Diseño y Pruebas II

ANALYSIS REPORT

Alumno: Julio Navarro Rodríguez
Repositorio: https://github.com/DP2-C1-02-07/Acme-L3-D04

1. Índice

2. Tabla de versiones

Versión	Fecha	Descripción
1.0.0	20/06/2023	Documento creado

3. Resumen del documento

Mediante este documento se desarrollará un análisis de los requisitos más complejos incluidos en este entregable, en el que se incluirá cada requisito citado textualmente y las decisiones tomadas para cumplimentar cada requisito.

4. Introducción

Este documento incluye únicamente requisitos de información, ya que los demás tipos de requisitos para este entregable no se han considerado merecedores de un análisis más elaborado.

5. Contenidos

5.1. Requisitos de información

REQUISITO 5): A practicum helps put hands-on courses into practice in the context of a real company. The system must store the following data about them: a code (pattern “[A-Z]{1,3}[0-9][0-9]{3}”, not blank, unique), a title (not blank, shorter than 76 characters), an abstract (not blank, shorter than 101 characters), some goals (not blank, shorter than 101 characters), and an estimated total time (in hours, computed from the corresponding sessions plus/minus 10%).

CONTEXTO: Este requisito de información nos exige crear una nueva entidad para la aplicación la cual guarde información acerca de prácticas sobre algún curso. Uno de sus atributos más interesantes es el que llamaré “estimatedTime”, el cual se encuentra subrayado en el enunciado.

La clave de este atributo es la elección del tipo de dato que va a almacenar ese tiempo, para el cual existen dos alternativas:

Alternativa 1: “estimatedTime” es un atributo derivado de tipo int o Integer.

Pros:

- Formato más simple a la hora de representar el atributo.

Contras:

- No es tan preciso y se puede llegar a perder información.
- Podría no contemplar el 10% de margen de error para duraciones cortas.

Alternativa 2: “estimatedTime” es un atributo derivado de tipo double.

Pros:

- Gran precisión a la hora de mostrar la información.
- La pérdida de información puede ser insignificante según los decimales a tener en cuenta.

Contras:

- No es tan preciso y se puede llegar a perder información.
- Para ciertos usuarios, puede dar lugar a confusión la parte decimal del número, ya que si se toman sólo dos decimales se pueden confundir con minutos en lugar de con la fracción de una hora.

DECISIÓN TOMADA: Tras consultar en el foro de la asignatura, vi que varios compañeros habían consultado el mismo dilema para atributos muy similares. Una vez leídas las publicaciones de los compañeros y las correspondientes respuestas de los profesores, llegué a la conclusión de qué debía optar por la alternativa 2.

FUENTE: **Foro de la asignatura**

REQUISITO 6): The system must store the following data about the sessions in a practicum: a title (not blank, shorter than 76 characters), an abstract (not blank, shorter than 101 characters), a time period (at least one week ahead, at least one week long), and an optional link with further information.

ANÁLISIS: Este requisito de información también cuenta con un atributo "abstract", por lo que se aplicará la misma estrategia que en el requisito anterior. Además, tiene un atributo "time period" con varias restricciones problemáticas si usáramos el tipo Duration de Java. Por ello, dividiremos el atributo en dos: "startDate" y "finishDate", para validar las restricciones en los servicios cuando se implementen las features.

CONTEXTO: Este requisito de información nos exige crear una nueva entidad para la aplicación la cual guarde información acerca de las distintas sesiones que se realizarán en una práctica.

El foco del análisis se encuentra en cómo representar la duración de dichas sesiones, para lo que se proponen dos alternativas:

Alternativa 1: "timePeriod" es un atributo de tipo Duration que muestra la duración del período.

Pros:

- Es la implementación más lógica que se nos viene a la mente cuando tenemos que implementar períodos de tiempo.

Contras:

- Gran dificultad a la hora de utilizar el tipo Duration dentro de Acme-Framework.
- Es más costoso representarlo de manera amigable al usuario.

Alternativa 2: Dividir el atributo en 2, “startDate” y “finishDate” de tipo Date para representar así el período de tiempo.

Pros:

- El tipo Date es cómodo de utilizar en Acme-Framework.
- Contamos con ejemplos de restricciones del tipo Date en los proyectos de ejemplo.
- Aplicar restricciones es más sencillo.

Contras:

- Se almacena mayor cantidad de información en la base de datos.
- No cumple al pie de la letra las exigencias del cliente, ya que se estarían implementando dos atributos en lugar de uno.
- Incluye una restricción adicional: la fecha de fin debe ser posterior a la fecha de inicio.

DECISIÓN TOMADA: Nuevamente, luego de consultar en el foro de la asignatura, vi que varios compañeros habían consultado el mismo dilema para atributos muy similares. Una vez leídas las publicaciones de los compañeros y las correspondientes respuestas de los profesores, llegué a la conclusión de qué debía optar por la alternativa 2.

FUENTE: **Foro de la asignatura**

5.2. Requisitos funcionales

No ha sido necesario realizar un análisis de los requisitos funcionales debido a que este entregable no incorporaba requisitos de este tipo.

5.3. Requisitos no funcionales

No ha sido necesario realizar un análisis de los requisitos no funcionales debido a que este entregable no incorporaba requisitos de este tipo.

5.4. Requisitos de testing

No ha sido necesario realizar un análisis de los requisitos de testing debido a que este entregable no incorporaba requisitos de este tipo dignos de un análisis en profundidad.

5.5. Requisitos de gestión

No ha sido necesario realizar un análisis de los requisitos de gestión debido a que este entregable no incorporaba requisitos de este tipo dignos de un análisis en profundidad.

6. Conclusiones

De nuevo, este entregable incorpora pocos requisitos aptos para un análisis en profundidad, ya sea por escasa complejidad (como el requisito 4 que quedó fuera de este documento) o porque se ha adquirido el conocimiento necesario en sprints anteriores para cumplimentar los requisitos (documentos de planning y analysis).

7. Bibliografía

- ”Annexes.docm”, proporcionado por el profesorado para la elaboración de documentos.