

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática

DP2-Informe de Planificación y Progreso


Student 4



Grado en Ingeniería Informática – Ingeniería del Software

Diseño y Pruebas II

Curso 2023 – 2024

<u>Group:</u>	C2.X02
<u>Repository:</u>	https://github.com/SoniaRM/Acme-SF-D04-24.5.0
<u>Student #4</u>	<div><p>UVUS: josmirmar2 Name: José Manuel Miret Martín Email: josmirmar2@alum.us.es</p></div>
<u>Date:</u>	Sevilla Julio 2, 2024

Índice de contenido

1. Versiones	3
2. Listado de tareas	3
3. Capturas de Pantalla	5
4. Presupuesto	6
5. Lista de registros	7
6. Conflictos surgidos	7
7. Comparación de costes estimados / reales	8
8. Tareas específicas para la convocatoria 2	8
9. Presupuesto actualizado	14
10. Tareas específicas para la convocatoria 3	14
11. Presupuesto actualizado	23

1. Versiones

Versión	Fecha	Autor
1.0	22/04/2024	José Manuel Miret Martín
1.1	02/07/2024	José Manuel Miret Martín
1.2	07/10/2024	José Manuel Miret Martín

2. Listado de tareas

Task 6: Operations by **sponsors** on **sponsorships**:

- List the **sponsorships** that they have created.
- Show the details of their **sponsorships**.
- Create, update, or delete their **sponsorships**. **Sponsorships** can be updated or deleted as long as they have not been published. For a **sponsorship** to be published, the sum of the **total amount** of all their **invoices** must be equal to the **amount** of the sponsorship.

Para ello llevaré a cabo el rol de desarrollador.

Task 7: Operations by **sponsors** on **invoices**:

- List the **invoices** in their **sponsorships**.
- Show the details of their **invoices**.
- Create and publish an **invoice**.
- Update or delete an **invoice** as long as it is not published.

Para ello llevaré a cabo el rol de desarrollador.

Task 8: Operations by **sponsors** on **sponsor** dashboards:

- Show their **sponsor** dashboards.

Para ello llevaré a cabo el rol de desarrollador.

Task 17: Operations by anonymous principals on user accounts:

- Sign up to the system and become a **sponsor**

Para ello llevaré a cabo el rol de desarrollador.

Task 18: Operations by **sponsors** on user accounts:

- Update their profiles.

Para ello llevaré a cabo el rol de desarrollador

Task 19: Operations by any principals on **sponsorships**:

- List the **sponsorships** in the system that are published.
- Show the details of the **sponsorships** that they can list (excluding their **invoices**).

Para hacerlo deberé hacer el rol de desarrollador.

Task 20. Produce an analysis report.

Para hacerlo llevaré a cabo el rol de project manager.

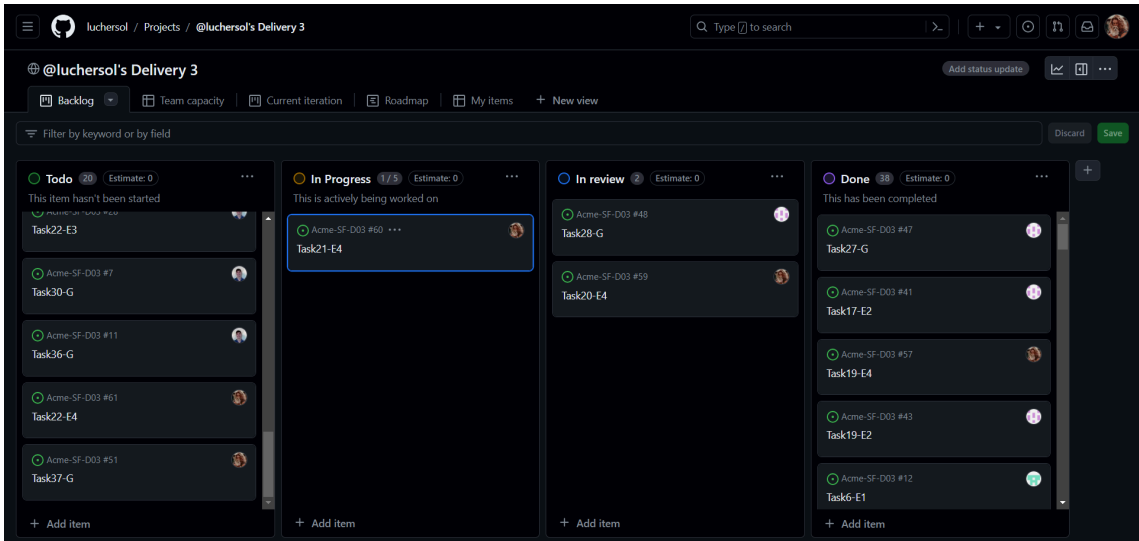
Task 21. Produce a planning and progress report.

Para hacerlo llevaré a cabo el rol de project manager.

Task 22. Produce a lint report.

Para ello llevaré a cabo el rol de project manager.

3. Capturas de Pantalla



[Home](#) [Sponsor](#) [Authenticated](#) [Any](#) [Account](#) [Sign out](#)

Sponsorship form

Code:

Project:

Start date:

End date:

Sponsorship type:

Email (Optional):

More information (Optional):

[Update](#) [Delete](#) [Publish](#) [Invoices](#) [Return](#)

HomeSponsorAuthenticatedAnyAccountSign out

Sponsorship form

Code:

Lorem ipsum

Must match pattern "[A-Z]{1,3}-[0-9]{3}":. Must not be blank.

Project:

May not be null.

Start date:

yyyy/mm/dd hh:mm

May not be null.

End date:

yyyy/mm/dd hh:mm

May not be null.

Sponsorship type:

May not be null.

Email (Optional):

user@mail.com

More information (Optional):

http://www.acme.com

CreateReturn

ACME ACNE SF

YOUR FIRST PROJECT IN DP2

HomeSponsorAuthenticatedAnyAccountSign out

Listing of sponsorships

Search:

Showing data 1 - 5 out of 10 data

Moment	Amount	Code
2021/01/15 05:20	EUR 1,050.24	ABC-001
2021/02/23 19:20	BGP 1,060.13	LMN-020
2021/03/25 11:30	BGP 1,080.00	MNO-005
2021/04/18 06:55	EUR 2,200.00	PQR-006
2021/05/25 12:15	USD 4,204.10	GHI-003

Show 5 items

12

CreateReturn

HomeSponsorAuthenticatedAnyAccountSign out

Listing of invoices

Search:

Showing data 1 - 1 out of 1 data

Registration time	Due date	Code
2020/06/15 08:30	2019/06/15 08:30	IN-1842-7159

Show 5 items

1

CreateReturn

4. Presupuesto

Tarea	Tiempo estimado (en horas)	Tiempo dedicado(en horas)	Coste estimado	Coste real	Rol
Task 6	12	11,88	240	237,6	Desarrollador
Task 7	9	6	180	120,00	Desarrollador
Task 8	5	3,33	100	66,60	Desarrollador
Task 17	3	1	60	20,00	Desarrollador
Task 18	1	0,26	20	5,20	Desarrollador
Task 19	4	1,2	80	24,00	Desarrollador
Task 20	1,5	0,5	30	10,00	Project Manager
Task 21	1,5	0,48	30	9,60	Project Manager
Task 22	3	1	60	20,00	Project Manager

Coste total estimado: 800 €

Coste total real: 513,00 €

5. Lista de registros

Para esta entrega no realicé ningún registro.

6. Conflictos surgidos

La mayoría de conflictos vinieron a raíz de una cuestión: la falta de conocimiento sobre el manejo y uso del framework para crear los servicios, el repositorio y el controller, junto las vistas en la primera tarea.

Tuve otros problemas pequeños, como la gestión de errores, ya que no tenía mucha idea de como poder mostrarlos y poder conocer donde se localizaba. Sin embargo, hacer debug y leer detenidamente los errores me ayudaron a comprenderlos. Si había alguno muy largo o no llegaba a conocer, preguntaba a la inteligencia artificial chatGPT para que me lo localizara, y a partir de su ayuda poder resolver el conflicto

Sobre las queries no hubo mucho inconveniente debido a mi conocimiento y mi previo manejo de estos en asignaturas anteriores. Sin embargo, sí que se me complicó un poco el tema de cómo elaborar algunas queries para poder realizar el promedio y la desviación. Sin embargo, con ayuda de la profesora en las

revisiones realizadas en clase y la ayuda de mis compañeros, pude resolver esas dudas surgidas.

7. Comparación de costes estimados / reales

En esta ocasión, he conseguido acabar las tareas en un tiempo menor al esperado. Si que es cierto que la primera tarea es la que mayor tiempo me llevó terminarla, debido a mi falta de conocimiento. Sin embargo, después de haber entendido el framework y cómo poder elaborar las funciones y el código, el resto de tareas las elaboré en mucho menos tiempo de lo que esperé, lo cual me alegra.

Viendo esta entrega con respecto a entregas anteriores, veo que he avanzado y mejorado a la hora de hacer código, y puedo observar que empiezo a acostumbrarme a este framework a medida que pasa el tiempo.

8. Tareas específicas para la convocatoria 2

Para esta segunda convocatoria se ha comentado varios cambios a realizar en el código para corregir lo elaborado en la anterior convocatoria, entregable 3. Estas tareas específicas son:

Se puede patrocinar un proyecto que no está publicado:

Se puede ~~patrocinar~~ un proyecto que no está publicado. Los proyectos no publicados sólo pueden verse por sus managers propietarios.

The image shows a 'Sponsorship form' and a database table view. The form has fields for Code, Project, Start date, End date, Sponsorship type, Amount, Email (Optional), and More information (Optional). The database table view shows a table with columns: id, version, abstract_project, code, cost_amount, cost_currency, draft_mode, indication, and link. The table contains two rows of data.

id	version	abstract_project	code	cost_amount	cost_currency	draft_mode	indication	link
1	10	2	Consectetur elit po	AAA-1234	10,000	EUR	0	0
2	19	0	Ut et libero officia	AAA-1112	5,000	EUR	1	0

Para esta tarea se elaboró una nueva restricción en la función validate de SponsorSponsorshipPublishService, elaborando una consulta que indique si el proyecto elegido está o no publicado:

```
if (!super.getBuffer().getErrors().hasErrors("project")) {  
    Boolean isDraftMode = this.repository.projectIsDraftMode(object.getProject().getId());  
    super.state(!isDraftMode, "project", "sponsor.sponsorship.form.error.not-published-project");  
}
```

Se puede patrocinar algo que empieza en el pasado:

Sponsorship form

Code: A-000

Project: AAA-1112

Start date: 2022/06/07 10:00

End date: 2022/07/08 10:30

Sponsorship type: FINANCIAL

Amount: EUR 0.00

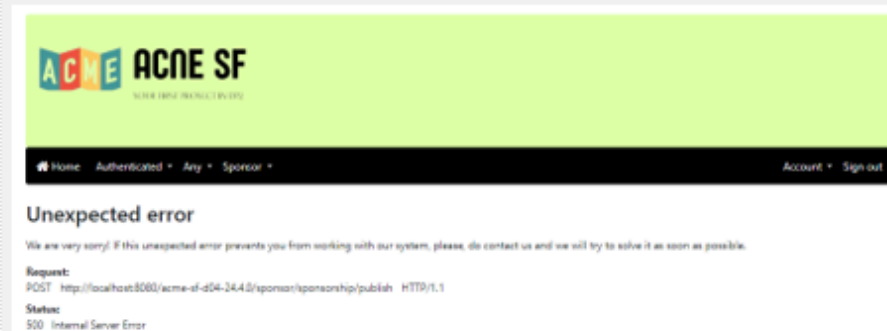
Email (Optional): consumer-test-1@mail.com

More information (Optional): http://www.acme.com

Update Delete Publish Invoices Return

En el enunciado, requisito 2, se indica que la duración tiene que comenzar después del momento en el que se crea el patrocinio, es decir, a partir del 2022/07/30 00:01.

Si hago clic en publicar la entidad anterior, el sistema lanza la siguiente excepción:



Una ~~sponsorización~~ no puede estar en el pasado. Se va a patrocinar un proyecto. No se hacen patrocinios en el pasado.

Para esta tarea se elaboró una modificación a las restricciones con respecto las fechas endDate y startDate, en la función validate de SponsorSponsorshipPublishService, SponsorSponsorshipCreateService y SponsorSponsorshipUpdateService:

```

if (!super.getBuffer().getErrors().hasErrors("startDate")) {
    Date minimumDeadline;

    if (!super.getBuffer().getErrors().hasErrors("moment")) {
        super.state(MomentHelper.isAfterOrEqual(object.getStartDate(), object.getMoment()), "startDate", "sponsor.sponsorship.form.error.invalid-moment");

        minimumDeadline = object.getEndDate() == null ? null : MomentHelper.deltaFromMoment(object.getEndDate(), 1, ChronoUnit.MONTHS);
        super.state(object.getEndDate() == null || MomentHelper.isBefore(object.getStartDate(), minimumDeadline), "startDate", "sponsor.sponsorship.form.error.invalid-moment");
    } else
        super.state(false, "startDate", "sponsor.sponsorship.form.error.invalid-moment");
}

if (!super.getBuffer().getErrors().hasErrors("endDate")) {
    Date maximumDeadline;

    if (!super.getBuffer().getErrors().hasErrors("moment")) {
        super.state(MomentHelper.isAfterOrEqual(object.getEndDate(), object.getMoment()), "endDate", "sponsor.sponsorship.form.error.invalid-moment");

        maximumDeadline = object.getStartDate() == null ? null : MomentHelper.deltaFromMoment(object.getStartDate(), 1, ChronoUnit.MONTHS);
        super.state(object.getStartDate() == null || MomentHelper.isAfter(object.getEndDate(), maximumDeadline), "endDate", "sponsor.sponsorship.form.error.invalid-moment");
    } else
        super.state(false, "endDate", "sponsor.sponsorship.form.error.invalid-moment");
}

```

Una sponsorización no puede estar en el pasado:

Para esta tarea se elaboró una modificación a las restricciones con respecto la entidad Sponsorship, con las propiedades endDate y startDate, eliminando “@Past”:

```

@PastOrPresent
@Temporal(TemporalType.TIMESTAMP)
@NotNull
private Date moment;

@Temporal(TemporalType.TIMESTAMP)
@NotNull
private Date startDate;

@Temporal(TemporalType.TIMESTAMP)
@NotNull
private Date endDate;

```

Se admiten invoices que no cumplen con la restricción relacionada con el due date:

Se admiten invoices que no cumplen con la restricción que indica que el due date debe ser al menos un mes posterior a la fecha de creación de la invoice, es decir, de la fecha del sistema. En otras palabras, solo se pueden crear invoices cuyo atributo due date sea a partir de 2022/08/30 00:00. En la siguiente captura se muestra una invoice creada en el 2020.2020/10/10 10:00

The screenshot shows a web form titled "Invoice form". It contains several input fields: "Code" with the value "IN-0000-9999", "Due date" with the value "2020/10/10 10:00", "Quantity" with the value "EUR 100.00", and "Tax" with the value "120.00". Below these is a section "More information (Optional):" with a text input field containing "http://www.acme.com". At the bottom of the form are four buttons: "Update", "Delete", "Publish", and "Return".

Para esta tarea se ha modificado la restricción relacionada con la relación entre due date y registration time. Esta restricción impide que la fecha establecida en el campo due date no sea al menos un mes posterior a la fecha de la creación de la invoice, elaborada en la función validate de SponsorInvoicePublishService, SponsorInvoiceCreateService y SponsorInvoiceUpdateService:

```
if (!super.getBuffer().getErrors().hasErrors("dueDate")) {
    Date maximumDeadline;

    if (!super.getBuffer().getErrors().hasErrors("registrationTime")) {
        maximumDeadline = MomentHelper.deltaFromMoment(object.getRegistrationTime(), 1, ChronoUnit.MONTHS);
        super.state(MomentHelper.isAfter(object.getDueDate(), maximumDeadline), "dueDate", "sponsor.invoice.form.error.to-close-from-");
    } else {
        super.state(false, "dueDate", "sponsor.invoice.form.error.incorrect-registration-time");
    }
}
```

Corrección relacionada con la cantidad establecida tanto en invoice como en sponsorship:

Inicio

Autenticado

Cualquiera

Patrocinador

Cuenta

Cerrar sesión

Detalles de patrocinio

Código:

ZZZ-123

Proyecto:

AAA-1234

Fecha inicio:

10/10/2021 10:00

Fecha final:

10/11/2021 10:30

Tipo de patrocinio:

FINANCIAL

Cantidad:

0,00 EUR

Correo (Opcional):

user@mail.com

Más información (Opcional):

http://www.acme.com

Actualizar

Borrar

Publicar

Facturas

Volver

Inicio

Autenticado

Cualquiera

Patrocinador

Cuenta

Cerrar sesión

Detalles de patrocinio

Código:

IN-0000-0000

Fecha de vencimiento:

10/05/2022 10:00

Cantidad:

400,00 GBP

Impuesto:

5,00

Más información (Opcional):

http://www.acme.com

Volver

Anteriormente, la cantidad de un sponsorship no se relacionaba adecuadamente con las cantidades de sus invoices. es decir, no se cumplía adecuadamente la restricción: *“For a **sponsorship** to be published, the sum of the **total amount** of all their **invoices** must be equal to the **amount** of the sponsorship.”*

Ahora, se ha modificado el código relacionado con esta restricción. Por lo que cada vez que creas, modifies o publiques una invoice se autoconfigura la cantidad

relacionada con su sponsorship. Esta configuración se elabora en la función perform de SponsorInvoicePublishService, SponsorInvoiceCreateService y SponsorInvoiceUpdateService:

```
@Override
public void perform(final Invoice object) {
    assert object != null;

    Sponsorship sponsorship;
    Double invoicesAmounts;
    Money finalMoney;
    String systemCurrency;

    object.setDraftMode(false);
    this.repository.save(object);

    sponsorship = object.getSponsorship();

    invoicesAmounts = this.repository.findManyInvoicesBySponsorshipId(sponsorship.getId()).stream() //
        .mapToDouble(i -> i.totalAmount().getAmount() / this.repository.findMoneyConvertByMoneyCurrency(i.totalAmount().getCurrency())) //
        .sum();

    systemCurrency = this.repository.findSystemConfiguration().getSystemCurrency();

    finalMoney = new Money();
    finalMoney.setAmount(Math.round(invoicesAmounts * this.repository.findMoneyConvertByMoneyCurrency(systemCurrency) * 100.0) / 100.0);
    finalMoney.setCurrency(this.repository.findSystemConfiguration().getSystemCurrency());

    sponsorship.setAmount(finalMoney);
    this.repository.save(sponsorship);
}
```

El cálculo no es correcto el número total de sponsorships with a link:

El cálculo no es correcto el número total de sponsorships with a link:

Sponsor form

General indicators

Total number of invoices with a tax less than or equal to 21.00%	0
Total number of sponsorships with a link	2
Average amount of the sponsorships	EUR 0.00
Deviation amount of the sponsorships	EUR 0.00
Minimum amount of the sponsorships	EUR 0.00
Maximum amount of the sponsorships	EUR 0.00
Average quantity of the invoices	EUR 500.00
Deviation quantity of the invoices	EUR 0.00
Minimum quantity of the invoices	EUR 500.00
Maximum quantity of the invoices	EUR 500.00

En la imagen anterior aparecen dos, pero en la base de datos, solo hay una con un link. De las tres que tengo en la BD solo las dos primeras son de ese sponsor, de modo que las están contando todas, las que tienen un link y las que no.

id	email	end_date	link	moment	start_date	type	project_id	status
1	gutydelgado@mailinator.com	2020-12-10 10:10:00.000	https://www.google.com	2022-07-30 00:00:00.000	2020-10-10 10:00:00.000	0	19	active
2	gutydelgado@mailinator.com	2021-12-10 10:10:00.000		2022-07-30 00:00:00.000	2021-10-10 10:00:00.000	0	19	active
3	gutydelgado@mailinator.com	2021-11-10 10:10:00.000		2022-07-30 00:00:00.000	2021-10-10 10:00:00.000	0	19	active

Para esta tarea se ha modificado la consulta a la base de datos elaborada en SponsorDashboardRepository, para que cuente todos los sponsorships tal que el id es el indicado, el valor que tiene en el campo link no sea nulo ni que, después de eliminar los espacios en blanco al inicio y al final, no sea una cadena vacía:

```
@Query("SELECT COUNT(s) FROM Sponsorship s WHERE s.sponsor.id = :id AND s.link IS NOT NULL AND TRIM(s.link) != '' ")
Integer totalSponsorshipWithLink(int id);
```

9. Presupuesto actualizado

Tarea	Tiempo estimado (en horas)	Tiempo dedicado(en horas)	Coste estimado	Coste real	Rol
Task 6	12	16,32	240	326,4	Desarrollador
Task 7	9	8,72	180	174,4	Desarrollador
Task 8	5	5,26	100	105,2	Desarrollador
Task 17	3	1	60	20,00	Desarrollador
Task 18	1	0,26	20	5,20	Desarrollador
Task 19	4	1,2	80	24,00	Desarrollador
Task 20	1,5	0,5	30	10,00	Project Manager
Task 21	1,5	0,81	30	16,2	Project Manager
Task 22	3	1	60	20,00	Project Manager

Coste total estimado: 800 €

Coste total real: 701,4 €

Esta vez el coste se ajusta mucho más al estimado, debido a que en las tareas 6, 7, 8 y 21 ha aumentado el tiempo. Las tareas 21, 7 y 8 tienen un mejor ajuste que la vez anterior, acercándose más al estimado. Sin embargo, la tarea 6 se ha alejado mucho más del coste estimado, haciendo que el coste real sea mucho mayor.

10. Tareas específicas para la convocatoria 3

Para esta tercera convocatoria se ha comentado varios cambios cambios a realizar en el código para corregir lo elaborado en la anterior convocatoria, entregable 3. Estas tareas específicas son:

Se puede patrocinar un proyecto que no está publicado:

[] Se puede patrocinar proyectos que aún no están publicados. Los proyectos que no están publicados, solo pueden ser vistos/editados por sus managers propietarios. En la siguiente vista se muestran tres proyectos pero solo el PRJ-0002 está publicado.

Sponsorship form

Code:

Must match pattern "[A-Z]{1,3}-[0-9]{3}". Must not be blank.

Project:

PRJ-0001

PRJ-0002

PRJ-0003

May not be null.

Moment:

May not be null.

Start date:

May not be null.

End date:

May not be null.

Sponsorship type:

May not be null.

Amount

Email (Optional):

En la siguiente captura se puede observar que solo el PRJ-0002 está publicado:

localhost (root)									
project									
	123 id	123 version	abc abstract_project	abc code	123 cost	123 draft_mode	123 indication	123	AD
1	9	1	asdlfjasdlf	PRJ-0001	20	1	0		
2	11	2	Ex harum culpa cons	PRJ-0002	50	0	0		
3	12	0	Consequat Quidem co	PRJ-0003	1,000	1	0		

Este problema ya se presentaba en la convocatoria anterior.

Para esta tarea se ha hecho una alternativa con respecto a la anterior convocatoria. En este caso, ya no aparece dentro de las múltiples opciones a elegir un proyecto que no esté publicado:

```
projects = this.repository.findAllPublishedProjects();
choices = SelectChoices.from(projects, "code", object.getProject());
choicesType = SelectChoices.from(TypeOfSponsorship.class, object.getType());
```

```
@Query("select p from Project p where p.draftMode = false")
Collection<Project> findAllPublishedProjects();
```

El momento del patrocinio es el momento de creación, no se puede modificar:

El momento del patrocinio, se refiere al momento de creación y eso es un dato que configura el sistema, no un usuario final, que puede indicar cualquier fecha no siendo ésta la realidad. El momento es la fecha en la que se crea el patrocinio

Para esta tarea se elaboró una modificación a la propiedad moment, en la visualización del mismo en el form, donde se ha cambiado para que ahora solo pueda leerse y no se pueda modificar:

```
sponsor = this.repository.findOneSponsorById(super.getRequest().getPrincipal().getActiveRoleId());

object = new Sponsorship();
object.setDraftMode(true);
object.setSponsor(sponsor);
object.setMoment(MomentHelper.getCurrentMoment());

super.getBuffer().addData(object);
```

```
7
8 <acme:form>
9   <acme:input-textbox code="sponsor.sponsorship.form.label.code" path="code"/>
10  <acme:input-select code="sponsor.sponsorship.form.label.project" path="project" choices="${projects}"/>
11  <acme:input-moment code="sponsor.sponsorship.form.label.moment" path="moment" readonly="true"/>
12  <acme:input-moment code="sponsor.sponsorship.form.label.startDate" path="startDate"/>
13  <acme:input-moment code="sponsor.sponsorship.form.label.endDate" path="endDate"/>
14  <acme:input-select code="sponsor.sponsorship.form.label.type" path="type" choices="${types}"/>
15  <acme:input-money code="sponsor.sponsorship.form.label.amount" path="amount"/>
16  <acme:input-email code="sponsor.sponsorship.form.label.email" path="email"/>
17  <acme:input-url code="sponsor.sponsorship.form.label.link" path="link"/>
18
```

Problemas a la hora de modificar la cantidad del patrocinio:

No puedo configurar la cantidad del patrocinio, ¿por qué? es un dato que el usuario final debería de poder introducir

Sponsorship form

Code:

ABC-111

Must match pattern "[A-Z]{1,3}-[0-9]{3}". Must not be blank.

Project:

PRJ-0002

May not be null.

Moment:

2020/10/10 10:00

May not be null.

Start date:

2020/10/10 10:00

May not be null.

End date:

2020/10/10 09:00

May not be null.

Sponsorship type:

FINANCIAL

May not be null.

Amount

EUR 123,456.78

Email (Optional):

user@mail.com

More information (Optional):

http://www.acme.com

Create

Return

Para esta tarea se elaboró una modificación a las restricciones, ya que anteriormente dicho valor se actualizaba de forma automática, sin que el usuario tuviese que introducir valor. Dicha función se eliminó y se cambió por que el usuario pudiese introducir un valor, y tras ello añadir una restricción que validase la restricción “For a **sponsorship** to be published, the sum of the total amount of all their invoices must be equal to the amount of the sponsorship” en el validate tanto en la clase “SponsorSponsorshipPublishService” como en “SponsorInvoicePublishService”:

```
Collection<Invoice> invoices = this.repository.findManyInvoicesBySponsorshipId(object.getId());
double sumTotal = 0.0;
if (object.getAmount() != null) {
    String systemCurrency;
    for (Invoice i : invoices) {
        systemCurrency = this.repository.findSystemConfiguration().getSystemCurrency();
        sumTotal += i.totalAmount().getAmount() * this.repository.findMoneyConvertByMoneyCurrency(systemCurrency);
    }

    double factor = Math.pow(10, 2);
    sumTotal = Math.round(sumTotal * factor) / factor;
} else
    sumTotal = -10000.00;

if (!super.getBuffer().getErrors().hasErrors("amount")) {
    Money amount;
    boolean bool;
    amount = object.getAmount();
    bool = amount.getCurrency().equals("EUR") || amount.getCurrency().equals("USD") || amount.getCurrency().equals("GBP");

    super.state(amount.getAmount() >= 0, "amount", "sponsor.sponsorship.form.error.negative-amount");
    super.state(bool, "amount", "sponsor.sponsorship.form.error.wrong-currency");
    if (bool)
        super.state(amount.getAmount() * this.repository.findMoneyConvertByMoneyCurrency(amount.getCurrency()) == sumTotal, "amount", "sp
```

```

if (!super.getBuffer().getErrors().hasErrors("quantity")) {
    double before;
    double res;
    Collection<Invoice> invoices;

    if (object.getTax() != null) {
        invoices = this.repository.findManyInvoicesBySponsorshipId(object.getSponsorship().getId());
        before = this.repository.findOneInvoiceById(object.getId()).totalAmount().getAmount();
        res = object.totalAmount().getAmount() - before;
        for (Invoice i : invoices)
            res += i.totalAmount().getAmount();
    } else
        res = -10000000000.00;

    super.state(object.getQuantity().getAmount() > 0, "quantity", "sponsor.invoice.form.error.negative-salary");
    super.state(Arrays.asList(this.repository.findAcceptedCurrencies().split(",")).contains(object.getQuantity().getCurrency()), "quantity", "quantity");
    super.state(res <= object.getSponsorship().getAmount().getAmount(), "quantity", "sponsor.invoice.form.error.bad-total-amount");
}

```

Se admiten invoices que no cumplen con la restricción relacionada con el due date:

Con los siguientes datos, ni se crea el patrocinio, ni se muestran mensajes de error:

Sponsorship form

Code:

ABC-111

Project:

PRJ-0002

Moment:

2020/10/10 10:00

Start date:

2020/10/10 10:00

End date:

2020/11/10 10:00

Sponsorship type:

FINANCIAL

Amount

EUR 123,456.78

Email (Optional):

user@mail.com

More information (Optional):

http://localhost:8080/Acme-SF-D04/sponsor/sponsorship/create

Create

Return

A veces sale el mensaje siguiente:

Sponsorship form

Code:
AAA-000

Project:
AAA-1021

You cannot make a sponsorship from a non published project.

Moment:
2022/07/10 10:00

Start date:
2022/07/10 10:00

End date:
2022/08/31 10:00

Sponsorship type:
FINANCIAL

Amount
EUR 0.00

Email (Optional):
user@mail.com

More information (Optional):
http://www.acme.com

Update Delete Publish Invoices Return

y otras veces no. No sé si depende de que anteriormente el usuario estaba creado o que el proyecto ya estaba en la base de datos o recién creado, pero no funciona en todos los casos.

Para esta tarea, como ya he comentado anteriormente, he modificado la propiedad "project". Ahora, como solamente salen los proyectos que están publicados y almacenados en la base de datos, no es posible que salga el mensaje de error, ya que el usuario no debería de poder introducir un proyecto que no esté publicado. Independientemente si ha sido recién creado:

```
projects = this.repository.findAllPublishedProjects();
choices = SelectChoices.from(projects, "code", object.getProject());
choicesType = SelectChoices.from(TypeOfSponsorship.class, object.getType());
```

```
@Query("select p from Project p where p.draftMode = false")
Collection<Project> findAllPublishedProjects();
```

Publicar un sponsorship sin invoices creadas ni publicadas:

De la misma forma, en algunos casos puedo publicar un patrocinio sin invoices y otras veces muestra el mensaje de error.

Sponsorship form

Code:
AAA-000

Project:
AAA-0001

Moment:
2022/07/10 10:00

Start date:
2022/07/10 10:00

End date:
2022/08/31 10:00

Sponsorship type:
FINANCIAL

Amount
EUR 0.00

Email (Optional):
user@mail.com

More information (Optional):
http://www.acme.com

Return

He publicado el proyecto anterior y no tiene ninguna invoice porque aún no he creado ninguna invoice en el sistema. Eso se puede ver porque el atributo amount está a 0, lo que significa que no se ha actualizado con las cantidades de ninguna invoice.

Anteriormente, no consideré la opción de publicar un patrocinio sin anteriormente creado y publicado sus facturas correspondientes. Es decir, se podía publicar un patrocinio sin facturas.

Ahora, se ha modificado el código relacionado con esta restricción. No solo se comprueba si se ha creado una factura para ese patrocinio, además comprueba de que ha sido publicado:

```
if (!super.getBuffer().getErrors().hasErrors()) {  
  super.state(!this.repository.noneInvoicesBySponsorshipId(object.getId()), "**", "sponsor.sponsorship.form.error.none-invoices");  
  super.state(this.repository.allInvoicesPublishedBySponsorshipId(object.getId()), "**", "sponsor.sponsorship.form.error.publish-invoices")  
}
```

Se puede modificar el momento de registro de una factura:

⌚ También se permite modificar el momento por el usuario cuando eso es un dato que configura el sistema.

Para esta tarea se ha hecho, como ya se ha comentado anteriormente, una modificación a la propiedad registration time, en la visualización del mismo en el form, donde se ha cambiado para que ahora solo pueda leerse y no se pueda modifica:

```

masterId = super.getRequest().getData("masterId", int.class);
sponsorship = this.repository.findOneSponsorshipById(masterId);

object = new Invoice();
object.setDraftMode(true);
object.setSponsorship(sponsorship);
object.setSponsor(sponsor);
object.setRegistrationTime(MomentHelper.getCurrentMoment());

super.getBuffer().addData(object);
}

<acme:form>
<acme:input-textbox code="sponsor.invoice.form.Label.code" path="code"/>
<acme:input-moment code="sponsor.invoice.form.Label.registrationTime" path="registrationTime" readonly="true"/>
<acme:input-moment code="sponsor.invoice.form.Label.dueDate" path="dueDate"/>
<acme:input-money code="sponsor.invoice.form.Label.quantity" path="quantity"/>
<acme:input-double code="sponsor.invoice.form.Label.tax" path="tax"/>
<acme:input-url code="sponsor.invoice.form.Label.Link" path="Link"/>

```

Algunos mensajes de error no se explican adecuadamente:

Muestra algunos errores, pero no queda claro qué hay que hacer. Por ejemplo, en el due date se indica que no es correcto, pero no indica por qué.

Invoice form

Code:
IN-9999-8888

Registration time:
2022/07/30 23:59
must be a date in the past or in the present.

Due date:
2022/07/30 23:59
The registration time is incorrect.

Quantity:
EUR 800.00

Tax:
1.00

More information (Optional):
http://www.acme.com

Create **Return**

Para esta tarea se ha mejorado la descripción de algunos mensajes de error, indicando que deben de cambiar y qué campos deben de cambiar si es necesario para que la propiedad en donde sale el error sea correcto, por ejemplo en ese caso se ha cambiado el mensaje de error a:

When publishing an invoice, the total amount of the invoice must not be greater than that of the sponsorship to which it belongs

Se puede modificar el momento de registro de una factura:

No deja persistir la invoice justo en la fecha del sistema (aunque en el error de validación se indica que se puede indicar una fecha en el pasado o en el presente.

Este problema fue solucionado anteriormente, cuando se ha cambiado la fecha de registro de la factura para que se ponga automáticamente, sin que el usuario pueda modificarla:

```
masterId = super.getRequest().getData("masterId", int.class);
sponsorship = this.repository.findOneSponsorshipById(masterId);

object = new Invoice();
object.setDraftMode(true);
object.setSponsorship(sponsorship);
object.setSponsor(sponsor);
object.setRegistrationTime(MomentHelper.getCurrentMoment());

super.getBuffer().addData(object);
}

<acme:form>
<acme:input-textbox code="sponsor.invoice.form.label.code" path="code"/>
<acme:input-moment code="sponsor.invoice.form.label.registrationTime" path="registrationTime" readonly="true"/>
<acme:input-moment code="sponsor.invoice.form.label.dueDate" path="dueDate"/>
<acme:input-money code="sponsor.invoice.form.label.quantity" path="quantity"/>
<acme:input-double code="sponsor.invoice.form.label.tax" path="tax"/>
<acme:input-url code="sponsor.invoice.form.label.link" path="link"/>
```

A su vez, se ha modificado las restricciones de validación de las fechas, tanto de invoice como de sponsorship:

```
if (!super.getBuffer().getErrors().hasErrors("dueDate")) {
    Date minimumDeadline;

    minimumDeadline = MomentHelper.deltaFromMoment(object.getRegistrationTime(), 30, ChronoUnit.DAYS);
    super.state(MomentHelper.isAfter(object.getDueDate(), minimumDeadline), "dueDate", "sponsor.invoice.form.error.too-close-moment");
}

if (!super.getBuffer().getErrors().hasErrors("startDate")) {
    Date minimumDeadline;

    super.state(MomentHelper.isAfterOrEqual(object.getStartDate(), object.getMoment()), "startDate", "sponsor.sponsorship.form.error.too-close-moment");

    minimumDeadline = object.getEndDate() == null ? null : MomentHelper.deltaFromMoment(object.getEndDate(), 1, ChronoUnit.MONTHS);
    super.state(object.getEndDate() == null || MomentHelper.isBefore(object.getStartDate(), minimumDeadline), "startDate", "sponsor.sponsorship.form.error.too-close-moment");
}

if (!super.getBuffer().getErrors().hasErrors("endDate")) {
    Date maximumDeadline;

    super.state(MomentHelper.isAfterOrEqual(object.getEndDate(), object.getMoment()), "endDate", "sponsor.sponsorship.form.error.too-close-moment");

    maximumDeadline = object.getStartDate() == null ? null : MomentHelper.deltaFromMoment(object.getStartDate(), 1, ChronoUnit.MONTHS);
    super.state(object.getStartDate() == null || MomentHelper.isAfter(object.getEndDate(), maximumDeadline), "endDate", "sponsor.sponsorship.form.error.too-close-moment");
}
```

No se debería de permitir publicar una factura si su cantidad fuera superior a la de su patrocinio correspondiente:

Esto no está implementado como se pide. Si a la hora de publicar una invoice, la cantidad fuera superior a la del patrocinio, debería de mostrarse un error de validación, pero como no está implementado como se pide, siempre lo permite.

Para solucionar este problema, se ha creado una restricción en la validación de la clase “SponsorInvoicePublishService”, para que en el caso de que la cantidad de las facturas fuera superior a la de su patrocinio correspondiente, se mostrase un mensaje de error que impidiese publicar cualquiera de estas invoices hasta que se cumpliera la propiedad:

```
if (!super.getBuffer().getErrors().hasErrors("quantity")) {
    double before;
    double res;
    Collection<Invoice> invoices;

    if (object.getTax() != null) {
        invoices = this.repository.findManyInvoicesBySponsorshipId(object.getSponsorship().getId());
        before = this.repository.findOneInvoiceById(object.getId()).totalAmount().getAmount();
        res = object.totalAmount().getAmount() - before;
        for (Invoice i : invoices)
            res += i.totalAmount().getAmount();
    } else
        res = -10000000000.00;

    super.state(object.getQuantity().getAmount() > 0, "quantity", "sponsor.invoice.form.error.negative-salary");
    super.state(Arrays.asList(this.repository.findAcceptedCurrencies().split(",")).contains(object.getQuantity().getCurrency()), "quantity", "
    super.state(res <= object.getSponsorship().getAmount().getAmount(), "quantity", "sponsor.invoice.form.error.bad-total-amount");
}
```

11. Presupuesto actualizado

Tarea	Tiempo estimado (en horas)	Tiempo dedicado(en horas)	Coste estimado	Coste real	Rol
Task 6	12	18,32	240	366,4	Desarrollador
Task 7	9	9,72	180	194,4	Desarrollador
Task 8	5	5,26	100	105,2	Desarrollador
Task 17	3	1	60	20,00	Desarrollador
Task 18	1	0,26	20	5,20	Desarrollador
Task 19	4	1,2	80	24,00	Desarrollador
Task 20	1,5	0,5	30	10,00	Project Manager
Task 21	1,5	1,3	30	26	Project Manager
Task 22	3	1	60	20,00	Project Manager

Coste total estimado: 800 €

Coste total real: 771,4 €

Esta vez el coste se ajusta mucho más al estimado, debido a que en las tareas 6, 7 y 21 ha aumentado el tiempo. Las tareas 21 y 7 tienen un mejor ajuste que la vez anterior, acercándose más al estimado. Sin embargo, la tarea 6 se ha alejado mucho más del coste estimado, haciendo que el coste real sea mucho mayor.