

OBJETIVO

Desarrollar una aplicación en Java (u otro lenguaje de programación de propósito general) que permita gestionar la información de un objeto XXX realizando las operaciones de inserción, eliminación, actualización, búsqueda individual, búsqueda en lista (CRUD) y representación de un valor en una gráfica de línea (u otro tipo). **La aplicación debe tener acceso a una base de datos donde la información del objeto XXX reside de forma permanente.**

CASO DE ESTUDIO

Considere que requiere gestionar la información de un objeto de su elección. Para hacerlo, requiere que la aplicación cumpla con el patrón de diseño MVC (Modelo Vista Controlador) y la capacidad del software de “refrescar” la información cada vez que haya una actualización (patrón observador). Las operaciones básicas que se pueden realizar con el objeto están definidas en CRUD (Create, Read, Update, Delete) + representación en una gráfica, adicionalmente es necesario que la aplicación permita almacenar de forma persistente los datos en una base de datos. En la figura 1 se describen las funcionalidades de la aplicación.

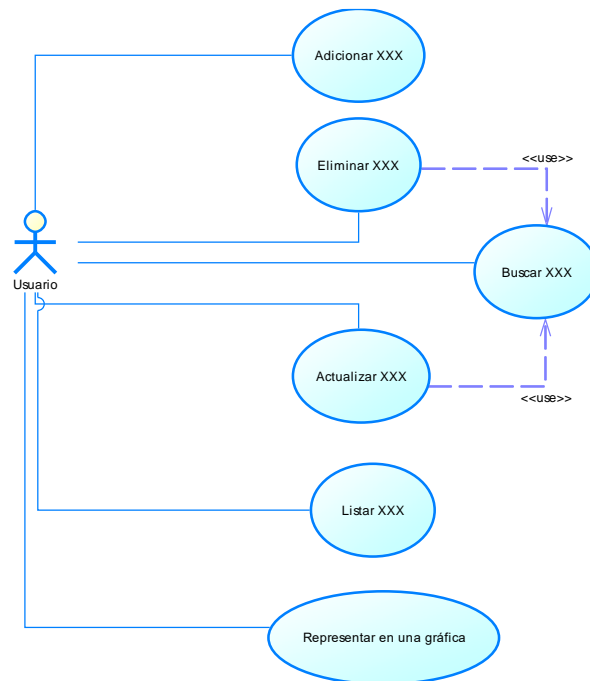


Figura 1. Diagrama de casos de uso para la aplicación XXX

ACTIVIDADES A REALIZAR

Desarrolle una aplicación en Java (u otro lenguaje de programación de propósito general) que permita gestionar un objeto XXX (CRUD + representación en una gráfica) utilizando el patrón de diseño MVC y permitiendo la actualización de las interfaces gráficas cuando un objeto sea **adicionado, modificado o eliminado** en otra ventana. La aplicación debe cumplir con las siguientes características:

1. Se debe implementar los patrones de diseño Singleton, MVC y observador.
2. Debe existir una clase estructural XXX que represente el objeto central estructural de la aplicación y objeto que será almacenado en una tabla de la BD.
3. La interfaz grafica debe cumplir con los requerimientos de usabilidad.
4. La aplicación debe actualizar la GUI de **búsqueda, listado y representación gráfica** en el caso de realizarse la modificación o eliminación del objeto XXX buscado.
5. La aplicación debe tener conexión a la BD donde reside la información del objeto XXX.

CONDICIONES PARA LA ENTREGA

1. La actividad será desarrollada de forma grupal (2). Cualquier taller que sea encontrado como copia de otro, será reportado al comité disciplinario según el reglamento estudiantil.
2. La implementación desarrollada deberá presentarse en la clase, en el tiempo estipulado para tal fin.
3. La calificación del aplicativo desarrollado se basará en la rúbrica especificada para tal fin.
4. La implementación deberá entregarse a través de la plataforma en el enlace dispuesto para tal fin.
5. La actividad tiene un valor proporcional correspondiente al primer taller del curso.

RÚBRICA DE CALIFICACIÓN

Ítem	Arquitectura	Implementación de PD	Insertar XXX	Eliminar XXX	Listar XXX	Consultar XXX	Actualizar XXX	Representar gráficamente XXX
Descripción	La aplicación debe cumplir con las especificaciones de diseño, las clases GUI (vistas), Model y estructurales deben existir así como la clase XXX en sus respectivos paquetes. Las operaciones deben ser coherentes con la naturaleza de la clase, por ej.: la clase GUI debe realizar procesos asociados a interfaz gráfica.	Los patrones de diseño Singleton, MVC y Observer deben estar implementados en la solución. La aplicación debe almacenar de forma permanente la información en una BD.	Operación que permite adicionar un nuevo objeto de la clase XXX de acuerdo a sus características (atributos).	Operación que permite eliminar un objeto previamente buscado por el usuario.	Operación que permite Listar todos los objetos en la Aplicación a través de una grilla. En el caso que la ventana de esta operación este abierta y el usuario ingrese, actualice o elimine un objeto, la lista debe refrescarse automáticamente.	Operación que permite consultar UN SOLO objeto mostrando sus características (atributos). En el caso que la ventana de esta operación este abierta y el usuario ingrese, actualice o elimine un objeto, la consulta debe refrescarse automáticamente.	Operación que permite actualizar un o varios atributos del objeto previamente buscado por el usuario.	Operación que permite representar gráficamente, utilizando una librería como JFreeChart, la información de uno o varios atributos del objeto XXX.
Peso	10%	REQUERIMIENTO	10%	10%	20%	20%	10%	20%

REQUERIMIENTO: En caso de NO cumplir con el requerimiento descrito, este producto NO será aceptado y tendrá una nota de 0.00