

# Analysis report



**Grado en Ingeniería Informática - Ingeniería del Software**

**Diseño y Pruebas II**

**Curso 2023 - 2024**

Código de Grupo: C1-001		
Autor	Correo	Rol
José María Baquero Rodríguez	<a href="mailto:josbaqrod@alum.us.es">josbaqrod@alum.us.es</a>	Desarrollador

Repositorio: <<https://github.com/DP2-2023-2024-C1-001/Acme-SF-D02.git>>

# Índice de Contenidos

1. Resumen Ejecutivo.....	3
2. Control de Versiones.....	4
3. Introducción.....	5
4. Contenido.....	6
5. Conclusión.....	10
6. Bibliografía.....	10

# 1. Resumen Ejecutivo

## **Objetivo:**

El objetivo de un Analysis report es proporcionar información relevante y detallada sobre las tareas realizadas en cada una de las entregas, sirviendo de respaldo para las decisiones tomadas en las diferentes etapas del proyecto. De esta forma se intenta garantizar que el diseño, desarrollo e implementación se lleve a cabo de forma efectiva y satisfaciendo las necesidades del usuario, de forma que al final del proyecto se logren todos los objetivos.

## **Participantes:**

Desarrolladores: José María Baquero Rodríguez

## **Rol y Responsabilidades:**

Desarrollador : Crear la configuración de desarrollo, personaliza el proyecto inicial, implementa características y realiza testing informal.

## **Plan de Acción:**

Crear e inicializar el repositorio en Github.

Añadir las tareas a realizar en Github.

Asignación, por parte del manager, de las tareas a los desarrolladores correspondientes.

Inicializar las ramas para ir resolviendo las tareas

Crear los pull request asignando un reviewer para comprobar que la tarea está resuelta de forma correcta. Si no, esa tarea en concreto pasa a estar resuelta y el manager se encarga de crear otra tarea para realizar la corrección de la misma.

## 2.Control de Versiones

Fecha	Versión	Descripción
07/03/2024	V1.0	Creación del documento y descripción de las implementaciones para la entrega individual D02

### **3.Introducción**

El propósito de este informe de análisis es evaluar el grado de cumplimiento de los requisitos definidos para el proyecto Acm-SF. Para ello se ha elaborado una análisis detallado para cada requisito, identificando las posibles mejoras y decisiones tomadas.

Este informe incluye un listado de registro para cada uno de los requisitos, que contienen la copia textual proporcionada para el requisito, las conclusiones derivadas del análisis y las decisiones tomadas para resolver los posibles problemas encontrados.

El informe se centra en los requisitos considerados más importantes y complejos, tanto de tipo funcional como documental.

Además al final del informe se incluye una conclusión general y la bibliografía consultada, sirviendo como una guía para la toma de decisiones y mejora continua del proyecto.

Con esta introducción se pretende facilitar la comprensión del informe a análisis y resaltar los aspectos más relevantes del proyecto.

## 4. Contenido

### Listado de registros de análisis

- 1) **Tarea individual 2:** The sponsorships are related to a project with the aim of achieving greater visibility in the market. The system must store the following data about them: a code (pattern “[A-Z]{1,3}-[0-9]{3}”, not blank, unique), a moment (in the past), a duration (after the moment, at least one month long), an amount (positive), a type of sponsorship (“Financial”, “In kind”), an optional contact email, and an optional link with further information.

La clase Sponsorship en el paquete acme.entities.sponsorship se encarga de representar los datos del patrocinio. Esta clase extiende la clase AbstractEntity, que proporciona funcionalidades comunes para todas las entidades en el sistema. La implementación de esto se ha realizado de la siguiente manera:

La clase Sponsorship tiene los siguientes atributos:

- code: representa el código del patrocinio, siguiendo el patrón especificado y siendo único.
- moment: indica el momento en el que se realizó el patrocinio, que debe estar en el pasado.
- initialDate y finalDate: representan la fecha de inicio y finalización del patrocinio respectivamente con el objetivo de calcular posteriormente una duración.
- amount: indica el monto del patrocinio, representado como un objeto de tipo Money.
- type: especifica el tipo de SponsorType, que puede ser "FINANCIAL" o "IN\_KIND".
- email: representa el correo electrónico de contacto opcional del patrocinador.
- link: un enlace opcional que proporciona más información sobre el patrocinio.
- project: establece la relación con el proyecto al que está asociado el patrocinio.
- sponsor: establece la relación con el patrocinador del proyecto.

- 2) **Tarea individual 3:** Each sponsorship is billed through the use of invoices. The system must store the following data about them: a code (pattern "IN-[0-9]{4}-[0-9]{4}", not blank, unique), a registration time (in the past), a due date (at least one month ahead the registration time), a quantity (positive not nought), the tax that it is applied (positive or nought), the total amount (calculated by adding together the quantity and the tax applied), and an optional link with further information.

La entidad Invoice se encarga de gestionar las facturas asociadas a cada patrocinio dentro del sistema. Cada factura contiene información crucial para el registro y la contabilidad de los patrocinios, asegurando una correcta gestión financiera. A continuación se detalla cómo se ha implementado esta funcionalidad:

- code: La propiedad code representa un código único para cada factura, siguiendo el patrón "IN-[0-9]{4}-[0-9]{4}". Esto garantiza la identificación exclusiva de cada factura y su integridad en la base de datos.
- registrationTime y dueDate: representan la fecha de registro de la factura y su fecha de vencimiento respectivamente. Ambas están validadas para asegurar que estén en el pasado. Además que la fecha "dueDate" debe ser al menos un mes posterior a la fecha de registro.
- quantity: representa como un objeto de tipo Money la cantidad facturada.
- tax: representa el impuesto aplicado a la factura, restringido a un rango entre 0 y 100 para representar un porcentaje válido.
- totalAmount: método que calcula el monto total de la factura sumando la cantidad facturada y el impuesto aplicado. Este enfoque asegura que el monto total sea calculado de manera consistente y precisa en función de los datos ingresados.
- link: es un campo opcional que permite adjuntar información adicional sobre la factura.

- 3) **Tarea individual 4:** The system must handle sponsor dashboards with the following data: total number of invoices with a tax less than or equal to 21.00%; total number of sponsorships with a link; average, deviation, minimum, and maximum amount of the sponsorships; average, deviation, minimum, and maximum quantity of the invoices.

La implementación de este panel de control se ha realizado mediante la clase SponsorDashboard en el paquete acme.forms. Esta clase proporciona una estructura para almacenar y manejar los datos del panel de control del patrocinador de manera eficiente. A continuación se detalla cómo se ha implementado esta funcionalidad:

- `totalNumberOfInvoicesWithTaxLessOrEquals21Percent` : almacena el número total de facturas que tienen un impuesto menor o igual al 21.00%.
- `totalNumberOfSponsorshipWithLink`: representa el número total de patrocinios que tienen un enlace asociado.
- Las propiedades `averageAmountOfSponsorship`, `deviationAmountOfSponsorship`, `minimumAmountOfSponsorship` y `maximumAmountOfSponsorship` almacenan estadísticas sobre los montos de los patrocinios.
- Las propiedades `averageQuantityOfInvoice`, `deviationQuantityOfInvoice`, `minimumQuantityOfInvoice` y `maximumQuantityOfInvoice` contienen estadísticas sobre las cantidades de las facturas.



**4) Tarea individual 5: Produce assorted sample data to test your application informally. The data must include two sponsor accounts with credentials “sponsor1/ sponsor1” and “sponsor2/ sponsor2”.**

En el proceso de desarrollo he generado datos de muestra que abarcan diversas áreas de la aplicación, como cuentas de usuario, patrocinios y facturación asociada.

- Archivo user-account.csv: Este archivo contiene las cuentas de usuario para varios roles dentro de nuestra aplicación. Incluye usuarios para roles como consumidores, proveedores, administradores, auditores, clientes y, lo más relevante para este informe, sponsors. Específicamente, se han creado dos cuentas de usuario para patrocinadores, con las credenciales "sponsor1" y "sponsor2", respectivamente.
- Archivo sponsorship.csv: en este archivo se encuentran datos de muestra para los patrocinios asociados a proyectos específicos. Cada patrocinio tiene atributos de prueba como code, registration-time y due-date, amount, type, email y link. Además se incluyen relaciones de prueba con otras entidades como son Project y con el rol Sponsor.
- Archivo invoice.csv: por último, el archivo de facturas contiene datos de muestra para las facturas asociadas a los patrocinios. Cada factura incluye detalles de prueba como code, registration-time, due-date, quantity, tax, link y la referencia con la relación al patrocinio correspondiente.

- 5) **Tarea individual 13:** There is a new project-specific role called sponsor, which has the following profile data: name (not blank, shorter than 76 characters), a list of expected benefits (not blank, shorter than 101 characters), an optional web page with further information, and an optional email contact.

El rol de Sponsor se ha implementado como una entidad en el sistema, representada por la clase Sponsor en el paquete acme.roles. Esta clase contiene los siguientes atributos:

- name: Almacena el nombre del patrocinador.
- benefits: Almacena la lista de beneficios esperados.
- webPage: Almacena la URL de la página web del patrocinador.
- email: Almacena la dirección de correo electrónico de contacto del patrocinador.

## 5. Conclusión

En este informe de análisis, se ha evaluado la implementación de nuevas funcionalidades en la aplicación, incluyendo la creación de un rol específico para patrocinadores, la gestión de patrocinios y facturación asociada, así como la adición de opciones en el menú de la aplicación. Se ha concluido que estas implementaciones son viables y ofrecen una base sólida para futuros desarrollos. Aunque no son requisitos complejos, proporcionan una estructura robusta y permiten un proceso de mejora continuo en el proyecto.

## 6. Bibliografía

Intencionadamente blanco.