

Tiempo estimado: 1,5 horas

Requisitos Iniciales del Dominio

Fragmento de los requisitos de la futura solución integral para **ECParcel**, pyme dedicada al *Transporte de Paquetería No Urgente*. Se trata de solucionar los problemas de *Control de Envíos* de esta pequeña empresa de transporte de paquetes que trabaja únicamente con destinos dentro de la Unión Europea. Un *destino* es una localidad que tiene un nombre, país, una distancia en Km. y un precio base (para cobrar a los *clientes*). Los Clientes de ECParcel suelen ser pequeñas empresas, de las que se debe controlar el CIF, los datos generales y el tipo de pago (inmediato o mensual). Un *envío* pertenece a un cliente, se debe entregar en un destino (en una dirección de entrega concreta) y posee una fecha de recepción, un nº de localizador, un estado (que puede ser 'almacenado', 'cargado', 'en tránsito' o 'entregado'; el estado inicial es 'almacenado'), un tipo de porte (portes debidos o pagados) y el importe a cobrar. Además, un envío está formado por un número de *paquetes*, cada uno de los cuales dispone de su propio nº de localizador y de cuatro medidas fundamentales para el transporte: su peso, alto, largo y ancho, con lo que se calcula el volumen en m³. Por otra parte, se consideran los paquetes *especiales* de los que, además de todas las propiedades anteriores, se necesita controlar el nivel de fragilidad, el tipo de contenido (líquido, electrónica, vidrio,...) y unas observaciones sobre el embalaje.

Esta pequeña empresa funciona subcontratando a *transportistas* independientes (nombre, nº de móvil y precio por viaje), que son dueños de pequeños *camiones* (capacidad en Kg., dimensiones de almacenaje en m³, fecha de matriculación, marca, modelo, matrícula). Un transportista sólo viaja a un destino y es dueño de un único camión (pero a un destino viajan varios transportistas).

Cuando el responsable de envíos planifica un viaje utiliza el concepto de *carga*. Una carga tiene una fecha de creación, una fecha prevista de viaje, un estado (inicialmente 'cargado', o 'en tránsito' o 'entregado'), un transportista y se vincula con el conjunto de paquetes que dicho transportista va a transportar (téngase en cuenta que en cuenta que cada paquete puede pertenecer a diferentes envíos, en fin, un envío puede viajar a su destino dividido en diferentes camionetas, véase más adelante).

Consideraciones de Diseño

1. **Cálculo del coste de un envío.** El importe del coste de un envío está formado por el precio base del destino más el sumatorio del importe del transporte de cada paquete. El cálculo del importe de un paquete (su signatura es *calcularImporte():importePaquete*) es el producto del peso por el volumen por un índice de tarifa (que es una constante). Si el paquete es especial, el cálculo del importe es el mismo pero se le añade un 10 %. Finalmente, se imprime un comprobante que indica el CIF del cliente, el nº de localizador del envío y el importe a pagar (esta fórmula puede cambiar en breve, por tanto es preciso modelar una solución flexible).

En resumen:

$$\text{ImporteEnvio} = \text{destino.precioBase} + \sum \text{ImportePaquete}$$

$$\text{ImportePaqueteNormal} = \text{peso} * \text{volumen} * \text{indicetarifa}$$

$$\text{ImportePaqueteEspecial} = (\text{peso} * \text{volumen} * \text{indicetarifa}) + 10\%$$

2. **Como planificar las entregas en un destino.** Nuestra solución software debe tener en cuenta que dado un conjunto de envío a entregar en un destino y un conjunto de camionetas, debe existir una manera optima de distribuir los envíos en camionetas minimizando el espacio utilizado. Una clase denominada *Colocador* se implementará en un futuro próximo, pero en este momento, hay que conectarse a un contrato: una operación cuya signatura es: *colocar(Paquete [], camioneta): Paquete[]*. [Para tu información, el método futuro recibirá la lista completa de paquetes y la camioneta, y el algoritmo seleccionará los paquetes que entran en la camioneta. Los paquetes que caben permanecerán en la lista original, mientras que los que no caben se extraen y se depositan en una nueva lista que la operación devuelve.]

Tienes que HACER

1. Produce el Modelo del Dominio usando la información proporcionada en la sección *Requisitos iniciales del Dominio*.
2. Produce un Diagrama de Clases de Diseño de Alto Nivel en UML que refleje las decisiones de diseño que has tomado. Completa y detalla el diagrama: todos los adornos UML.
3. Riel 5.10: “*If two or more classes have common data and behaviour (i.e. methods) then those classes should each inherit from a common base class which captures those data and methods*”. ¿Se puede aplicar en tu modelo? Revisa y justifica tu respuesta.
4. *Polimorfismo*. ¿Hay una situación polimórfica en tu dominio? Si es así, revisa que la has modelado correctamente e indica cuál es por medio de una nota UML.
5. Necesitas modelar el *índice de tarifa* que es un atributo particular. ¿Dónde lo pondrás? ¿Cómo lo modelarás? Revisa el modelo y actualiza.
6. Revisa el concepto de *Atributo Derivado*. ¿Puedes encontrar un ejemplo interesante de atributo derivado en este caso? Si es así, añádelo al modelo utilizando la notación UML apropiada.
7. *Colocador* es una clase que no existe todavía. Pero tienes el contrato de la clase con la que se comunica. ¿Cómo puedes modelar el contrato en UML? Revisa el modelo y actualiza.
8. Supongamos que cambian los requisitos, de tal manera que cada transportista puede hacer distintos destinos y que el precio de cada transportista depende del destino ¿cuál es la forma más precisa de representar la nueva situación? Representa gráficamente la solución.