

Grupo: C1.015

Repositorio: <https://github.com/jormunrod/Acme-SF-24.1.1>

Analysis Report

Alejandro Pérez Santiago (alepersan3@alum.us.es)

Sevilla, 8-3-2024

Contenido

1.	Resumen ejecutivo	2
2.	Tabla de revisión.....	3
3.	Introducción	4
4.	Contenidos	5
4.1.	Requisito de información: Implementación de la entidad “CodeAudit”	5
4.1.1.	Descripción del requisito.....	5
4.1.2.	Análisis detallado	5
4.1.3.	Decisiones tomadas	5
4.1.4.	Enlace de validación	5
4.2.	Requisito de información: Implementación de la entidad “AuditRecord”	6
4.2.1.	Descripción del requisito.....	6
4.2.2.	Análisis detallado	6
4.2.3.	Decisiones tomadas	6
4.2.4.	Enlace de validación	6
5.	Conclusiones.....	7
6.	Bibliografía.....	8

1. Resumen ejecutivo

En el marco de este informe de análisis, se examinan los requisitos de información para la implementación de las entidades "CodeAudit" y "AuditRecord". Ambas entidades son cruciales para garantizar la calidad y seguridad del proyecto en cuestión. Se detallan los requisitos específicos para cada una de ellas, así como las decisiones tomadas para abordarlos de manera efectiva.

2. Tabla de revisión

Nº Revisión	Fecha	Descripción
1	08/03/2024	Versión inicial

3. Introducción

El presente informe tiene como objetivo analizar y proponer soluciones para los requisitos de información relacionados con las entidades "CodeAudit" y "AuditRecord" en el contexto de un proyecto específico. Se abordan desafíos particulares, como la implementación del atributo "mark" en la entidad "CodeAudit" y la definición de la duración en la entidad "AuditRecord". Se exploran diversas alternativas y se proponen recomendaciones fundamentadas en buenas prácticas y requisitos específicos del proyecto.

4. Contenidos

4.1. Requisito de información: Implementación de la entidad "CodeAudit"

4.1.1. Descripción del requisito

Los análisis de código son piezas esenciales para garantizar la calidad de un proyecto. El sistema debe almacenar los siguientes datos sobre ellos: un código (patrón "[A-Z]{1,3}-[0-9]{3}", no en blanco, único), una fecha de ejecución (en el pasado), un tipo ("Estático", "Dinámico"), una lista de acciones correctivas propuestas (no en blanco, más corta que 101 caracteres), una calificación (calculada como la moda de las calificaciones en los registros de auditoría correspondientes; los empates deben resolverse arbitrariamente si es necesario) y un enlace opcional con información adicional.

4.1.2. Análisis detallado

Se plantea la necesidad de implementar el atributo "mark" dentro de la entidad de Code Audit. Sin embargo, surge la dificultad de que este atributo parece derivarse de otra entidad y requiere una validación compleja debido a la solicitud de un desempate.

En la primera alternativa propuesta, se sugiere computar "mark" a través de una asociación @OneToMany hacia la entidad de records junto con un método customizado. Esto permitiría el cálculo del atributo, pero presenta el inconveniente de que la asociación @OneToMany no se considera adecuada según los criterios de la asignatura. Además, la recomendación (@ManyToOne) no sería aplicable en esta entidad debido a la naturaleza de un "Code Audit", que se basa en el análisis de múltiples registros. También implicaría una complejidad adicional en la entidad.

La segunda alternativa propone implementar la lógica para calcular el atributo en la capa de servicio de la aplicación. Esta opción elimina la necesidad de implementar la asociación @OneToMany de la alternativa uno, permitiendo el cálculo del atributo de manera similar. Además, promueve la división de responsabilidades. Sin embargo, dada la falta de abordaje de esta casuística en nuestras clases teóricas, sería necesario posponer la implementación de este requisito.

4.1.3. Decisiones tomadas

La solución propuesta sugiere optar por la segunda alternativa, ya que se alinea mejor con las buenas prácticas enseñadas en la teoría de la asignatura. Por lo tanto, si se debe tomar una decisión, se recomienda seguir esta opción.

4.1.4. Enlace de validación

Nótese que la consulta realizada al cliente a través del foro se ha realizado por parte de otros alumnos que no forman parte del grupo de trabajo, pero se ha tenido en cuenta a la hora de la implementación de los requisitos de información solicitados:

Información relacionada con la marca de la entidad "CodeAudit" [link](#).

4.2. Requisito de información: Implementación de la entidad "AuditRecord"

4.2.1. Descripción del requisito

El resultado de cada auditoría de código se basa en el análisis de sus registros de auditoría. El sistema debe almacenar los siguientes datos sobre ellos: un código (patrón "AU-[0-9]{4}-[0-9]{3}", no en blanco, único), el período durante el cual el sujeto fue auditado (en el pasado, de al menos una hora de duración), una calificación ("A+", "A", "B", "C", "F" o "F-"), y un enlace opcional con información adicional.

4.2.2. Análisis detallado

En el contexto de esta incidencia, estoy evaluando cómo implementar la duración requerida para la entidad. Inicialmente, considero que podría ser un valor numérico que indique cuánto tiempo se asigna a dicho objetivo, pero me doy cuenta de que esta duración debe comenzar después del momento de instanciación.

En mi primera alternativa, propongo implementar el atributo de duración con tipo Date. Esto me permite compararlo con el atributo de momento de instanciación para verificar que es posterior, y no requiere la creación de un tipo de dato personalizado. Sin embargo, me pregunto si esta opción concuerda con el significado de la palabra "duración".

En mi segunda alternativa, sugiero utilizar un tipo numérico para representar la duración. Esto se alinea más con el significado de la palabra "duración" como una cantidad de tiempo, pero no permite comprobar la restricción de que sea posterior al momento de instanciación.

En mi tercera alternativa, propongo utilizar un tipo de dato personalizado que tenga como atributos un momento de finalización y una diferencia en horas entre el momento de instanciación y el de finalización. Esto cumple con el significado de la palabra "duración" y permite la comparación con el momento de instanciación, pero aumenta la complejidad de la solución al requerir la creación de un tipo de dato personalizado.

4.2.3. Decisiones tomadas

Finalmente, adopto la recomendación del profesor. Sugiere implementar el "execution period" como dos atributos de tipo "Date" con un conjunto de restricciones apropiadas: ninguno de los dos puede ser nulo, el momento de inicio debe ser posterior al de creación, el de inicio debe ser anterior al de finalización, y se establece una duración mínima de una hora, según lo acordado con el cliente.

4.2.4. Enlace de validación

Nótese que la consulta realizada al cliente a través del foro se ha realizado por parte de otros alumnos que no forman parte del grupo de trabajo, pero se ha tenido en cuenta a la hora de la implementación de los requisitos de información solicitados:

Información relacionada con la duración de la entidad "AuditRecord" [link](#).

5. Conclusiones

En conclusión, este análisis detallado de los requisitos para las entidades "CodeAudit" y "AuditRecord" resalta la importancia de considerar cuidadosamente las alternativas disponibles y los desafíos específicos de implementación. Nuestras decisiones se basaron en un equilibrio entre las buenas prácticas y las necesidades del proyecto, tomando en cuenta la retroalimentación del cliente. Esto destaca la complejidad inherente en la implementación de requisitos de información y la necesidad de abordarlos de manera metódica y reflexiva para garantizar una implementación efectiva y satisfactoria del proyecto.

6. Bibliografía

[Foro: Tutorials](#)