

Analysis report



Diseño y pruebas II

Sprint 2

Versión 1.0

Fecha 06/03/2024

Preparado por:
Fernando José de Celis Hurtado

Índice

1.) Introducción	3
2.) Contenido	3
2.1) Implementación entidad code audit (Information requirement 2)	3
2.2) Implementación entidad audit record (Information requirement 3)	4
2.3) Implementación auditor dashboard (Information requirement 4)	4
2.4) Implementación del testing individual de la aplicación (Information requirement 5)	4
2.5) Implementación rol auditor (Information requirement 13)	5
2.6) Hacer el UML (Information requirement 14)	5
2.7) Hacer el analysis report (Information requirement 15)	5
2.8) Hacer el planning and progress report (Information requirement 16)	5
3.) Conclusiones	5

1.) Introducción

Un análisis detallado resulta esencial para el progreso y la constante optimización. La elaboración de un informe de análisis desempeña un papel fundamental en este proceso, ofreciendo una perspectiva minuciosa sobre los requisitos del sistema. Este documento no solo evalúa el grado de cumplimiento de dichos requisitos, sino que también destaca áreas susceptibles de mejora, proponiendo soluciones que buscan potenciar la eficiencia y funcionalidad del sistema.

El propósito fundamental de este informe consiste en ofrecer una guía clara y estructurada para el perfeccionamiento del sistema. Se busca no solo asegurar el cumplimiento de los requisitos establecidos, sino también alcanzar los más altos estándares de calidad en cada uno de ellos.

2.) Contenido

Para poder hacer este requisito correctamente, se ha tenido que crear otra clase enumerado `CodeAuditType` para poder hacer uno de los atributos de la entidad `CodeAudit`.

2.1) Implementación entidad code audit (Information requirement 2)

Descripción: 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.

Este requisito nos pide almacenar en la base de datos la información correspondiente de los code audits, los cuales tienen ciertas características y deben satisfacer una serie de restricciones. Para cumplir con este requisito, se ha creado la entidad `CodeAudit`. Para ello se creará un paquete dentro de entidades llamado `codeAudits`, y dentro de este una clase `CodeAudit`.

Para poder hacer este requisito correctamente, se ha tenido que crear otra clase enumerado `CodeAuditType` para poder hacer uno de los atributos de la entidad `CodeAudit`.

Uno de los atributos de la clase (`Mark`) debe ser calculado a partir de otro atributo de otra clase que se explicará posteriormente (atributo `Mark` de la entidad `AuditRecord`), pero lastimosamente con los conocimientos adquiridos hasta ahora en la asignatura no se puede calcular, por lo que se hará en la siguiente entrega.

2.2) Implementación entidad audit record (Information requirement 3)

Descripción: 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.

Este requisito nos pide almacenar en la base de datos la información correspondiente de los audit records, los cuales tienen ciertas características y deben satisfacer una serie de restricciones. Para cumplir con este requisito, se ha creado la entidad AuditRecord. Para ello se creará un paquete dentro de entities llamado auditRecords, y dentro de este una clase AuditRecord.

Para poder hacer este requisito correctamente, se ha tenido que crear otra clase enumerado Mark para poder hacer uno de los atributos de la entidad AuditRecord.

En esta entidad, el atributo auditEndTime debe ser superior al atributo auditStartTime como mínimo en una hora. Esta restricción tampoco se puede hacer en esta entrega por lo que se realizará en el siguiente sprint cuando se adquieran los conocimientos.

2.3) Implementación auditor dashboard (Information requirement 4)

Descripción: The system must handle auditor dashboards with the following data: total number of code audits for "Static" and "Dynamic" types; average, deviation, minimum, and maximum number of audit records in their audits; average, deviation, minimum, and maximum time of the period lengths in their audit records.

Este requisito nos pide almacenar una serie de información relacionada con los auditors, que se irá actualizando a medida que se vayan añadiendo code audits y audit records. Para ello se creará una clase dentro de forms llamada AuditorDashboard.

2.4) Implementación del testing individual de la aplicación (Information requirement 5)

Descripción: Produce assorted sample data to test your application informally. The data must include two auditor accounts with credentials "auditor1/auditor1" and "auditor2/auditor2".

Este requisito nos indica la necesidad de generar datos para poder realizar un testing informal sobre nuestra aplicación. Para cumplir con este requisito, se realizarán datos de prueba siguiendo las buenas prácticas de la asignatura. Se añadirán datos del rol auditor, de la entidad CodeAudit y de la entidad AuditRecord.

En este requisito surgieron diversos problemas puesto que la estructura de los csv son muy delicadas y si no haces a la perfección las tablas de datos, te salta error.

2.5) Implementación rol auditor (Information requirement 13)

Descripción: Produce assorted sample data to test your application informally. The data must include two auditor accounts with credentials "auditor1/auditor1" and "auditor2/auditor2".

Aquí se nos pide crear un rol auditor, que será el rol que se encargue de los code audits. Para ello se ha creado una clase en la carpeta roles llamada Auditor.

2.6) Hacer el UML (Information requirement 14)

Descripción: Produce a UML domain model.

Este requisito nos pide crear un UML que represente todas las entidades junto al rol y al dashboard creados que se han indicado previamente, así como las relaciones entre ellas.

2.7) Hacer el analysis report (Information requirement 15)

Descripción: Produce an analysis report.

Para este requisito no se han encontrado dificultades, ya que es relativamente sencillo. No ha sido necesario analizar este requisito.

2.8) Hacer el planning and progress report (Information requirement 16)

Descripción: Produce a planning and progress report.

Para este requisito no se han encontrado dificultades, ya que es relativamente sencillo. No ha sido necesario analizar este requisito.

3.) Conclusiones

En conclusión, se han tenido pocos inconvenientes en cuanto a la hora de trabajar, la única acción que tomaré de aquí a los siguientes sprints va a ser dividir mis actividades en ramas distintas para tener un mejor control del trabajo a futuro. Por lo demás, al surgir problemas me he sabido desenvolver bien y espero que este rendimiento siga para las entregas próximas.