

Item 1

Cambios en los modelos conceptual y de dominio

1. En la relación actor->message se ha cambiado el nombre del rol de receiver a recipient. Cambio realizado tanto en el modelo conceptual como el de dominio para usar la denominación utilizada en el documento de requisitos (requisito nº2).
2. Únicamente en el modelo de dominio se ha eliminado la entidad workPlan y cambiado el nombre a la relación por "compoundBy"; ambos cambios se han realizado por recomendación del profesor tras concluir que tener una entidad completamente vacía no tenía sentido.
3. Se ha aclarado el patrón del nº de teléfono en actor y añadido la anotación de que se le debe pedir confirmación al administrador.
4. Se ha cambiado el nombre del atributo linkAttachment a attachment en la entidad PersonalRecord por motivos estéticos.
5. Añadida nota a la entidad Finder para el nº de resultados (requisito 41). En la presente revisión nos hemos percatado de que faltaba esta anotación.
6. Únicamente en el modelo de dominio la direccionalidad de la relación HandyWorker<->Tutorial se cambió a HandyWorker -> Tutorial ya que se ha llegado a la conclusión de que no hacía falta una relación bidireccional; no siempre que veas un tutorial vas a querer saber el manitas que lo escribió.
7. Se ha añadido el atributo number a la entidad Section, el cual no existía debido a un error en la comprensión del requisito pertinente (nº43) en el entregable anterior.
8. Tal y como se muestra en el material suministrado por el profesorado en esta entrega, los atributos de useraccount y authority deben llevar un "+" en vez de un "-".
9. Se elimina la restricción {NotBlank} del atributo comment en la entidad Application ya que se encontraba junto a [0..1].
10. Se ha eliminado la 'd' en el atributo lastUpdated en la entidad Tutorial para usar el nombre suministrado en el documento de requisitos.
11. Por directriz del profesorado se ha utilizado el tipo envoltura Boolean en vez del tipo primitivo.
12. En la entidad FixUpTask el atributo Date startMoment, debe llevar restricción {Future} tal y como se indica en el documento de requisitos. Tras hablar con el cliente, se ha decidido no poner un periodo mínimo de tiempo entre la fecha de creación de la FixUpTask y la fecha de comienzo de la misma; sólo se requiere que sea una fecha futura.
13. Por recomendación del profesorado con el fin de una implementación más sencilla, los datatype que anteriormente eran enum (Status y Priority) se han y puesto como atributo de tipo String con un pattern en las clases que usasen dichos datatypes. Este pattern restringe a los valores que podía tomar el enum.
14. Con el fin de utilizar el mismo vocabulario del documento de requisitos, se ha cambiado de nombre la relación FixUpTask -> Category; pasa de areOrganisedIn a belongsTo.
15. Se ha cambiado el datatype CreditCard a entidad, eliminando el atributo de tipo CreditCard de las clases que tuviesen relación con el datatype y uniendo dichas clases a la nueva entidad. Dicho cambio se ha realizado ya que el requisito 24 indica que el sistema debe almacenar la información de las tarjetas de crédito.

16. Se han cambiado los atributos de tipo String que quedaban como [0..*]. Tal y como se ha explicado en clase, se puede poner como solo un String [0..1]. A su vez, a aquellos String con [1..*] se les ha añadido la restricción {NotBlank}.
17. Añadida en ambos modelos la restricción {Unique} al atributo userName en la entidad userAccount ya que aparecía en la clase de java proporcionada pero no se encontró la discrepancia en los chequeos anteriores.
18. Añadida bidireccionalidad a la relación Application<->FixUpTask tras el estudio de direccionalidades.
19. Modificada la multiplicidad en la relación Message->Actor (+recipient) de 0..*->1 a 0..*->1..* puesto que un mensaje puede ser enviado a más de un destinatario.