Analysis report D02

Grupo C1.03.10 - Group

https://github.com/DP2-2023/Acme-L3-D02

María del Carmen García García

Email: margargar81@alum.us.es

Francisco García de la Vega García

Email:fragargar15@alum.us.es

Ignacio Jordano de Castro

Email: ignjorde@alum.us.es

Javier Ulecia García

Email: javulegar@alum.us.es

Gabriel Vidal Tévar

Email: gabvidtev@alum.us.es

17/03/2023



Grado en Ingeniería del Software – Diseño y Pruebas II

Índice

[**Resumen ejecutivo**](#_52n8aip1s8vt) **3**

[**Historial de versiones**](#_470a83r18fw2) **3**

[**Introducción**](#_in8nh87qsr73) **4**

[**Contenidos**](#_ww27ibfd8h) **4**

[**Conclusiones**](#_byyylknpky9l) **4**

[**Bibliografía**](#_hrt1r1nxhyj) **4**

# Resumen ejecutivo

En este documento se resume un análisis de los requisitos grupales para el D02. Este pretende eliminar las posibles ambigüedades en ellos y permitir su correcta implementación.

# Historial de versiones

| **Versión** | **Fecha** | **Descripción** |
| --- | --- | --- |
| 1.0 | 23/02/2023 | Versión inicial |
| 1.1 | 02/03/2023 | Añadidas algunas dudas de los requisitos |
| 1.2 | 09/03/2023 | Revisión de documento tras follow up |

# Introducción

En este archivo se realiza un listado de todos los requisitos solicitados para el entregable 2, el listado se hará en forma de tabla, donde se hablará del requisito, se analizará y se explicará la decisión que se tomará para realizarlo. Repitiendo de esta forma el proceso que ya se realizó para el anterior entregable.

# Contenidos

| **Requisito** | **Análisis** | **Decisiones** | **Validación** |
| --- | --- | --- | --- |
| 9 - Monedas del sistema | El hecho de que la moneda por defecto deba ser inicializada nos da a entender que debe persistir en la base de datos. | El requisito se implementará como una entidad, de tal forma que pueda realizarse la configuración de las monedas del sistema por medio de los datos iniciales. | Se valida esta forma de implementar el requisito. |
| 10 - Mensajes de “Peep” | - | - | - |
| 11 - Boletines | - | - | - |
| 12 - Ofertas | El sistema debe almacenar una oferta con un periodo de disponibilidad de entre al menos un día después de que se oferte y que dure minimo una semana | Como es un periodo de disponibilidad en sí se podría poner como un tipo Integer que cuente los días, ¿o habría que tener en cuenta instante de tiempo porque dependa de horas también?   * Hacerlo con un integer y contando solo los días disponibles, pero esto no tendría en cuenta que fuese el día después de la oferta, se validaría con poner @Future y mínimo 7 días? Aunque el @future podría ser complicado de usar * Separarlo en dos fechas, uno cuando inicia el período de tiempo y otro cuando acaba y a partir de ahí sacar el tiempo que está disponible tomando en cuenta de que debe ser mayor a una semana | Se aconseja implementar periodos con fechas de inicio y fin. Aunque actualmente con las entidades no se pueda validar su duración, esta comprobación se realizará más tarde.  Se desaconseja totalmente el uso de @future en general. |
| 18 - Notas | - | - | - |
| 14 - Banners | El sistema debe almacenar un banner permite a los administradores anunciar productos, servicios u organizaciones. | Se implementará como una entidad. Con un momento tipo Date de inicialización del banner y un periodo de visualización y consta de 2 atributos tipo Date uno de inicio y otro de fin. Tendrá un eslogan tipo String de menos de 75, una imagen y un enlace al documento web. | Se aconseja implementar periodos con fechas de inicio y fin. Aunque actualmente con las entidades no se pueda validar su duración, esta comprobación se realizará más tarde. |
| 15 - Tableros de administrador | El sistema debe almacenar un tablero de administrador con algunos indicadores | Se creará una entidad con todos los atributos que te piden, dentro del paquete forms | Se valida esta forma de implementar el requisito. |
| Requisitos de pruebas | - | - | - |
| 16 - Datos iniciales y de prueba | - | - | - |
| 17 - Producir un Analysis report | - | - | - |
| 8 - Producir un Planning report | - | - | - |
| 19 - Modelo UML de domínio | - | - | - |

# Conclusiones

Las dudas que han surgido son referentes a algunas validaciones que no sabíamos si había que tenerlas en cuenta ahora o posteriormente, como sería la de duración de tiempo. También han surgido dudas a la hora de cómo representar estos períodos de tiempo, pero una vez se aclara, son parecidos para todos los requisitos.

Por tanto, el problema ha sido intentar abarcar las validaciones antes de tiempo y saber cómo representar los períodos o datos con intervalos.

# Bibliografía

*Intencionadamente en blanco*