



CHANGELOG

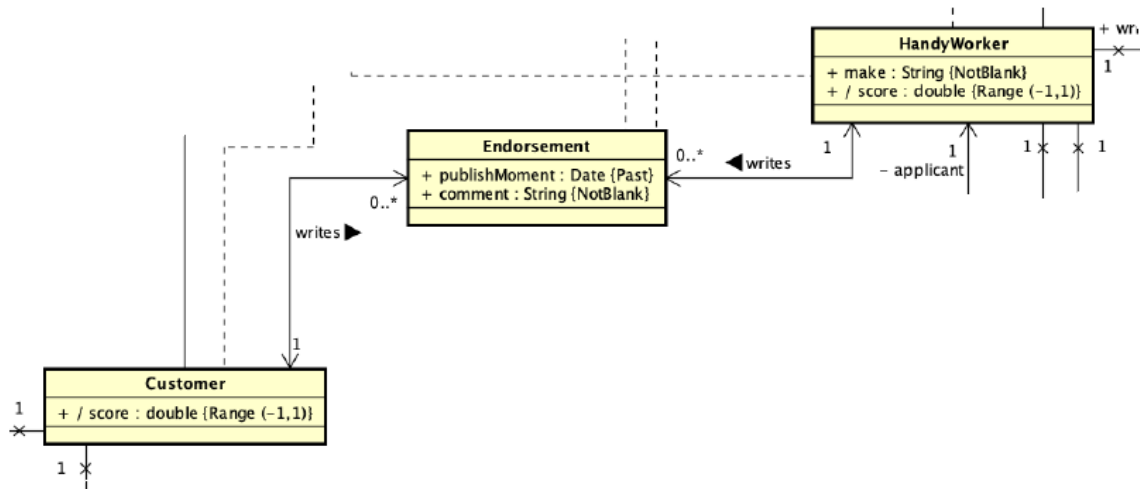
29/11/2018

Registro de cambios

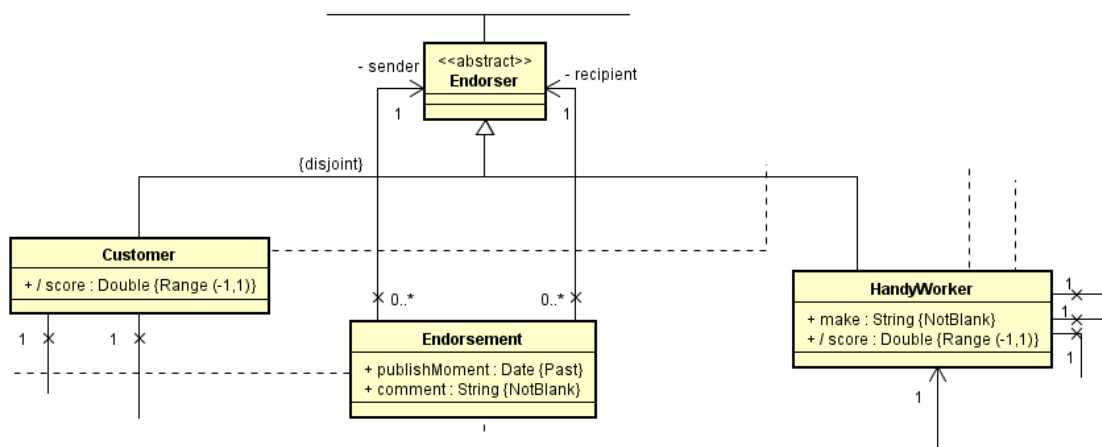
Registro de cambios respecto a la versión anterior del proyecto.

1. CAMBIOS EN LOS MODELOS DE DOMINIO

- ❖ La relación “*HandyWorker* → *Endorsement* ← *Customer*” se ha cambiado ya que ésta no representa bien el requisito 46. Se ha creado una nueva clase abstracta para identificar con facilidad quién recomienda a quién. Esta clase se llama *Endorser*. A continuación, se muestra como estaba representado este requisito en la versión anterior del proyecto:



La versión del entregable D04:



- ❖ Se ha cambiado la relación “*MessageBox → Message*” a bidireccional porque de esta manera facilitamos la obtención de datos desde cualquier extremo de la relación.
- ❖ La relación entre *Actor* y todos sus descendientes ha pasado de ser {*overlapping*} a {*disjoint*}. De esta manera limitamos que un actor tenga más de una autoridad.
- ❖ Se ha cambiado la relación “*Sponsor → Sponsorship*” a bidireccional porque es necesario conocer el dueño de los *Sponsorship* y qué *Sponsorships* tiene cada *Sponsor*.
- ❖ En la relación “*Actor → Message*” se ha cambiado el nombre del rol de *recipient* a *recipients*. Cambio realizado tanto en el modelo conceptual como el de dominio para usar la denominación utilizada en el documento de requisitos (requisito nº2).
- ❖ Tanto la multiplicidad de la relación “*Application → CreditCard*” como la de “*Sponsorship → CreditCard*” se ha cambiado de 1 → 0..1 a 1..* → 0..1, porque en varias *Application/Sponsorship* se puede utilizar la misma *CreditCard*.
- ❖ La clase *Money* se ha eliminado ya que su implementación se vuelve compleja al tratarse de un *datatype*. Esto se debe a que los atributos de ésta son obligatorios, por lo que al tratar de referenciarlos en una clase en la que *CreditCard* es opcional surgen problemas.
- ❖ La direccionalidad de la relación “*Sponsorship → Tutorial*” se ha invertido y su multiplicidad se ha cambiado de 1 → 0..* a 1..* → 0..* ya que, en vistas, cuando en un tutorial se elige aleatoriamente un *sponsorship*, necesitamos que éstos estén en distintos tutoriales.
- ❖ Cambiada las relaciones “*Complaint → Report*” a “*Complaint ← Report*” y “*Report → Note*” a “*Report ↔ Note*”.
- ❖ Todos los tipos de los atributos se han cambiado de primitivos a objetos en los modelos. En Java, el programador decide qué es lo más conveniente.
- ❖ Cambios menores de etiquetas en Java para ajustarlo a los requisitos y los recientes servicios.
- ❖ Cambio en *HandyWorker* en su atributo *Finder*. Se ha puesto opcional debido a que consideramos que un *HandyWorker* puede o no realizar una búsqueda.