

Grupo: C1.015

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

Analysis Report

Ángel García Escudero	(anggaesc1@alum.us.es)
Jorge Muñoz Rodríguez	(jormunrod@alum.us.es)
Isaac Solís Padilla	(isasolpad@alum.us.es)
Javier Rodríguez Reina	(javrodrei@alum.us.es)
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. Claim 5

 4.2. Objective..... 5

 4.3. Risk..... 6

 4.4. Notice..... 6

 4.5. Banner..... 7

5. Conclusiones..... 8

6. Bibliografía 9

1. Resumen ejecutivo

El análisis detallado de las entidades "Claim", "Objective", "Risk", "Notice" y "Banner" revela un enfoque proactivo y meticuloso en el diseño y desarrollo del sistema de gestión. Se han agregado restricciones y anotaciones adicionales para garantizar la integridad de los datos y la funcionalidad del sistema, incluso en casos donde los requisitos no están explícitamente definidos. Este compromiso con la calidad se refleja en la atención especial prestada al mapeo preciso de las propiedades a tipos de datos de fecha y hora en la base de datos, destacando la importancia de la implementación técnica y la interoperabilidad del sistema. Estas medidas garantizan la coherencia y la robustez del sistema en su conjunto, fortaleciendo su capacidad para satisfacer las necesidades de los usuarios y adaptarse a los cambios en el entorno operativo.

2. Tabla de revisión

Nº Revisión	Fecha	Descripción
1	07-03-2024	Versión inicial
2	08-03-2024	Versión final

3. Introducción

En el proceso de diseño y desarrollo de un sistema de gestión, es fundamental comprender las necesidades y requisitos específicos de cada entidad que interactúa con él. Este análisis detallado permite identificar y aplicar las restricciones y anotaciones adecuadas para garantizar la integridad de los datos y la funcionalidad del sistema. En este contexto, hemos examinado las entidades "Claim", "Objective", "Risk", "Notice" y "Banner", y hemos observado la implementación de restricciones adicionales, así como ajustes proactivos para satisfacer los requisitos no explícitos. Además, se ha dado especial importancia al mapeo preciso de las propiedades a tipos de datos de fecha y hora en la base de datos, evidenciando un enfoque meticuloso en la implementación técnica y la interoperabilidad del sistema. Este enfoque cuidadoso y adaptativo asegura la coherencia y la robustez del sistema de gestión en su conjunto.

4. Contenidos

4.1. Claim

- instantiationMoment: A parte de la restricción de que esa fecha esté en pasado, hemos tenido que añadir una restricción más:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
- email: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @Email: La cual he tenido que añadir para verificar que la cadena que se introduce en la base de datos tenga una estructura de email.
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.
- link: He tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.

4.2. Objective

- instantiationMoment: A parte de la restricción de que esa fecha esté en pasado, hemos tenido que añadir una restricción más:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
- status: Como esta propiedad hace referencia a que algo es crítico o no, es decir, solo hay dos posibles opciones, podremos extrapolarlo como si fuera verdadero o falso, con lo que optamos por utilizar un atributo booleano para indicar esta característica de manera clara y eficiente.
- duration: Esta propiedad hemos tenido que dividirla en dos fechas para poder tener una duración de tiempo, debido a que si solo tenemos una propiedad que se llame *duration* no podemos tener una duración de tiempo. Las dos fechas son las siguiente:
 - startDate: Indicar el comienzo de la duración y tiene como restricciones:
 - @NotNull: La cual hemos tenido que añadir porque no es un atributo opcional.
 - @Temporal (TemporalType.TIMESTAMP): La cual hemos tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - endDate: Indica el fin del periodo y tiene como restricciones:
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
 - @Temporal (TemporalType.TIMESTAMP): La cual hemos tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.

- link: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.

4.3. Risk

- identificationDate: A parte de la restricción que nos dice que la fecha debe estar en pasado, nosotros hemos añadido dos restricciones más:
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
- propability: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @Range(min = 0, max = 1): Esta restricción la hemos puesto para delimitar la propiedad y asegurarnos de que sea un porcentaje.
 - @Digits(integer = 1, fraction= 2): Esta restricción la hemos puesto para delimitar el número de decimales con los que trabajar, que en este caso hemos decidido dos decimales.
- link: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.
- value: Es una propiedad derivada, con lo que, a parte de calcularla, hemos tenido que añadirle una propiedad:
 - @Transient: Lo añadimos para que esta propiedad no se guarde en la base de datos, si no que se calcule cada vez que la necesitamos.

4.4. Notice

- instatiationMoment: A parte de la restricción de que esa fecha esté en pasado, hemos tenido que añadir una restricción más:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
- email: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @Email: La cual he tenido que añadir para verificar que la cadena que se introduce en la base de datos tenga una estructura de email.
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.
- link: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:

- @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
- @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.

4.5. Banner

- instantiationMoment: A parte de la restricción de que esa fecha esté en pasado, hemos tenido que añadir una restricción más:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
- updateMoment: A parte de la restricción de que esa fecha esté en pasado, hemos tenido que añadir una restricción más:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
- displayPeriod: Esta propiedad hemos tenido que dividirla en dos fechas para poder cumplir con la restricción que en un futuro haremos en el service. Estas dos fechas son:
 - displayStart: Que tiene como restricciones:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
 - displayEnd: Que tiene como restricciones:
 - @Temporal (TemporalType.TIMESTAMP): La cual he tenido que añadir para poder mapear la propiedad a un tipo fecha y hora en la base de datos.
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
- picture: No nos dicen ninguna restricción de forma clara con lo que nosotros hemos añadido las siguientes:
 - @NotNull: La cual he tenido que añadir porque no es un atributo opcional.
 - @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.
- link: Hemos tenido que añadir dos restricciones a pesar de que no se mencione ninguna en el documento de requisitos:
 - @URL: La cual hemos añadido para verificar que la cadena que metemos en la base de datos tenga el formato de una URL
 - @Length (max = 255): La cual hemos añadido para evitar que se produzca un error de longitud a nivel de base de datos.

-

5. Conclusiones

En resumen, el análisis detallado de las entidades "Claim", "Objective", "Risk", "Notice" y "Banner" revela una cuidadosa atención a los detalles y una adaptación proactiva a los requisitos del sistema. Se han agregado restricciones y anotaciones adicionales, incluso cuando no se mencionan explícitamente en la documentación, para garantizar la integridad de los datos y la funcionalidad del sistema. Además, se ha prestado especial atención al mapeo de las propiedades a tipos de datos de fecha y hora en la base de datos, destacando un enfoque meticuloso en la implementación técnica y la interoperabilidad del sistema.

6. Bibliografía

Intencionalmente en blanco.