prueba diseñados utilizando *FEST*.
  - iii. Ejecutar los casos de prueba y corregir errores.
  - iv. Cuando todas las pruebas de caja negra sean correctas, comprobar la cobertura.



- v. Diseñar nuevos casos de prueba para alcanzar la cobertura deseada en caso de no haberse alcanzado con las pruebas de caja negra.

Nota: Para la realización de esta práctica, consultar el Anexo 1.

### 3. Criterios de Evaluación y Aclaraciones

Se deberá entregar un fichero comprimido que contenga los siguientes elementos:

- Proyectos Maven completos, con todas las clases involucradas (tanto de aplicación como de prueba), debidamente exportados.
- Un informe que resuma el proceso de pruebas aplicado, identificando las fases, las técnicas utilizadas y los casos de prueba diseñados. Se proporciona, a través de la plataforma *Moodle*, una plantilla para la elaboración de dicho informe.

Se valorará:

- a) El formato del informe.
- b) La corrección del proceso de pruebas de las clases Empleado y Tienda.
- c) La corrección del proceso de pruebas de la clase VistaGerente.

*Patricia López Martínez*  
*Juan Rivas Concepción*  
*Adolfo Garandal*

#### ANEXO 1. IMPLEMENTACION DE LA CAPA DE PERSISTENCIA

La capa de persistencia, capa DAO, está implementada usando H2, una base de datos implementada en Java que puede configurarse para ser usada como base de datos en memoria, siendo ese el modo en que se ha utilizado en esta aplicación. Esto implica que los datos solo existen durante el tiempo de vida de la aplicación.

El estado inicial de la base de datos cuando se lanza la aplicación, que es necesario conocer para la realización de la fase de verificación y validación, es el que se muestra en la Tabla 1.

Tabla 1. Estado inicial de la base de datos

Tiendas							
ID	Nombre	Dirección	Empleados				
			DNI	Nombre	Baja	Fecha	Categoría
1	Tienda A	Dirección A	11111111A	Juan Pérez	False	2002-01-15	ENCARGADO
			11111111B	María Rodríguez	True	2016-05-20	VENDEDOR
			11111111C	Luis Martínez	False	2022-05-21	AUXILIAR
2	Tienda B	Dirección B	11111111D	Lucía Ibáñez	False	2010-06-01	ENCARGADO
3	Tienda C	Dirección C					