Actividad Grupal: Ejercicios sobre sistemas dinámicos

* **Objetivos** de la actividad: Diseñar y modelar clases en java así como aprender a utilizar el polimorfismo, la agregación y la composición
* **Descripción** de la actividad: Se plantea el siguiente ejercicio:

**EJERCICIO 1 (Diseño de clases I):**

Una empresa ha decidido informatizar el sistema de control de stock de sus productos. La empresa tiene productos físicos y productos virtuales (licencias de software) que se identifican por un ID único. Los productos físicos tienen unos costes de transporte asociados, y unas características físicas (peso, alto, ancho, largo) mientras que los virtuales no.

**Realizar el modelado de clases en UML y exportar las clases desde ArgoUML.**

**Se debe entregar:**

* Parte de la memoria correspondiente a esta sección en la que se explique en todo detalle el procedimiento seguido para diseñar las clases. Debe de aparecer también el diseño final de la clase en UML (captura de pantalla).
* Parte de la memoria con la explicación sobre cómo se ha hecho la exportación de clases a .java.
* Archivo con la clase en ArgoUML.

**EJERCICIO 2 (Diseño de clases II):**

Se desea informatizar el proceso de compra online de la empresa anterior.

Los usuarios pueden registrarse en la web, y realizar pedidos tanto de productos físicos como virtuales.

Un mismo usuario no podrá comprar más de una unidad de un producto virtual.

Un pedido se compone de una serie de líneas de pedido, y una dirección de envío y otra de facturación.

**Realizar el modelado de clases en UML y exportar las clases desde ArgoUML.**

**Se entrega:**

* Parte de la memoria correspondiente a esta sección en la que se explique en todo detalle el procedimiento seguido para diseñar las clases. Debe de aparecer también el diseño final de la clase en UML (captura de pantalla).
* Parte de la memoria con la explicación sobre cómo se ha hecho la exportación de clases a .java.
* Archivo con la clase en ArgoUML.

**EJERCICIO 3 (Diseño de clases III):**

Se requiere realizar un programa informático que gestione el almacén de la empresa anterior.

El programa debe tener 2 funcionalidades principales:

1- Introducir stock de producto. Se piden los datos de los mismos si procede: (id de artículo, tipo: virtual/físico, ancho, alto, largo, peso, unidades stock)

2- quitar stock de producto. Se ha realizado una venta y se necesita retirar unidades del almacén. Se pide id de artículo y cantidad a retirar (no puede ser mayor que el stock disponible)

**Realizar la implementación en Eclipse.**

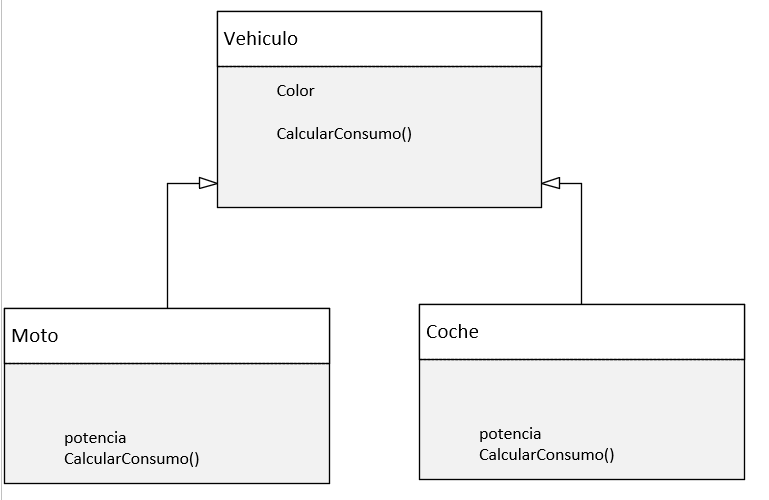
**A tener en cuenta:**

* La implementación de las clases debe de utilizar la herencia.
* Se debe de implementar un programa informático que pida por pantalla los datos y luego los muestre.

**Se entrega:**

* Parte de la memoria correspondiente a esta sección en la que se explique en todo detalle el procedimiento seguido para diseñar las clases en Eclipse (con el código comentado es suficiente). Se debe de explicar la estrategia seguida para crear el programa y añadir el código comentando para qué sirve cada parte.
* Archivos ejecutables con el proyecto exportado de Eclipse con todos los archivos asociados.

**EJERCICIO 4 (Utilización del polimorfismo):**

Se requiere realizar un programa informático que según el diagrama de clases adjunto haga uso del polimorfismo.

Nota: El cálculo del consumo se realiza como:

potencia \* nº de ruedas del vehículo \*0,12

**Realizar la implementación en Eclipse.**

**A tener en cuenta:**

* El programa puede pedir por pantalla los parámetros necesarios.
* El programa debe de mostrar por pantalla alguna figura.
* Se valorará especialmente el uso del polimorfismo.

**Se entrega:**

* Parte de la memoria correspondiente a esta sección en la que se explique en todo detalle el procedimiento seguido para diseñar las clases en Eclipse (con el código comentado es suficiente). Se debe de explicar cómo y dónde se ha utilizado el polimorfismo.
* Archivos ejecutables con el proyecto exportado de Eclipse con todos los archivos asociados.

**EJERCICIO 5 (Composición y agregación):**

Se pretende definir un sistema para dibujar figuras geométricas en JAVA, de distintos tipos como polígonos o figuras circulares por medio de sus vértices, centros y magnitudes necesarias para su representación. Se debe contar con tablero de figuras. En este se pueden representar distintas figuras que no puedan cambiar de posición pero si de color. Además se deben calcular sus áreas y perímetros o longitudes e imprimir el resultado para cada figura.

**Realizar el diseño en UML.**

**Se entrega:**

* Parte de la memoria correspondiente a esta sección en la que se explique en todo detalle el procedimiento seguido para diseñar las clases y dónde y porqué se utiliza composición y/o agregación. Debe de aparecer también el diseño final de la clases en UML (captura de pantalla)
* Realice la especificación de las clases haciendo hincapié en la definición de sus atributos, los prototipos de los métodos, mostrar los métodos polimórficos, si es que existen y la implementación de las relaciones de agregación y composición que se identifiquen. No es necesario implementar los métodos completamente y por tanto no se generará un ejecutable.
* **Rúbrica.** La rúbrica de corrección de la actividad es la siguiente:

|  |  |  |  |
| --- | --- | --- | --- |
| (valor real: 3 puntos) | Descripción | Puntuación máxima  (puntos) | Peso  % |
| Ejercicio 1 | Modelo UML y exportar clases | 2 | 20% |
| Ejercicio 2 | Modelo UML y exportar clases | 2 | 20% |
| Ejercicio 3 | Implementación de eclipse | 2 | 20% |
| Ejercicio 4 | Implementación de eclipse | 2 | 20% |
| Ejercicio 5 | Implementación UML | 2 | 20% |
|  |  | **10** | **100 %** |