



Analysis Report

C3.017

<https://github.com/vicgrabru/Acme-SF-D04>

Francisco de Asís Rosso Ramírez
frarosram@alum.us.es

Tabla de contenidos

Tabla de contenidos	2
Resumen ejecutivo	3
Tabla de versiones	3
Introducción	3
Análisis realizados durante el sprint 1	4
Análisis realizados durante el sprint 2	5
Análisis 2.1	5
Análisis 2.2	6
Análisis realizados durante el sprint 3	7
Análisis 3.1	7
Análisis 3.2	7
Análisis realizados durante el sprint 4	9
Análisis 4.1	9
Análisis 4.2	10
Conclusión	11
Bibliografía	11

Resumen ejecutivo

En el presente informe se listan los análisis realizados a requisitos individuales a completar por el alumno 5, si procede, en cuyo caso se proporciona también la conclusión detallada de dicho análisis y la validación de un profesor en caso de haber sido aprobado en el foro de la asignatura.

Tabla de versiones

Versión	Fecha	Descripción
1.0	15/02/2024	Primera versión del documento
2.0	08/03/2024	Segundo sprint
3.0	26/04/2024	Tercer sprint
4.0	27/05/2024	Cuarto sprint
5.0	15/10/2024	Revisión para la tercera convocatoria

Introducción

Este documento es referente a los requisitos individuales del alumno 5.

El objetivo de este documento es ofrecer una visión detallada de los problemas identificados a la hora de interpretar y enfrentar los requisitos así como las decisiones tomadas al respecto y las conclusiones obtenidas.

Por cada caso, se presentará el requisito en cuestión y el análisis realizado, del que se aportarán las conclusiones detalladas, decisiones tomadas y enlace a la validación de los profesores en el foro, si procede.

Se incluyen los análisis de todas las versiones anteriores.

Análisis realizados durante el sprint 1

No fue necesario analizar ningún requisito durante el entregable.

Análisis realizados durante el sprint 2

Análisis 2.1

Requisito analizado

2) **Code audits** are essential pieces to ensure the quality of a **project**. The system must store the following data about them: a **code** (pattern “[A-Z]{1,3}-[0-9]{3}”, not blank, unique), an **execution** date (in the past), a **type** (“Static”, “Dynamic”), a list of proposed **corrective actions** (not blank, shorter than 101 characters), a **mark** (computed as the mode of the marks in the corresponding auditing records; ties must be broken arbitrarily if necessary), and an **optional link** with further information.

Análisis realizado

En una primera versión del requisito, se decidió efectuar el cálculo de Mark en la entidad Audit como un atributo transient.

Tras consultar a la profesora, se ha decidido en su lugar tratar Mark como un atributo persistente en la base de datos y posponer su cálculo a cuando se desarrolle el servicio mediante una consulta.

Desventajas

- Aún no calculamos Mark para Audit.

Ventajas

- Mark persiste en la base de datos.
- La consulta para el nuevo cálculo puede llevarse a cabo cuando la base de datos ya esté completa.

Análisis 2.2

Requisito analizado

3) The result of each code audit is based on the analysis of their audit records. The system must store the following data about them: a code (pattern "AU-[0-9]{4}-[0-9]{3}", not blank, unique), the period during which the subject was audited (in the past, at least one hour long), a mark ("A+", "A", "B", "C", "F", or "F-"), and an optional link with further information.

Análisis realizado

No ha parecido posible crear el enumerado Mark tal como se pedía, ya que provocaba un error al introducir los caracteres "+" y "-". Por tanto, se ha optado por sustituir "A+" y "F-" por "APlus" y "FMinus".

Desventajas

- Los valores no son exactamente iguales a los pedidos.

Ventajas

- Es igualmente comprensible de un modo funcional.

Análisis realizados durante el sprint 3

Análisis 3.1

Requisito analizado

1) Operations by **auditors** on **code audits**:

- List the **code audits** that they have created.
- Show the details of their **code audits**.

Create, update, or delete their **code audits**. **Code audits** can be updated or deleted as long as they have not been published. For a **code audit** to be published, the mark must be, at least, "C".

Análisis realizado

Originalmente, el enumerado Mark solo tenía como opciones las notas requeridas (APlus, A, B, C, F y FMinus).

Tras analizar este requisito y ver que existe un punto tras crear un code audit en el que este no tiene audit records a partir de los que calcular la moda de las calificaciones para tener la suya, se ha decidido añadir un valor más de Mark, *None*, para asignar a los code audits en el momento de crearlos.

Desventajas

- Se añade a Mark un valor más que los pedidos.

Ventajas

- Se puede crear un code audit con un valor real provisional de Mark.

Análisis 3.2

Requisito analizado

8) Operations by **auditors** on **auditor** dashboards:

- Show their **auditor** dashboards.

Análisis realizado

El requisito no aclara si los audits dashboards deberían publicarse con todos los datos de code audits o solo los publicados.

Tras analizar el requisito se ha llegado a la conclusión de que tiene más sentido hacerlo solo con los datos de los code audits y audit records ya publicados.

Desventajas

- El código resulta ligeramente más complicado (apenas es una desventaja real).

Ventajas

- Tiene más sentido de cara a un entorno real calcular las estadísticas sobre datos definitivos que sobre borradores.
- Es más consistente con el resto del código (la nota de los code audits está calculada a partir de la de los audit records publicados).

Respuesta de un profesor

https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id= 85092_1 &nav=discussion_board&conf_id= 405265_1&forum_id= 234042_1&message_id= 415517_1

Afortunadamente esta duda ya estaba preguntada en el foro de la asignatura y, aunque se refiriera a los requisitos de otro alumno, es aplicable a este caso.

Conclusión

La respuesta del profesor corrobora que la opción de utilizar solo las entidades publicadas sea probablemente la más correcta.

Análisis realizados durante el sprint 4

Análisis 4.1

Requisito analizado

8) Operations by **auditors** on **auditor** dashboards:

- Show their **auditor** dashboards.

Análisis realizado

Contexto

En los dashboards de auditor, entre otras cosas, se incluyen el máximo, mínimo, promedio y desviación de los AuditRecords. AuditRecord tiene un periodStart y un PeriodEnd pero una query JPQL en el repositorio no puede realizar la diferencia necesaria para calcular la longitud de periodo entre el inicio y el final

Cuestión

Es necesario determinar una forma de calcular la longitud del periodo a partir de los timestamps Date periodEnd y periodStart.

Alternativa 1

Explicación: Implementar en el repositorio las funciones en varias partes, de modo que una consulta obtiene la colección de periodStarts y periodEnds y en segundo lugar, para cada valor buscado (promedio, máximo, mínimo y desviación) una función recibe la colección como parámetro, calcula la diferencia de tiempos y realiza la función estadística en cuestión.

- **Pros:** Facilita el cálculo de las funciones estadísticas al hacerse en su propia función cada una.
- **Contras:** Las funciones estadísticas no forman parte de la query.

Alternativa 2

Explicación: Añadir a AuditRecord un atributo derivado periodLength que devuelva la diferencia entre periodStart y periodEnd.

- **Pros:** No son necesarias funciones adicionales y las operaciones estadísticas se realizan en las queries.
- **Contras:** Se añade a AuditRecord un atributo que no se pedía y que solo tiene una función específica.

Respuesta de un profesor

https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id= 85092_1 &nav=discussion_board&conf_id= 405265_1&forum_id= 234042_1&message_id= 421357_1

Conclusión

Se tomará la alternativa 3 propuesta por Rafael Corchuelo; esto es, calcular el intervalo en la propia consulta realizándola de forma nativa SQL.

Análisis 4.2

Requisito analizado

9) Produce a test suite for Requirements #6 and #7.

Análisis realizado

A partir del Excel Sample-Data.xlsx y palabras del profesor Rafael Corchuelo, hemos llegado a las siguientes conclusiones:

- Para fechas:
 - *01/01/2000 0:00:00* es teóricamente el momento mínimo, aunque se puedan introducir fechas anteriores.
 - *30/07/2022 0:00:00* es el momento presente, aunque pueda usarse en fechas con restricción @Past.
- Para enlaces:
 - *ftp://* es un enlace mal formado aunque se acepte como válido en los campos @Url.

Dada la situación, hemos considerado oportuno respetar los límites teóricos de los atributos a la hora de generar sample-data y realizar los tests.

Desventajas

- No se prueban todos los límites prácticos de cada valor.

Ventajas

- Los sample-data y los tests son consistentes con los valores que deberían tener.

Conclusión

Con los requisitos analizados a lo largo del proyecto y las respuestas obtenidas por profesores, se puede observar que puede llegar a ser conveniente adaptar las condiciones pedidas a las necesidades del proyecto más que mantenerse en la literalidad de los requisitos. Es importante la iniciativa para llevar el proyecto adelante.

Bibliografía

Intencionalmente en blanco.