



Facultade de Informática

UNIVERSIDADE DA CORUÑA

TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN ENXEÑARÍA DO SOFTWARE

Aplicación web para a xestión da formación bonificada nas empresas

Estudante: Martín Caamaño Rodríguez
Dirección: Sonia María Suárez Garaboa
Roberto García Pazos

A Coruña, setembro de 2021.

A miña avoa Carmen, sei que estarías moi orgullosa de min.

Agradecementos

En primeiro lugar, dar as grazas a Sonia e a Roberto, por todo o tempo, axuda e dedicación aportada, sen eles non sería posible este proxecto.

Tamén agradecer a meus pais, avós e achegados todo o cariño e confianza que me deron sempre e que tanto me serviu para seguir adiante en momentos difíciles.

Gracias a Ana por acompañarme dende o primeiro momento ata o último, sendo un pilar fundamental para chegar ata aquí.

Quero agradecerlle a Tomás todas as horas de estudo compartidas, todos os momentos de felicidade e por ser capaz de animarme cando as cousas se complican, teño claro que eres un amigo para a vida.

Agradecer, tamén, o apoio de todos os meus amigos, que estiveron sempre ó meu lado, nos bos e nos malos momentos, valedes ouro.

Por último, quero dar as grazas a todos os compañeiros e amigos cos que tiveron o pracer de traballar, estudar e desfrutar durante estes anos de carreira, sen vós non sería o mesmo.

Resumo

A formación continua das persoas, unha vez comezan a desenvolver a súa vida laboral, é un proceso moi importante para seguir aumentando as súas capacidades e aptitudes. Dita formación supón un custo para as empresas que queiran formar ós seus traballadores, pero existe a [FUNDAE](#) que bonifica a aquelas empresas que realizan estas accións, á cal hai que comunicarlle toda a información que precisa mediante uns formularios que presenta na súa páxina web ou por medio de entidades externas, provocando neste último caso unha perda do control do proceso co seu consecuente custo.

Para axudar a facilitar esta comunicación, crearase unha aplicación web que permita rexistrar usuarios asociados a empresas e posibilitar que estes empreguen a aplicación para que lles sexa máis doado solicitar axudas para a formación dos seus traballadores, podendo rexistrar nesta aplicación información sobre os centros de formación, os formadores, as accións formativas e os grupos formativos. A aplicación tamén permitirá que os seus usuarios poidan xerar ficheiros XML para comunicar a información das súas formacións á plataforma da [FUNDAE](#) que concede as axudas.

Abstract

Continuous training of people, when they have already started their working life, is a very important process to continue increasing their own capabilities and skills. This training represents a cost for companies that want to train their workers, but there is the [FUNDAE](#) that rewards those companies that carry out these actions. It's necessary to communicate to the [FUNDAE](#) all the information that it requests through the forms on its website or by external entities, causing in this case a loss of process control and an additional cost.

To help make this happen, a web application will be created that allows to register companies as users and that enables these users to use the application to make it easier for them to request financial aid for the training of their employees. This application also allows information about the training centers, the trainers, the formative actions and the formative groups. The application will also allow users to generate XML files to communicate their training information to the [FUNDAE](#) platform, the granting authority aid.

Palabras chave:

- Aplicación Web
- Formación
- Empleado
- Empresa
- Angular

Keywords:

- Web Application
- Training
- Employee
- Company
- Angular

Índice Xeral

1	Introdución	1
1.1	Motivación	1
1.2	Obxectivos	2
1.3	Organización	3
2	Análise de requisitos	5
2.1	Obtención de Requisitos	5
2.2	Actores	6
2.3	Requisitos funcionais	6
2.4	Requisitos non funcionais	8
3	Metodoloxía	9
3.1	Scrum	9
3.2	Adaptación	10
4	Aspectos tecnolóxicos	11
4.1	Linguaxes de programación	11
4.1.1	Java	11
4.1.2	TypeScript	11
4.1.3	HTML	11
4.1.4	SCSS	11
4.1.5	LaTeX	12
4.2	Entorno de desenvolvemento	12
4.2.1	Visual Studio Code	12
4.2.2	Google Chrome	12
4.2.3	Equipo	12
4.3	Tecnoloxías	12
4.3.1	Gradle	12

4.3.2	Spring	12
4.3.3	Hibernate	13
4.3.4	Node.js	13
4.3.5	Angular	13
4.3.6	REST	13
4.3.7	HTTP	13
4.4	Base de datos	13
4.4.1	PostgreSQL	13
4.4.2	PgAdmin	13
4.5	Ferramentas para o control de versións	14
4.5.1	Git	14
4.5.2	GitLab	14
4.5.3	GitKraken	14
4.6	Outros	14
4.6.1	Overleaf	14
4.6.2	WhiteStarUML	14
4.6.3	Microsoft Teams	14
5	Planificación	15
5.1	Sprint 0	15
5.2	Sprint 1	15
5.3	Sprint 2	16
5.4	Sprint 3	16
5.5	Sprint 4	16
5.6	Sprint 5	16
5.7	Sprint 6	16
5.8	Sprint 7	16
5.9	Sprint 8	17
5.10	Sprint 9	17
5.11	Sprint 10	17
5.12	Sprint 11	17
5.13	Custo	17
6	Desenvolvemento	19
6.1	Deseño	19
6.1.1	Arquitectura	19
6.1.2	Patróns de deseño	19
6.2	Sprint Xeral	20

6.2.1	Modelo de datos	20
6.2.2	Backend	20
6.2.3	Frontend	21
6.3	Sprint 0	22
6.3.1	Modelo de datos	22
6.3.2	Backend	22
6.3.3	Frontend	23
6.4	Sprint 1	23
6.4.1	Modelo de datos	23
6.4.2	Backend	23
6.4.3	Frontend	24
6.5	Sprint 2	25
6.5.1	Modelo de datos	25
6.5.2	Backend	25
6.5.3	Frontend	26
6.6	Sprint 3	27
6.6.1	Modelo de datos	27
6.6.2	Backend	27
6.6.3	Frontend	27
6.7	Sprint 4	27
6.7.1	Modelo de datos	27
6.7.2	Backend	28
6.7.3	Frontend	28
6.8	Sprint 5	28
6.8.1	Modelo de datos	28
6.8.2	Backend	29
6.8.3	Frontend	29
6.9	Sprint 6	29
6.9.1	Modelo de datos	29
6.9.2	Backend	30
6.9.3	Frontend	30
6.10	Sprint 7	31
6.10.1	Modelo de datos	31
6.10.2	Backend	31
6.10.3	Frontend	33
6.11	Sprint 8	33
6.11.1	Modelo de datos	33

6.11.2	Backend	33
6.11.3	Frontend	34
6.12	Sprint 9	34
6.12.1	Modelo de datos	34
6.12.2	Backend	34
6.12.3	Frontend	35
6.13	Sprint 10	35
6.13.1	Modelo de datos	35
6.13.2	Backend	35
6.13.3	Frontend	37
6.14	Sprint 11	37
6.14.1	Modelo de datos	37
6.14.2	Backend	37
6.14.3	Frontend	38
7	Probas	39
7.1	Inicialización de datos	39
7.2	Probas de sistema	39
7.3	Validación de ficheiros xerados	40
7.4	Probas de aceptación	40
8	Conclusións e Liñas Futuras	41
8.1	Conclusións	41
8.2	Liñas futuras	41
A	Manual de usuario	45
A.1	Inicio de sesión	45
A.2	Menú administrador	46
A.3	Listado de usuarios	46
A.4	Formulario de usuario	47
A.5	Listado de empresas bonificadas	48
A.6	Formulario de empresa bonificada	48
A.7	Menú xestor	49
A.8	Listado de empregados	50
A.9	Formulario de empregados	50
A.10	Listado de centros de formación	51
A.11	Formulario de centros de formación	52
A.12	Listado de formadores	52

ÍNDICE XERAL

A.13 Formulario de formador	53
A.14 Listado de accións formativas	54
A.15 Formulario de acción formativa	54
A.16 Listado de grupos formativos	55
A.17 Formulario de grupo formativo	56
Relación de Acrónimos	59
Bibliografía	61

Índice de Figuras

1.1	Forma de traballo de Ebo Gestión cás empresas de formación	2
1.2	Forma de traballo de Ebo Gestión cás empresas bonificadas	2
2.1	Diagrama do dominio obtido durante as reunións co cliente	6
3.1	Procedemento do ciclo de desenvolvemento de Scrum	10
6.1	Entidade para Empresa bonificada	23
6.2	Diagrama de clases Backend	24
6.3	Entidade para Empregado	25
6.4	Entidade para Centro de formación	27
6.5	Entidade para Formador	28
6.6	Entidade para Acción formativa	29
6.7	Entidade para Grupo formativo	30
6.8	Cabeceira do ficheiro Excel para engadir empregados	31
A.1	Pantalla de inicio de sesión	45
A.2	Pantalla de menú do usuario Administrador	46
A.3	Pantalla de listado de Usuarios	47
A.4	Pantalla de formulario de Usuario	47
A.5	Pantalla de listado de Empresas bonificadas	48
A.6	Pantalla de formulario de Empresa bonificada	49
A.7	Pantalla de menú do usuario Xestor	49
A.8	Pantalla de listado de Empregados	50
A.9	Pantalla de formulario de Empregado	51
A.10	Pantalla de listado de Centros de formación	51
A.11	Pantalla de formulario de Centro de formación	52
A.12	Pantalla de listado de Formadores	53

A.13 Pantalla de formulario de Formadores	53
A.14 Pantalla de listado de Accions formativas	54
A.15 Pantalla de formulario de Acción formativa	55
A.16 Pantalla de listado de Grupos formativos	56
A.17 Pantalla de formulario para a primeira pestana de Grupo formativo	57
A.18 Pantalla de formulario para a segunda pestana de Grupo formativo	58
A.19 Pantalla de formulario para a terceira pestana de Grupo formativo	58

Índice de Táboas

5.1	Salarios	18
5.2	Custo	18

Introdución

1.1 Motivación

A idea de desenvolver este proxecto nace da necesidade de facilitar que as empresas que queiran formar ós seus traballadores poidan realizar o proceso de comunicación coa aplicación web da [FUNDAE](#) para recibir axudas de maneira máis sinxela. É importante facilitar este proceso xa que a formación continua das persoas enriquece as súas propias capacidades e aptitudes, sendo así capaces de xerar un maior desenvolvemento profesional, mellorar a súa produtividade, optimizar a xestión de tarefas e aumentar o valor engadido da empresa na que traballen. Ademais, se ás empresas lles resulta sinxelo e económico mellorar a formación dos seus traballadores dende o momento no que se incorporan á vida laboral, tamén lles será máis doado ós empregados adquirir eses novos conceptos que reciban nas formacións, posto que xa están acostumbrados a estudar e aprender.

Actualmente, a forma directa empregada para comunicarse ca [FUNDAE](#) é mediante os formularios que presenta a propia aplicación da fundación. Mediante ditos formularios, usuarios autorizados polas empresas poden informar sobre os seus empregados, as accións formativas que presentan na empresa e os grupos formativos que desenvolven ditas accións, sendo un proceso tedioso, xa que deben completar toda a información cada vez que pretendan notificar algunha destas actividades.

Aínda que non se atoparon aplicacións que realicen esta comunicación, si existen empresas externas que se encargan de xestionar ditos trámites para que as empresas bonificadas non teñan que ser as encargadas de comunicarse directamente ca [FUNDAE](#). Isto supón depender de terceiros para realizar estes trámites, co conseguinte custo e perda de control do proceso. Un exemplo deste tipo de empresas é **Ebo Gestión**, sendo unha compañía xestora recoñecida pola [FUNDAE](#) que se encarga de tramitar o proceso de informar sobre as formacións que reci-

ben as empresas que queren ser bonificadas e tamén ofrecen a posibilidade de que os centros de formación externalicen o servizo de xestión de bonificacións para os seus propios clientes, seguindo o proceso que se mostra na figura 1.1.

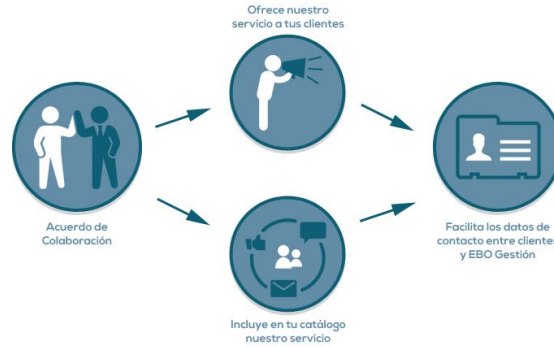


Figura 1.1: Forma de traballo de Ebo Gestión cás empresas de formación

A forma que empregan para tramitar as formacións das empresas bonificadas, como se aprecia na figura 1.2, é mediante un contrato de adherencia e logo, ditas empresas comunican as accións formativas que van realizar, xunto con toda a información necesaria.



Figura 1.2: Forma de traballo de Ebo Gestión cás empresas bonificadas

Empregando esta nova aplicación desenvolta no proxecto, as empresas bonificadas poderán realizar elas mesmas o traballo de comunicar a información á **FUNDAE** de maneira sinxela, aproveitando os datos que xa teñen rexistrados sobre os seus empregados, os centros formativos e os formadores.

1.2 Obxectivos

O obxectivo principal será desenvolver unha aplicación web que permita realizar certas funcionalidades, dependendo do tipo de usuario que a estea empregando. En primeiro lugar, teremos un usuario administrador que poderá rexistrar e modificar datos de empresas dentro da aplicación e, ó mesmo tempo, poderá dar de alta e modificar datos de usuarios asociados a

unha destas empresas. Por outro lado, estes usuarios creados polos administradores, poderán crear e modificar datos de centros formativos, formadores, empregados da empresa, accións formativas e grupos formativos. Ademais, tamén poderán xerar documentos XML para comunicarse coa aplicación da [FUNDAE](#) e dar de alta os empregados da empresa, as accións formativas e o inicio e o fin dos grupos formativos. Finalmente, contarán cá opción de xerar documentos PDF para realizar os partes de asistencia sobre os grupos formativos.

1.3 Organización

Esta memoria está estruturada en oito capítulos.

- **Capítulo 1:** Capítulo no que se fai unha introdución para poñer en contexto o proxecto, así como a motivación e os obxectivos.
- **Capítulo 2:** Capítulo no cal se describe como se realizou a análise e a extracción de requisitos, indicando os actores e funcionalidades obtidas como resultado de ditas accións.
- **Capítulo 3:** Capítulo empregado para indicar a metodoloxía empregada para a realización deste proxecto, indicando os motivos da elección, definindo as súas características e as adaptacións realizadas.
- **Capítulo 4:** Capítulo no que se recollen as ferramentas e tecnoloxías empregadas no proxecto.
- **Capítulo 5:** Capítulo no que se expón a planificación realizada para o desenvolvemento do proxecto e o custo que suporía dito desenvolvemento segundo a planificación.
- **Capítulo 6:** Capítulo empregado para describir o desenvolvemento do proxecto, incluíndo o deseño e a implementación.
- **Capítulo 7:** Capítulo no cal se explican as probas que foron feitas na aplicación e os resultados obtidos.
- **Capítulo 8:** Capítulo no que se recollen as conclusións obtidas tras a realización do proxecto e as futuras liñas de traballo.

Análise de requisitos

2.1 Obtención de Requisitos

Os requisitos foron obtidos mediante unha serie de reunións co cliente, representado polo director profesional do proxecto, para que este explicase cales eran os obxectivos que debería cumprir a aplicación. Estas reunións fixéronse mediante Microsoft Teams (mencionado no apartado 4.6.3), onde o cliente tivo a posibilidade de compartir a pantalla do seu equipo para, desta maneira, mostrar como era o funcionamento da páxina da [FUNDAE](#) e poder obter de aí toda a funcionalidade que se necesitaría. Para ter un maior entendemento co cliente, tras a realización das reunións foise desenvolvendo un diagrama do dominio da aplicación (ver figura 2.1, dado que o cliente esta familiarizado con este tipo de representación, para asegurarse de que ambas partes estaban entendendo da mesma maneira os requisitos obtidos. Ademais destas reunións, tamén se procurou información sobre como deberían ser os ficheiros [XML](#) para comunicarse cá aplicación da [FUNDAE](#). A propia páxina da fundación proporciona esquemas para ditos ficheiros e mostra algúns exemplos. Dita procura resultou de utilidade para engadir algúns campos adicionais ás entidades que non foran obtidos antes.

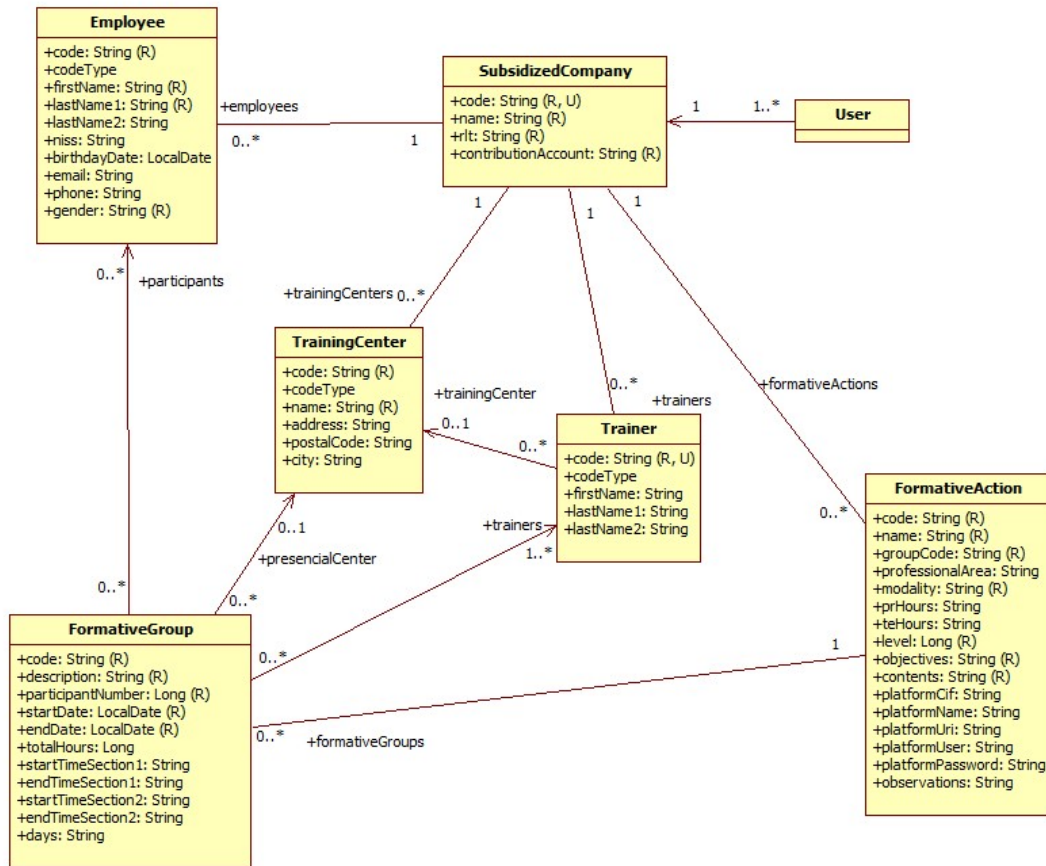


Figura 2.1: Diagrama do dominio obtido durante as reunións co cliente

2.2 Actores

Durante a análise de requisitos obtivéronse dous tipos de actores que van interactuar cá aplicación:

- Administrador: persoa encargada de rexistrar empresas bonificadas e crear usuarios xestores asociados a ditas empresas.
- Xestor: persoa encargada da xestión dunha empresa bonificada.

2.3 Requisitos funcionais

Para esta aplicación obtivéronse dous módulos de funcionalidades, asociados a cadanseu actor:

- De Administración: crear e modificar empresas bonificadas e usuarios.

- Para as empresas bonificadas rexistrarse o CIF, o nome e a conta de cotización da empresa.
- Para os usuarios rexistrarse un login, un correo electrónico, o nome, os apelidos e o rol, podendo ser este de administrador ou xestor. En caso de que sexa xestor, haberá que informar tamén a empresa que xestionará.
- De Xestión: crear e modificar empregados, centros formativos, formadores, accións formativas e grupos formativos manualmente; rexistrar empregados mediante un ficheiro Excel no que se recolla información de ditos traballadores; xerar ficheiros XML para dar de alta participantes, accións formativas e grupos formativos; e xerar documentos PDF para os partes de asistencia dos grupos formativos.
 - Para os empregados rexistrarse o número do documento que o identifique xunto co tipo de documento, o nome, os apelidos, o número de identificación da Seguridade Social, a data de nacemento, un correo electrónico, un número telefónico, o sexo, a categoría profesional, o grupo de cotización e o nivel de estudos.
 - Para os centros de formación rexistrarse o número do documento que o identifique xunto co tipo de documento, o nome, a dirección, o código postal e a cidade.
 - Para os formadores rexistrarse o número do documento que o identifique xunto co tipo de documento, o nome, os apelidos, un número telefónico, un correo electrónico e o centro formativo ó que pertenza.
 - Para cada acción formativa rexistrarse un código interno que a identifique, un nome, o código do grupo ó que pertence a acción, o tipo de modalidade (centrándose na modalidade presencial), o número de horas presenciais, o nivel da formación, os obxectivos, o contido e observacións sobre a acción formativa.
 - Para cada grupo formativo haberá tres pestanas para realizar o rexistro. Na primeira delas rexistrarse un código interno que o identifique, a acción formativa á que se asocia, a data de comezo e de fin do grupo formativo, unha descrición, a persoa a cargo do grupo formativo xunto cun número telefónico de contacto, o centro de formación que impartirá a formación xunto cá información do espazo donde terá lugar e os formadores coas horas que realizará cada un deles. Ademais, tamén se informará sobre o horario da formación, indicando os días da semana e os períodos horarios, e finalmente indicarse a información sobre a aula virtual que se empregue. Na segunda pestana, rexistraranse os empregados que participan no grupo formativo. Na última pestana informarse sobre os custos da formación, indicando os importes directos, indirectos e salariais, xunto cas bonificacións mensuais recollendo os meses e as cuantías correspondentes.

- Para importar na aplicación empregados dende un ficheiro Excel, crearase un modelo a seguir con este formato, para que todos os documentos presenten o mesmo estilo e a información de cada empregado se recolla na columna correspondente.
- Para xerar os ficheiros XML empregarase un checkbox para indicar que empregados, accións ou grupos formativos se queren engadir ó documento e un botón para crear o documento (no caso dos grupos formativos haberá dous para indicar se se quere xerar o ficheiro de inicio ou de fin dos grupos).
- Para xerar ficheiros PDF empregarase un botón dentro da visualización do detalle dun grupo formativo para crear o documento.

2.4 **Requisitos non funcionais**

Os requisitos non funcionais obtidos para este proxecto foron os seguintes:

- Presentar unha interface amigable para o usuario, cunha aparencia sinxela e seria.
- Usabilidade da aplicación, sen que as tarefas a realizar polo usuario sexan moi longas nin de moitos pasos.
- Amosar a información de forma clara.
- Por seguridade, non permitir que usuarios xestores dunha empresa bonificada poidan ler ou escribir información pertencente a unha empresa distinta á súa.

Metodoloxía

A METODOLOXÍA é un aspecto importante no desenvolvemento do software posto que define como vai ser o ciclo de vida do produto. Por este motivo, é importante realizar unha correcta elección tendo en conta as características do proxecto a desenvolver e a forma de traballo que se queira adoptar. Actualmente, no desenvolvemento software as metodoloxías máis empregadas son as áxiles, debido a que presentan unha maior adaptación en entornos nos que se traballa con requisitos inestables ou pouco definidos, sendo a máis utilizada Scrum. Decídese seguir esta metodoloxía debido a que os requisitos non están totalmente definidos no inicio do proxecto e a que o cliente pode engadir, modificar ou eliminar requirimentos a medida que se avanza no desenvolvemento do produto. Debido a esta condición cambiante dos requisitos do sistema, descártase o uso de metodoloxías predictivas.

3.1 Scrum

A metodoloxía empregada neste proxecto é Scrum [1], que consiste nunha metodoloxía áxil de desenvolvemento, moi útil en entornos e contextos onde se traballan con requisitos inestables ou pouco definidos e que requiren rapidez e flexibilidade. Tamén é unha metodoloxía incremental baseada en iteracións (denominadas Sprints e que adoitan ter unha duración de 15 días) e revisións, o que fai que a forma de traballo sexa adaptativa. Scrum establece un Product Backlog, que consiste nunha lista de requisitos que se empregarán para realizar a planificación dos Sprints, xunto cas tarefas que se realizarán en cada un deles. O ciclo de desenvolvemento desta metodoloxía vese representado na figura 3.1. Ademais, existen varios tipos de roles:

- **Equipo de desenvolvemento:** encargado de implementar as funcionalidades do produto recollidas nos Sprints. Terán que realizar reunións diarias para comentar o traballo realizado despois da última reunión, as tarefas que farán no propio día e os problemas cos que se atoparon.

- **Product Owner:** encargado de representar os intereses do cliente.
- **Scrum Master:** encargado de protexer ó equipo de desenvolvemento e solucionar obstáculos que poidan aparecer durante o desenvolvemento. Tamén é o responsable de facer que a metodoloxía sexa entendida e aplicada correctamente polo resto de persoas que compoñen o proxecto.

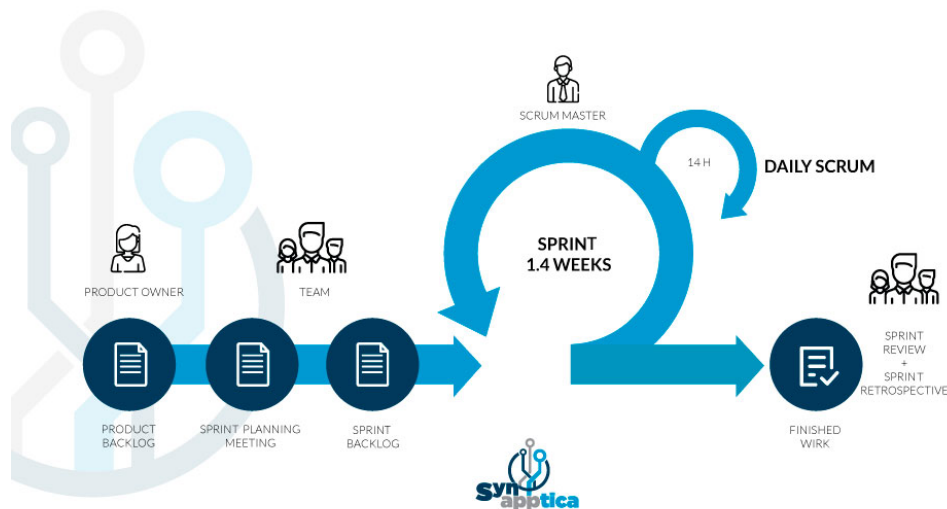


Figura 3.1: Procedemento do ciclo de desenvolvemento de Scrum

3.2 Adaptación

Neste proxecto realizouse unha adaptación da metodoloxía Scrum, dado que o equipo de desenvolvemento está formado por unha soa persoa. Algunhas das adaptacións levadas a cabo foron:

- Os Sprints presentarán unha duración aproximada de 7 días.
- O equipo de desenvolvemento será o estudante.
- O Product Owner será o director profesional do proxecto.
- O Scrum Master será a directora académica do proxecto.
- Suprímense as comunicacións periódicas entre os membros do equipo de desenvolvemento, dado que está formado por unha soa persoa.

Aspectos tecnológicos

4.1 Linguaxes de programación

4.1.1 Java

Java [2] é unha linguaxe de programación de alto nivel orientada a obxectos e baseada en clases. Proporciona unha xestión automática da memoria empregando un recolector de lixo que libera ó programador da solicitude a da liberación da memoria, facendo que mellore a produtividade e a ausencia de erros de seguridade. Foi a linguaxe empregada para o desenvolvemento da parte Backend deste proxecto.

4.1.2 TypeScript

TypeScript [3] é unha linguaxe de programación baseada en JavaScript, incorporando unha sintaxis adicional máis amigable para o programador, dado que se poden detectar os erros durante a programación. O código xerado con esta linguaxe transfórmase en JavaScript para poder ser executado nos mesmos entornos (Node.js, mencionado no apartado 4.3.4, por exemplo). Foi empregada para programar o Frontend desde proxecto.

4.1.3 HTML

HTML [4] é unha linguaxe de programación de marcado que serve para definir o contido das páxinas web. Componse en base a etiquetas, as cales se consegue expresar a información presente no contido da páxina. Empregada para crear as páxinas que compoñen a aplicación.

4.1.4 SCSS

SCSS [5] é unha linguaxe de folia de estilos baseada en CSS pero que presenta máis características, como son as variables e as regras anidadas. Empregada para crear os estilos da aplicación.

4.1.5 LaTeX

LaTeX [6] é unha linguaxe para a preparación de documentos que permite obter unha composición tipográfica de alta calidade. Foi empregado para realizar a memoria deste proxecto.

4.2 Entorno de desenvolvemento

4.2.1 Visual Studio Code

Visual Studio Code [7] é un IDE que presenta un editor de código fonte lixeiro pero potente que se executa no escritorio e está dispoñible para Windows, macOS e Linux. Conta con soporte integrado para JavaScript, TypeScript e Node.js e ten un rico ecosistema de extensións para outras linguaxes (como C ++, C #, Java, Python, PHP, Go) e runtimes (como .NET y Unity). Foi empregado para a realización do código deste proxecto, así como, para executar a aplicación.

4.2.2 Google Chrome

Google Chrome [8] foi o navegador web empregado durante a realización deste proxecto debido a que permite inspeccionar sitios web, o que foi de utilidade para facer comprobacións das peticións que se envían na aplicación, comprobar e facer cambios en quente do [HTML](#) e ver os mensaxes que se devolven na consola.

4.2.3 Equipo

O equipo empregado para desenvolver este proxecto presenta o sistema operativo Windows 10, un procesador Intel Core i7, unha memoria RAM de 16GB e un disco [SSD](#) de 128GB.

4.3 Tecnoloxías

4.3.1 Gradle

Gradle [9] é unha ferramenta de automatización de compilación de código aberto centrada na flexibilidade e no rendemento. Foi empregado no proxecto como ferramenta de construción do código para o backend.

4.3.2 Spring

Spring [10] é a ferramenta empregada para interceptar as invocacións dos métodos transaccionais e, ademais, tamén se encarga da transacción e da inxección de dependencias.

4.3.3 Hibernate

Hibernate [11] é a ferramenta empregada para facer o mapeo dos datos entre a aplicación e a base de datos relacional, dado que facilita a implementación dos DAOs.

4.3.4 Node.js

Node.js [12] é un entorno de execución de JavaScript orientado para eventos asíncronos e deseñado para crear aplicacións escalables. Foi empregado neste proxecto para a execución do Frontend.

4.3.5 Angular

Angular [13] é un framework de deseño de aplicacións para crear aplicacións web móbiles e de escritorio eficientes e sofisticadas. Foi empregado para realizar a parte Frontend deste proxecto.

4.3.6 REST

REST [14] é un servizo web, que se invoca empregando o protocolo HTTP, pensado para enviar datos en formato de representación XML e JSON. Foi empregado neste proxecto, dado que se adapta á arquitectura da aplicación.

4.3.7 HTTP

HTTP [15] é un protocolo de comunicación que segue un esquema de petición/resposta. Foi empregado neste proxecto para intercambiar información entre o cliente e o servidor.

4.4 Base de datos

4.4.1 PostgreSQL

PostgreSQL [16] é un sistema de base de datos relacional de código aberto que estende e usa a linguaxe SQL. Foi empregado neste proxecto para almacenar as entidades que compoñen a aplicación.

4.4.2 PgAdmin

PgAdmin [17] é unha ferramenta de xestión para PostgreSQL e bases de datos relacionais derivadas como EDB Advanced Server de EnterpriseDB. Pode executarse como una aplicación web ou de escritorio. Foi empregado para facer comprobacións de se os datos se estaban

gardando correctamente tras realizar algunha operación de escritura e para revisar se se estaba lendo a información correctamente.

4.5 Ferramentas para o control de versións

4.5.1 Git

Git [18] é un sistema de control de versións distribuído e eficiente que permite crear e fusionar ramas de traballo con distintas versións do produto, útil cando se traballa con un grupo de varias persoas.

4.5.2 GitLab

GitLab [19] é un servizo web de control de versións baseado en Git. Foi empregado creando un repositorio para este proxecto, no cal se crearon as tarefas correspondentes a cada unha das funcionalidades extraídas e cadansúa rama. Isto fíxose ca idea de poder moverse entre ramas e así ter a posibilidade de facer varias funcionalidades a un mesmo tempo, é dicir, sen necesidade de rematar unha para poder comezar outra.

4.5.3 GitKraken

GitKraken [20] é unha ferramenta gráfica que facilita o uso de Git, permitindo confirmar, subir e baixar cambios de forma sinxela e moverse entre ramas dunha forma visual e intuitiva.

4.6 Outros

4.6.1 Overleaf

Overleaf [21] é un editor de texto LaTeX online e gratuito. Foi empregado para realizar a memoria deste proxecto.

4.6.2 WhiteStarUML

WhiteStarUML [22] é unha ferramenta que permite crear diagramas UML e poder descargalos en formato XML. Foi empregado para crear o diagrama do dominio deste proxecto.

4.6.3 Microsoft Teams

Microsoft Teams [23] é unha plataforma de comunicación que permite intercambiar mensaxes de forma directa entre persoas, empregada neste proxecto para comunicar o equipo de desenvolvemento co Product Owner e co Scrum Master.

Planificación

PLANIFICAR o desenvolvemento do proxecto é unha tarefa importante para organizar o tempo de dedicación e facer unha estimación sobre o tempo que requirirá dito desenvolvemento. Seguindo a metodoloxía Scrum, planifícase cada Sprint en función dos requirimentos obtidos durante a fase de análise de requisitos. Cada un destes Sprints recollerá unha funcionalidade concreta ou un conxunto de funcionalidades das desenvoltas no proxecto, de xeito que presenten unha complexidade similar e supoñan un tempo de dedicación moi semellante. Tamén se realiza unha estimación do posible custo que supón o desenvolvemento do proxecto, tendo en conta o planificado.

5.1 Sprint 0

Este Sprint inicial será o encargado de recoller todas aquelas accións que servirán para comezar o desenvolvemento da aplicación. Isto implica dedicar tempo a xerar o proxecto empregando o xerador de proxectos da empresa, o cal inclúe no produto obtido o propio framework da organización. Tamén se configurará dito proxecto para adaptalo ás necesidades que debemos cubrir.

5.2 Sprint 1

Neste primeiro Sprint de desenvolvemento agrúpanse aquelas tarefas propias do usuario administrador, é dicir, listar, rexistrar e modificar usuarios e empresas bonificadas. Estas dúas funcionalidades recóllense nun mesmo Sprint dado que, para poder crear un empregado xestor é necesario poder rexistrar empresas bonificadas previamente e para crear unha empresa bonificada precisase rexistar antes un usuario administrador que poida facelo.

5.3 Sprint 2

Neste segundo Sprint comezase a primeira tarefa do módulo xestor, que consiste no listado, rexistro e modificación de empregados, o que implica ter que configurar tamén dito módulo, para diferencialo do módulo de administración e que cada tipo de usuario só teña acceso ás funcionalidades que lle corresponden.

5.4 Sprint 3

Neste terceiro Sprint realízase a funcionalidade de listar, rexistrar e modificar os centros de formación, que serán as entidades encargadas de impartir as formacións e agrupar ós formadores.

5.5 Sprint 4

Neste cuarto Sprint realízase a funcionalidade de listar, rexistrar e modificar os formadores asociados ós centros de formación previamente citados.

5.6 Sprint 5

Neste quinto Sprint desenvólvense as tarefas de listar, rexistrar e modificar as accións formativas, que serán as entidades que agrupen e definan a formación que se vai impartir nos grupos formativos.

5.7 Sprint 6

Este sexto Sprint recolle a funcionalidade de listar, rexistrar e modificar os grupos formativos, que é a entidade máis complexa que compón este proxecto, debido a que recolle unha lista de empregados que participan na formación, un centro de formación encargado de impartir o curso e unha lista de formadores que serán os que teñan que explicar os contidos. Tamén se indicará nesta entidade os costes directos, indirectos e salariais do grupo formativo, xunto cunha lista de bonificacións mensuais onde se recollerá o mes de cobro e o importe a recibir.

5.8 Sprint 7

Este sétimo Sprint é empregado para realizar a tarefa de exportar empregados dende un ficheiro Excel. Para realizar esta funcionalidade dedícase tempo a investigar como poder facer

dita exportación e a implementar a solución que mellor se adapte ós obxectivos do proxecto.

5.9 Sprint 8

Neste oitavo Sprint realízase a funcionalidade de xerar ficheiros [XML](#) para dar de alta os empregados da empresa, é dicir, os posibles participantes dos grupos formativos. É necesario procurar información para coñecer como se pode realizar a creación deste tipo de ficheiro a partir dunha lista de entidades e para saber como darlle o formato desexado.

5.10 Sprint 9

Este noveno Sprint dedícase a xerar os ficheiros [XML](#) para dar de alta as accións formativas. Empréganse os coñecementos obtidos no Sprint anterior para a xeración destes documentos.

5.11 Sprint 10

Este décimo Sprint dedícase a xeración dos ficheiros [XML](#) encargados de indicar o inicio e o fin dos grupos formativos, empregando os coñecementos obtidos no Sprint 9 para a xeración destes documentos. Agrúpanse estas dúas funcionalidades nun mesmo Sprint dado a similitude das tarefas a realizar e a que traballan sobre as mesmas entidades.

5.12 Sprint 11

Este undécimo Sprint dedícase a xeración dos documentos PDF para que as empresas bonificadas poidan crear partes e asistencia e, así, levar un control sobre que empregados están presentes nas accións formativas que rexistran.

5.13 Custo

Unha vez coñecido o número de Sprints planificados, realízase unha avaliación do posible custo que supón o desenvolvemento do proxecto. Para calcular ditos custos séguese o "XVII Convenio colectivo estatal de empresas de consultoría, y estudios de mercados y de la opinión pública" [24] publicado no [BOE](#), obtendo un salario anual para cada membro do equipo de desenvolvemento de 22.993,74€ (grupo C, nivel I) e para o Product Owner e o Scrum Master un salario anual de 25.000,00€ (grupo A). Con estes datos obtéñense os custos mostrados na táboa 5.1, tendo en conta que a xornada máxima anual é de 1.800 horas.

Rol	Número de persoas	Salario/Hora
Equipo de desenvolvemento	1	12,77€
Product Owner	1	13,89€
Scrum Master	1	13,89€

Táboa 5.1: Salarios

Coñecido o custo por hora de cada membro do proxecto, realízase o cálculo total do desenvolvemento, tendo en conta que se trata de 11 Sprints cunha duración media de 7 días, é dicir, 56 horas por Sprint para os membros do equipo de desenvolvemento. O Scrum Master estímase que realice 1 hora por Sprint e o Product Owner 2 horas por Sprint. O custo total é o que se mostra na táboa 5.2.

Rol	Número de Sprints	Horas/Sprint	Salario/Hora	Custo
Equipo de desenvolvemento	11	56	12,77€	7.866,32€
Product Owner	11	2	13,89€	305,58€
Scrum Master	11	1	13,89€	152,79€
Total				8.324,69€

Táboa 5.2: Custo

Desenvolvemento

6.1 Deseño

6.1.1 Arquitectura

Para o desenvolvemento deste proxecto séguese unha arquitectura formada por un servidor de datos (a base de datos), un servidor de aplicacións (correspondéndose co backend consituído polos controladores e o modelo) e un cliente (o frontend formado pola interface de usuario e acceso a servizos).

6.1.2 Patróns de deseño

DAO

Este patrón arquitectónico [25] é empregado para separar a lóxica de negocio da lóxica de acceso a datos, facendo, así, que os DAOs proporcionen todos os métodos que os servizos necesiten para realizar as funcións de escritura e lectura da base de datos. Isto provoca unha abstracción á hora de xestionar a persistencia das entidades da aplicación dende os servizos, facilitando a implementación das funcionalidades.

DTO

Este patrón de deseño [25] emprégase para transferir información de unha ou varias entidades entre o cliente e o servidor nun obxecto, podendo ser adaptado ó que o cliente necesita e deste xeito prescindir de comunicar información dende o servidor que non vai ser empregada no cliente.

Fachada

Este patrón estrutural [26] emprégase de xeito que os servizos proporcionan unha interface simple cos métodos que poden ser chamados por parte do cliente, ocultado a complexidade da que dispón a implementación das funcionalidades que ofrece dito servizo.

Modelo-Vista-Controlador

Este patrón arquitectónico [25] é empregado para separar a lóxica de negocio da interface de usuario. Dita separación prodúcese mediante un controlador (compoñente intermedio entre a vista e o modelo) que é o encargado de recoller a modo de petición as accións que o usuario realiza dente a vista e comunícalle ó modelo ditas accións para escribir ou ler datos e devolverlle un resultado ó usuario.

6.2 Sprint Xeral

Para desenvolver as funcionalidades obtidas na análise de requisitos seguindo a metodoloxía Scrum e planificación mencionada, aplícase en todos os Sprints unha organización similar, que consiste en deseñar o modelo de datos (no caso de que o haxa), desenvolver o Backend e finalmente o Frontend.

6.2.1 Modelo de datos

Crear unha nova táboa para almacenar os datos da entidade á que corresponde o Sprint, engadindo como chave primaria un ID único na táboa que será asignado por defecto seguindo unha secuencia incremental (comezando en 1 con incrementos de 1) cando esta sexa engadida á base de datos. A maiores do identificador, engádense os atributos propios da entidade, indicando se admite valores nulos, se ten que ser único e o tipo.

6.2.2 Backend

Crear un paquete que conteña os subpaquetes relativos ó dominio, DAOs, servizos e controladores da entidade principal á que corresponda o Sprint.

Dominio

O subpaquete dominio contén a clase Entidade, na cal se definen todos os atributos que conforman a entidade, xunto cun construtor valeiro, getters, setters, hashCode e equals.

DAO

O subpaquete DAO contén a interface do **DAO** xunto ca súa implementación para comunicarse ca base de datos e engádese un método para buscar e devolver de forma paxinada todas as entidades (facendo uso do framework Spring).

Servizo

O subpaquete Servizo contén a interface na que se especifican os métodos dos que se dispón e unha clase que implementa dita interface. O servizo conta con dúas funcións de lectura para buscar todas as entidades de forma paxinada e para buscar unha entidade concreta; e dúas funcións de escritura para crear e modificar unha entidade. Ademais, tamén dispón dun método privado para validar que cando se crea ou modifica unha entidade, os atributos que a identifiquen non estean presentes noutra entidade, lanzando unha excepción no caso de que xa exista, para evitar o gardado dos cambios. Tamén se definen os dous **DTOs** empregados no Sprint para enviar e recibir información da entidade. Un destes **DTOs** contén a información da entidade que se mostra no listado xunto co ID. Por outro lado, o segundo **DTO** contén a información que se mostra cando se ve o detalle e tamén será o que se reciba no servidor como resultado dunha petición de creación ou modificación.

Controlador

O subpaquete Controlador contén a clase do controlador REST, que conta con dúas funcións para peticións GET para os métodos de lectura mencionados; unha para a petición POST de creación, que contén no corpo da petición o **DTO** da entidade a crear; e outra para a petición PUT de modificación, formando, o ID da entidade a modificar, parte do path da petición e o corpo da petición tamén ten o **DTO** cos cambios.

6.2.3 Frontend

Crear un módulo Angular con enrutamento para a entidade principal do Sprint e permitir, así, a navegación entre varias páxinas dentro do módulo (páxina de listado e de detalle). Engádense a este módulo creado dous compoñentes de Angular para o listado das entidades e para o formulario dunha entidade concreta. Ambos compoñentes están formados por un ficheiro **HTML**, no que se describe a aparencia da páxina; un ficheiro **SCSS**, no que indica o estilo dalgún elemento concreto da páxina; e un ficheiro **TypeScript**, onde se recollen as funcións que recollen a lóxica da páxina e as comunicacións co servidor. No relativo ó compoñente de listado, o **HTML** conta cunha táboa que permite a paxinación para mostrar todas as entidades rexistradas. Dita táboa permite a ordenación por columnas de forma ascendente e descendente, enviando na petición que se lanza ó servidor o atributo polo que se quere ordenar e

o sentido da orde. Tamén conta cun botón para crear unha nova entidade, o cal redirixe ó usuario á pantalla de creación, correspondente ó segundo compoñente creado. Finalmente, se o usuario fai clic nunha fila do listado, redirixese mediante `routerLink` ó detalle da entidade. Por outro lado, no compoñente de detalle da entidade, móstranse os atributos da entidade nun formulario e engádese un botón para habilitar a edición. Sendo o contexto de edición ou de creación, engádese un botón para confirmar os cambios e outro para cancelar, controlando o botón de gardado para que só estea habilitado se se cumpren todas as condicións do formulario, é dicir, que estean informados os campos requeridos e que sexan válidos (que cumpran, se se especifica, un patrón concreto).

Ademais, no relativo ó frontend, engádese no menú da aplicación a opción para poder acceder á pantalla de listado da entidade, e, por consecuencia, a toda a funcionalidade da que dispón.

6.3 Sprint 0

Neste Sprint inicial comézase o desenvolvemento da aplicación, o que implica realizar un estudo sobre as ferramentas que se empregarán. Moitas destas ferramentas xa foran usadas nalgúns materias da titulación, como son Spring, Hibernate ou Java; pero outras, como PostgreSQL, Node.js, Angular, TypeScript ou SCSS, non se traballou con elas durante o grao e hai que realizar unha investigación para entender o seu funcionamento e poder empregarlas correctamente.

6.3.1 Modelo de datos

O xerador de proxectos da empresa empregado para iniciar esta aplicación crea unha táboa para os usuarios no esquema público de PostgreSQL. O primeiro que se fai é crear un esquema propio para o proxecto e engadir a citada táboa de usuarios, engadindo modificacións, posto que non se precisan todos os atributos dos que dispón por defecto.

6.3.2 Backend

A aplicación obtida do xerador de proxecto dispón de algunhas funcións relacionadas cos usuarios, como son o inicio e o peche de sesión, o rexistro e a modificación. Estas funcionalidades aparecen recollidas directamente nunha clase xeral. Por tanto, créase unha interface que especifique os métodos desenvolvidos nesa clase, e faise que dita clase xeral implemente a nova interface creada. Tamén se configura a entidade `User` para que se corresponda ca nova táboa de usuarios creada no esquema do proxecto.

6.3.3 Frontend

Realízanse cambios para indicar unha mensaxe de benvida ós usuarios e tamén o nome da aplicación para que se mostre no título da páxina no navegador.

6.4 Sprint 1

6.4.1 Modelo de datos

Créase unha táboa para almacenar información sobre as empresas bonificadas, a cal conta con ID, un código único que sempre deberá ser informado, correspondente ó CIF da empresa; a razón social da empresa, para que os usuarios poidan identificar facilmente cada organización; e a conta de cotización, dado que será preciso informala cando se comuniquen as accións formativas. A entidade resultante é a que se mostra na figura 6.1.

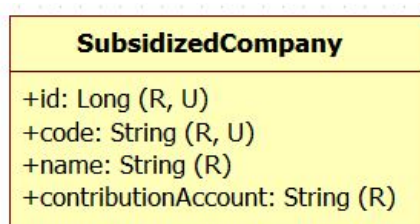


Figura 6.1: Entidade para Empresa bonificada

Unha vez creada a táboa para as empresas bonificadas, engádese un novo campo á táboa de usuarios mencionada no Sprint anterior (apartado 6.3). Este novo campo é o ID dunha empresa, empregándose como chave foránea entre ambas entidades, permitindo non ser informado dado que aqueles usuarios que dispoñan dun rol administrador non terán que ter unha organización asociada. Tamén se engade un atributo para indicar rol do usuario e diferenciar deste xeito se se trata dun usuario xestor ou administrador.

6.4.2 Backend

No desenvolvemento do backend neste Spring hai que resaltar que se engade á empresa bonificada na entidade do usuario e tamén se añade en ambos DTOs do usuario, empregando un DTO básico (BasicDTO) que unicamente informa sobre o ID e unha etiqueta de tipo String nomeada label. Nas entidades que se precise facer uso deste DTO haberá que definir un método getLabel() que devolva a información desexada (neste caso devólvese a razón social da empresa xunto co código entre parénteses) facendo posible, así, o mapeo da entidade a este BasicDTO. A continuación móstrase a función comentada:

```

1 public String getLabel() {
2     return this.name + " (" + this.code + ")";
3 }

```

Ademais, engádesse un método para obter todas as empresas bonificadas e devólvelas como unha lista destes DTOs básicos. Tamén se engade no controlador das empresas bonificadas a función para recoller a petición GET para este método. Na entidade do usuario engádesse tamén o rol que desempeña na aplicación. Na figura 6.2 móstrase a estrutura comentada neste backend, a cal será análoga nos seguintes Sprints.

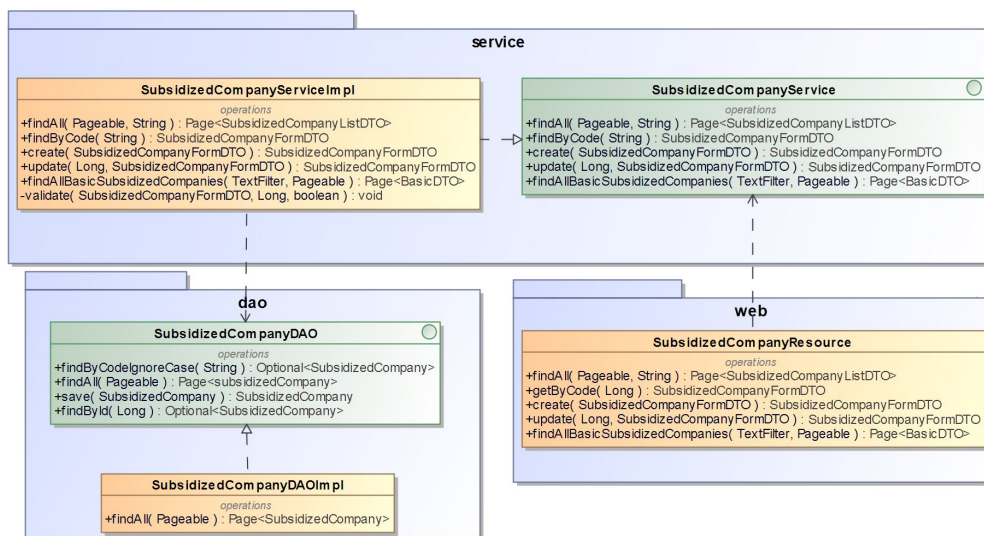


Figura 6.2: Diagrama de clases Backend

6.4.3 Frontend

No desenvolvemento do frontend deste primeiro Sprint créanse varios compoñentes xerais nun módulo compartido da aplicación. Un destes compoñentes é un botón xeral para a aplicación que informará unha mensaxe e un icono descritivo da acción que realice e se está habilitado ou non, sombreando a cor do botón en caso de que estea desactivado. Tamén se creou un compoñente para definir a estrutura que terán as cabeceiras de todas as páxinas de detalle dunha entidade, recollendo o título da páxina e os botón de modificación da entidade concreta que se estea mostrando. Finalmente, créase un compoñente que consiste nun input que dispón dun desplegable de entidades, é dicir, un selector, empregándose do seguinte xeito:

```

1 <pfobe-select label="Company" [required]="isRoleUser"
2     [disabled]="!isRoleUser" [options]="companySelector"
3     [(ngModel)]="entity.subsidizedCompany">

```

```
4 </pfobe-select>
```

Este último compoñente é empregado para seleccionar a empresa asociada ó usuario xestor. Deste xeito, o usuario administrador pode engadir de forma sinxela a empresa bonificada, dado que se permite que no propio input se introduza texto que se envía ó servidor para que devolva aquelas entidades que conteñen dito texto no seu nome. En caso de que non se engada ningún texto, mostraranse todas as entidades dispoñibles. Para buscar as empresas do despregable, faise uso da petición GET engadida no controlador que devolve unha lista co ID e a etiqueta, sendo esta última a que se mostra no despregable. Durante este desenvolvemento tamén se fai que, tras o inicio de sesión do usuario, se mostren as funcionalidades das que dispón segundo o seu rol.

6.5 Sprint 2

6.5.1 Modelo de datos

Créase a táboa para almacenar a información relativa ós empregados, que conta cun ID, un código correspondente ó número de identificación da persoa, o tipo de documento que a identifica (NIF ou NIE), o nome e apelidos, o número de identificación da Seguridade Social, a data de nacemento, o correo electrónico, o número telefónico, o sexo, a categoría profesional, o grupo de cotización, e o nivel de estudos. Finalmente, almacénase o ID da empresa bonificada á que pertence o empregado. A entidade resultante é a que se mostra na figura 6.3.

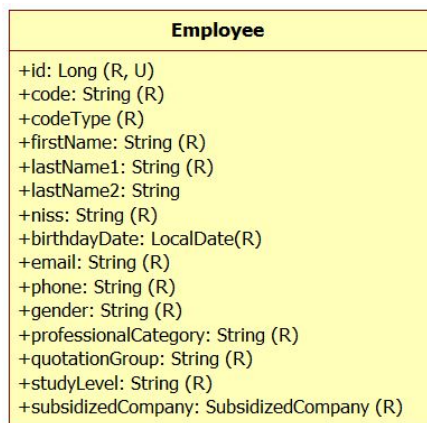


Figura 6.3: Entidade para Empregado

6.5.2 Backend

No desenvolvemento do backend deste Sprint créanse no paquete do dominio enumerados para os atributos tipo de documento, sexo, categoría profesional, grupo de cotización e nivel de

estudos. Ademais, engádense ós parámetros das funcións de listado, modificación e creación de empregados o código do usuario que está chamando as funcións, para poder coñecer deste xeito a empresa bonificada á que pertence e, así, no listado de empregados só se devolven os que pertencen a dita empresa. Cando se crea ou modifica un empregado tamén se engade a empresa do usuario ó empregado, para que quede gardada a que empresa está asociado. Isto implica unha nova función no DAO similar a de listar todas as entidades de forma paxinada, pero engadindo a condición WHERE para obter só aquelas nas que o ID da empresa coincida ca indicada. Na validación que se realiza ó crear ou modificar empregados, compróbase que non exista ningún empregado co mesmo código na empresa, permitindo que dúas ou máis empresas poidan ter rexistrado un empregado co mesmo código, pero que non existan dous usuarios co mesmo código nunha mesma empresa (isto ocorrerá de igual xeito en todas as entidades correspondentes ó módulo de xestión, é dicir, en centros formativos, formadores, accións formativas e grupos formativos). A continuación móstrase o validador implementado:

```

1 private void validate(EmployeeFormDTO employee, Long companyId,
2     boolean isNew) {
3     if (isNew) {
4         employeeDAO.findByCodeAndSubsidizedCompany(
5             employee.getCode(), companyId).ifPresent(u -> {
6                 throw new CodeAlreadyUsedException();
7             });
8     } else {
9         employeeDAO.findByCodeAndSubsidizedCompany(
10            employee.getCode(), companyId).ifPresent(u -> {
11                if (!u.getId().equals(employee.getId())) {
12                    throw new CodeAlreadyUsedException();
13                }
14            });
15    }
16 }
17 }

```

6.5.3 Frontend

No desenvolvemento do frontend deste Sprint faise que, cando o usuario elixe a opción do menú para amosar a lista de empregados, o primeiro que se consulta é o código de dito usuario, para poder realizar a petición correspondente ó servidor e só obter aqueles empregados que pertencen á empresa bonificada á que está asociado o usuario. O mesmo se realiza cas peticións de modificación e creación, garantindo, así, que un usuario non poida ler nin escribir datos doutras empresas que non sexan a súa.

6.6 Sprint 3

6.6.1 Modelo de datos

A táboa creada para almacenar a información dos centros formativos dispón dun ID, un código correspondente o número de identificación do centro xunto co tipo de documento (podendo ser CIF, NIF, NIE ou Pasaporte), o nome do centro e a dirección física xunto co código postal e a cidade. Finalmente, almacénase o ID da empresa bonificada á que se asocia o centro formativo. A entidade resultante é a que se mostra na figura 6.4.

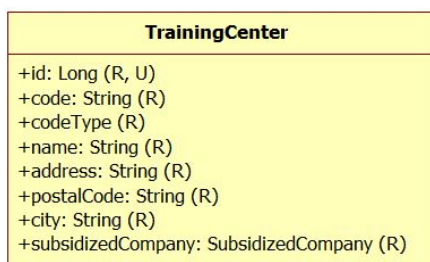


Figura 6.4: Entidade para Centro de formación

6.6.2 Backend

No desenvolvemento do backend deste Sprint engádesse ó paquete do dominio un enumerado que contén as os tipos de documento que identifican ó centro formativo. De igual xeito que no Sprint anterior (apartado 6.7.2), tamén se engade ós parámetros das funcións principais o login do usuario para poder identificar a que empresa corresponden os centros formativos que se están consultando ou escribindo.

6.6.3 Frontend

No desenvolvemento do frontend deste Sprint realízanse as mesmas accións ca no Sprint 2 (apartado 6.7.2) adaptadas á entidade centro formativo.

6.7 Sprint 4

6.7.1 Modelo de datos

A táboa que almacena a información dos formadores conta cun ID, un código que se corresponde co número de identificación do formador xunto co tipo documento ó que se corresponde dito número (NIF ou NIE), o nome e os apelidos da persoa, un número telefónico e

un correo electrónico de contacto e, tamén, o ID do centro formativo ó que pertence (empregado como chave foránea), xunto co ID da empresa bonificada correspondente. A entidade resultante é a que se mostra na figura 6.5.

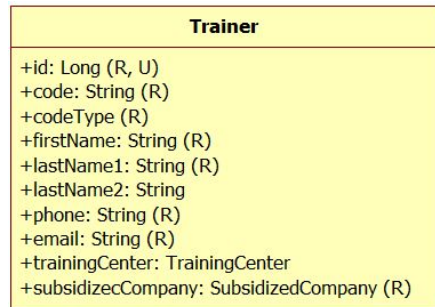


Figura 6.5: Entidade para Formador

6.7.2 Backend

No desenvolvemento do backend deste cuarto Sprint reutilízase o enumerado creado no Sprint 2 para o tipo de documento dos formadores. Tamén se engade unha función para obter unha lista de BasicDTO dos centros formativos rexistrados na empresa asociada ó usuario.

6.7.3 Frontend

No desenvolvemento do frontend deste Sprint engádese un selector no formulario de creación/modificación de formadores para poder seleccionar o centro formativo ó que pertence, facendo unha petición GET, na que se envía o login do usuario, para invocar á función do backend comentada.

6.8 Sprint 5

6.8.1 Modelo de datos

Créase unha táboa que almacena as accións formativas informando o ID, o código e o nome da acción formativa, o código do grupo do tipo de formación, o tipo de modalidade (Presencial, Teleformación ou Mixta), o número de horas de formación, o nivel (Básico ou Superior), os obxectivos, os contidos e as observacións. Mencionar que as horas de formación serán as correspondentes a horas presenciais, dado que será o tipo de modalidade que se trate durante este proxecto. Tamén se garda o ID da empresa bonificada correspondente. A entidade resultante é a que se mostra na figura 6.6.

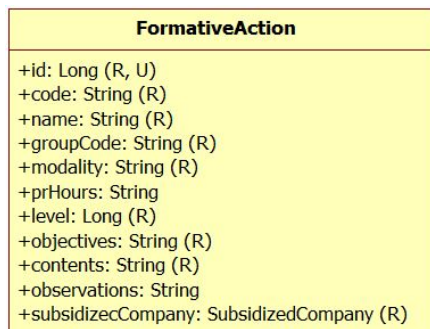


Figura 6.6: Entidade para Acción formativa

6.8.2 Backend

No desenvolvemento do backend deste Sprint engádense ó paquete do dominio os enumerados para os tipos de modalidade e os niveis das formacións.

6.8.3 Frontend

No desenvolvemento do frontend deste Sprint realízase o mesmo que nas últimas iteracións, adaptado á entidade da acción formativa.

6.9 Sprint 6

6.9.1 Modelo de datos

Créase unha táboa para almacenar os grupos formativos, a cal conta con: ID; código; descrición; datas de comezo e de fin; nome, apelidos e número de teléfono da persoa responsable do grupo; CIF, nome, dirección, código postal, cidade e país do lugar no que terá cabida a formación; horas totais, días da semana e horarios nos que se imparte a formación; medio, conexión e persoa de contacto xunto co número de teléfono relativos á aula virtual empregada; número de participantes; e custo directo, indirecto e salarial, xunto cas bonificacións mensuais e os importes do grupo formativo. Ademais da táboa principal, tamén se creou unha táboa intermedia (N a M) para relacionar os empregados cos grupos formativos, gardando o ID do empregado e o ID do grupo como chaves foráneas; e outra táboa intermedia (N a M) para asociar formadores cos grupos, almacenando o ID do formador, o ID do grupo e o número de horas que imparte. Así mesmo, engádese á táboa principal o ID do centro formativo que imparte a formación presencial e o ID da empresa bonificada ó que se corresponde o grupo. Finalmente, créase unha nova táboa para relacionar as bonificacións mensuais co grupo formativo, gardando o ID do grupo, o número de mes ó que se corresponde e o importe. A entidade resultante é a que se mostra na figura 6.7.

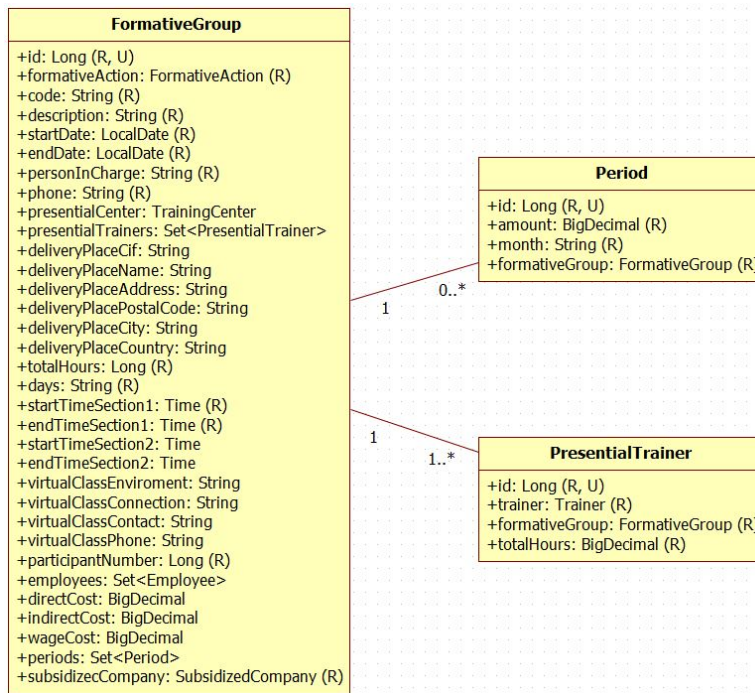


Figura 6.7: Entidade para Grupo formativo

6.9.2 Backend

No desenvolvemento do backend deste Sprint, como se mostra na figura 6.7, créase unha entidade para representar a táboa intermedia entre os grupos formativos e os formadores e poder almacenar as horas que imparten. Ademais, engádense novas funcións para obter listados de empregados, accións formativas e formadores. En todas elas se emprega o login do usuario para obter só aquelas entidades que pertencen á empresa bonificada do usuario, e tamén, no caso dos formadores, emprégase o ID do centro formativo para conseguir os formadores asociados a este.

6.9.3 Frontend

No desenvolvemento do frontend deste Sprint, a diferenza dos anteriores, a pantalla de detalle da entidade conta con 3 pestanas para distribuír a información. Para conseguir isto, créase un compoñente xeral, no cal se engaden os tres subcompoñentes relativos a cada unha das pestanas, empregando o módulo de Angular NgbNavModule, que permite implementar este comportamento das xanelas. Ditas pestanas están agrupadas da seguinte forma: a primeira conta con toda a información xeral do grupo formativo; a segunda cos empregados que participan e o número deles; e a última recolle a información relativa ós custos do grupo, así como, as bonificacións mensuais. Ademais, créase un compoñente similar ó selector creado

anteriormente, pero que permite engadir unha lista de elementos, en lugar dun só. Este novo compoñente é empregado para engadir os participantes do grupo formativo e emprégase da seguinte forma:

```

1 <pfobe-input-list label="Participants"
2   [editing]="contextService.state.editing"
3   [list]="entity.employees" [options]="employeesSelector"
4   (onChangeList)="saveParticipatingEmployees($event)">
5 </pfobe-input-list>

```

Para obter o total de horas de formación de impartida no grupo formativo realízase un cálculo no frontend tendo en conta as datas de inicio e fin, os días da semana que se imparte a formación e o horario de impartición. Ademais, para facilitarlle a asignación das bonificacións mensuais ó usuario, tamén se calculan os importes máximos que se poden bonificar en función dos custos, da modalidade e do nivel da formación.

6.10 Sprint 7

Neste sétimo Sprint desenvólvese o planificado no apartado 5.8 referente á importación de empregados dende un ficheiro de Excel. Realizar esta funcionalidade supón indagar acerca de como poder extraer información deste tipo de arquivos para poder engadila en entidades Java e logo gardala na base de datos.

6.10.1 Modelo de datos

Para este Sprint non se realiza ningún cambio no modelo de datos da aplicación, posto que se traballa sobre a táboa xa creada de empregados. Por outro lado, créase un ficheiro Excel o cal serve de referencia para establecer a estrutura que deberán ter todos aqueles arquivos deste tipo que sexan empregados para importar os empregados da empresa bonificada. Dito ficheiro establece que a información de cada empregado ten que ser recollida nunha fila da táboa Excel e tamén indica a posición na columna que debe ocupar cada dato mediante unha primeira fila que representa a cabeceira da táboa, como se mostra na figura 6.8.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Código	TipoCódigo	Nombre	Apellido1	Apellido2	NISS	FechaNacimiento	Email	Teléfono	Género	CatProfesional	GrupoCotiz	NivelEstudios

Figura 6.8: Cabeceira do ficheiro Excel para engadir empregados

6.10.2 Backend

No desenvolvemento do backend deste Sprint, tras buscar información sobre como poder extraer información dun ficheiro Excel para poder crear entidades en Java, decídese empregar a librería Apache POI [27], que conta con funcionalidades para ler e escribir arquivos en

formados de Microsoft Office, como é o caso de Excel. Unha vez elixida a forma de tratar o ficheiro Excel, impleméntase unha función no servizo dos empregados á cal lle chega un login de usuario e un ficheiro. Co login do usuario obtense a empresa bonificada á que pertence. O seguinte que se realiza é un obxecto tipo Map para representar unha estrutura de datos para almacenar pares "chave/valor". En dito obxecto engádense pares indicando como chave o nome da columna do ficheiro Excel e como valor o nome do atributo da entidade empregado co que se corresponde, como se mostra no seguinte código:

```
1 Map<String, String> columns = new HashMap<>();
2 columns.put("Código", "code");
3 columns.put("TipoCódigo", "codeType");
4 columns.put("Nombre", "firstName");
5 columns.put("Apellido1", "lastName1");
6 columns.put("Apellido2", "lastName2");
7 columns.put("NISS", "niss");
8 columns.put("FechaNacimiento", "birthdayDate");
9 columns.put("Email", "email");
10 columns.put("Teléfono", "phone");
11 columns.put("Género", "gender");
12 columns.put("CatProfesional", "professionalCategory");
13 columns.put("GrupoCotiz", "quotationGroup");
14 columns.put("NivelEstudios", "studyLevel");
```

A continuación, créase un obxecto tipo Workbook da librería mencionada a partir do ficheiro, do cal se obtén a primeira folla do Excel e se almacena nunha variable tipo Sheet (da librería Apache POI). Logo, indícase que a primeira fila da folla é a cabeceira do documento e obtense o número de filas e columnas que presenta o ficheiro. Finalmente, con esta información créanse dous bucles FOR anidados para recorrer as celdas de Excel como se fose unha matriz e poder ir extraendo a información de cada unha das celdas e almacenándoa nunha lista de Maps na que cada Map representa unha fila (un empregado) sendo a chave o nome da columna (o nome do atributo do empregado) e o valor a información que está na celda (o valor do atributo do empregado). Unha vez obtida esta lista de Maps, créase un String mediante ObjectMapper (pertencente a librería Jackson), para logo, empregando a función readValue de ObjectMapper, obter a lista de empregados ca entidade correspondente. A cada empregado asínaselle a empresa bonificada obtida a partir do login do usuario e gárdase na base de datos facendo uso da función creada no Sprint 2 (apartado 6.7.2) para rexistrar empregados. Tamén se engade no controlador dos empregados o método para recoller a petición POST para importar os empregados desde o ficheiro Excel.

6.10.3 Frontend

No desenvolvemento do frontend deste Sprint non se crea ningún compoñente novo, pero engádese na pantalla de listar empregados a opción de seleccionar o ficheiro Excel e o botón para enviar a petición POST e importar os empregados informados no ficheiro seleccionado.

6.11 Sprint 8

Neste oitavo Sprint realízase a funcionalidade citada no apartado 5.9 relacionada ca xeración de ficheiros XML para dar de alta empregados. Para iniciar este desenvolvemento búscase información sobre como xerar un documento XML e decídese empregar a librería W3C [28].

6.11.1 Modelo de datos

No desenvolvemento deste Sprint non se realizan modificacións sobre o modelo de datos.

6.11.2 Backend

No desenvolvemento do backend deste Sprint, créase unha función que recibe unha lista de empregados e, mediante a funcionalidade que ofrece a librería mencionada, créase un documento XML ó cal se lle van engadindo etiquetas e o contido destas recorrendo a lista de empregados cun bucle FOR. O nome das etiquetas e o formato do seu contido obtense do esquema que ofrece a FUNDAE na súa páxina web, onde tamén se atopa un ficheiro de exemplo:

```
1 <participantes>
2   <participante>
3     <D_DOCUMENTO>00000002W</D_DOCUMENTO>
4     <N_TIPO_DOCUMENTO>10</N_TIPO_DOCUMENTO>
5     <D_APELLIDO1>Perez</D_APELLIDO1>
6     <D_APELLIDO2>Lopez</D_APELLIDO2>
7     <D_NOMBRE>Juan</D_NOMBRE>
8     <D_NISS>201111111224</D_NISS>
9     <B_SEXO>M</B_SEXO>
10    <F_NACIMIENTO>11/06/1982</F_NACIMIENTO>
11  </participante>
12 </participantes>
```

Finalmente, engádese no controlador dos empregados un método para recoller a petición POST ca lista de empregados para ser incluídos no ficheiro XML.

6.11.3 Frontend

No desenvolvemento do frontend deste Sprint non se crea ningún compoñente novo, pero modifícase a pantalla de listar empregados engadindo un checkbox en cada fila da lista para seleccionar, así, os empregados que se queiran incluír no ficheiro XML, e un botón para enviar a petición POST ca lista de empregados.

6.12 Sprint 9

Neste noveno Sprint, aproveitando os coñecementos obtidos no Sprint anterior (apartado 6.11), realízase a funcionalidade mencionada no apartado 5.10 para xerar o ficheiro XML para dar de alta as accións formativas.

6.12.1 Modelo de datos

No desenvolvemento deste Sprint non se realizan modificacións sobre o modelo de datos.

6.12.2 Backend

No desenvolvemento do backend deste Sprint, créase unha función que recibe unha lista de accións formativas e, seguindo o desenvolvemento do Sprint 8 (apartado 6.11), créase un documento XML recorrendo a lista de accións formativas cun bucle FOR. O nome das etiquetas e o formato de seu contido obtense do esquema que ofrece a FUNDAE na súa páxina web, onde tamén ofrecen un ficheiro de exemplo:

```

1 <AccionesFormativas>
2   <AccionFormativa>
3     <codAccion>0001</codAccion>
4     <nombreAccion>Accion Formativa 0001</nombreAccion>
5     <codGrupoAccion>035-05</codGrupoAccion>
6     <codAreaProfesional>ADGN</codAreaProfesional>
7     <modalidadPresencial>
8       <horasPr>10</horasPr>
9     </modalidadPresencial>
10    <nivelFormacion>0</nivelFormacion>
11    <objetivos>Objetivos de la Acción Formativa</objetivos>
12    <contenidos>Contenidos de la Acción Formativa</contenidos>
13  </AccionFormativa>
14 </AccionesFormativas>

```

Finalmente, engádese no controlador das accións formativas un método para recoller a petición POST ca lista de accións para ser incluídas no documento. A continuación móstrase un fragmento de código sobre como se crean as etiquetas no ficheiro XML:

```
1 // Elementos acción formativa
2 for (FormativeAction fa : formativeActions) {
3     Element accFor = document.createElement("AccionFormativa");
4
5     // Elemento para código
6     Element cod = document.createElement("codAccion");
7     Text textCod = document.createTextNode(fa.getCode());
8     cod.appendChild(textCod);
9     accFor.appendChild(cod);
10
11    // Elemento para Nombre de Acción
12    Element nombre = document.createElement("nombreAccion");
13    Text textNombre = document.createTextNode(fa.getName());
14    nombre.appendChild(textNombre);
15    accFor.appendChild(nombre);
16
17    [...]
18 }
```

6.12.3 Frontend

No desenvolvemento do frontend deste Sprint non se crea ningún compoñente novo, mais modificouse a pantalla de listar accións engadindo un checkbox en cada fila da lista para seleccionar, así, os accións que se queiran engadir no ficheiro XML e un botón para enviar a petición POST ca lista de accións.

6.13 Sprint 10

Neste décimo e último Sprint, seguindo o desenvolvemento realizado nos dous Sprints anteriores (apartados 6.11 e 6.12), impleméntase a funcionalidade recollida no apartado 5.11 sobre a xeración dos ficheiros XML para informar o inicio e fin dos grupos formativos.

6.13.1 Modelo de datos

No desenvolvemento deste Sprint non se realizan modificacións sobre o modelo de datos.

6.13.2 Backend

No desenvolvemento do backend deste Sprint créanse dúas funcións que reciben unha lista de grupos formativos, unha para xerar o ficheiro de inicio e a outra para xerar o de fin. A forma de crear ambos ficheiros segue a estrutura do feito anteriormente nos Sprints 8 e 9

(apartados 6.11 e 6.12), ca diferencia de que o ficheiro XML de fin de grupos formativos é máis complexo, posto que conta con moita máis información. A continuación móstrase o exemplo ofrecido pola FUNDAE para o ficheiro de inicio grupos formativos:

```

1 <!-- Inicio grupos formativos -->
2 <grupos>
3   <grupo>
4     <idAccion>1</idAccion>
5     <idGrupo>1</idGrupo>
6     <descripcion>Ejemplo grupo formativo 2021</descripcion>
7     <NumeroParticipante>8</NumeroParticipante>
8     <fechaInicio>01/01/2021</fechaInicio>
9     <fechaFin>02/01/2021</fechaFin>
10    <responsable>Responsable</responsable>
11    <telefonoContacto>911111111</telefonoContacto>
12    <jornadaPresencial>
13      [...]
14    </jornadaPresencial>
15    <aulaVirtual>
16      [...]
17    </aulaVirtual>
18  </grupo>
19 </grupos>

```

A continuación móstranse o exemplo ofrecido pola FUNDAE para o ficheiro de fin grupos formativos:

```

1 <!-- Fin grupos formativos -->
2 <grupos>
3   <grupo>
4     <idAccion>00002</idAccion>
5     <idGrupo>00001</idGrupo>
6     <participantes>
7       <participante>
8         [...]
9       </participante>
10      <participante>
11        [...]
12      </participante>
13    </participantes>
14    <costes>
15      <coste>
16        <directos>1500</directos>
17        <indirectos>2000</indirectos>
18        <salariales>1500</salariales>
19        <periodos>
20          <periodo>

```

```
21         <mes>10</mes>
22         <importe>900</importe>
23     </periodo>
24     [...]
25 </periodos>
26 </coste>
27 </costes>
28 </grupo>
29 </grupos>
```

Tamén se engaden no controlador dos grupos formativos os métodos para recoller as peticións POST cas listas de grupos formativos para ser incluídos nos documentos.

6.13.3 Frontend

No desenvolvemento do frontend deste Sprint non se crea ningún compoñente, pero modifícase o compoñente de listado de grupos formativos para engadir un checkbox en cada fila da lista para seleccionar os grupos a engadir no ficheiro; e engádense dous botóns, un para enviar a petición para crear o ficheiro de inicio de grupos formativos e o outro para o de fin.

6.14 Sprint 11

Neste undécimo Sprint realízase a funcionalidade citada no apartado 5.12 relacionada ca xeración de documentos PDF para que as empresas bonificadas poidan realizar un control da asistencia dos seus empregados ós grupos formativos. Para iniciar este desenvolvemento búscase información sobre como xerar un documento PDF dende o servidor e, tras probar e barallar varias opcións, decídese empregar a librería iTextPDF [29]

6.14.1 Modelo de datos

No desenvolvemento deste Sprint non se realizan modificacións sobre o modelo de datos.

6.14.2 Backend

No desenvolvemento do backend deste Sprint é necesario crear unha función que recibe o ID dun grupo formativo, empregado para obter a entidade do grupo concreto xunto cos datos necesarios para xerar o documento. No ficheiro inclúese información relativa á empresa bonificada (razón social e CIF), a acción formativa (nome e código) e ó grupo formativo. Deste último infórmase o código, as datas de inicio e fin, os días da semana que se imparte e a persoa a cargo do grupo xunto co teléfono de contacto. Tamén se engade unha táboa ó documento ca información dos participantes, deixando un oco para que poidan asinar e outro para engadir

observacións. Finalmente, engádesse no controlador a función que recibe a petición co ID do grupo formativo para xerar o documento.

6.14.3 Frontend

No desenvolvemento do frontend deste Sprint non se crea ningún compoñente novo, pero engádesse no compoñente do formulario dos grupos formativos un botón para lanzar a petición de creación do documento PDF para os partes de asistencia.

7.1 Inicialización de datos

Para poder realizar probas sobre a aplicación desenvolvida, é necesario inicializar algúns datos de exemplo para poder rexistrarse na aplicación. O primeiro que se crea tras a realización do Sprint 1 (apartado 6.4) é un usuario con rol administrador, o cal pode ser empregado para crear calquer outro tipo de usuario e empresa bonificada. Ademais de dito usuario, créanse dúas empresas bonificadas e dous usuarios xestores asociados a estas empresas. Desta forma próbase que as táboas están ben deseñadas e as claves foráneas funcionan correctamente, complementando, así, as probas unitarias. Logo, durante o desenvolvemento de cada Sprint no que se engade unha nova táboa á base de datos, tamén se engaden algunhas entidades de exemplo para probar que as novas táboas recollen a información de forma axeitada. Para comprobar que os datos se rexistran correctamente na base de datos, emprégase a ferramenta PgAdmin, mencionada no apartado 4.4.2.

7.2 Probas de sistema

Durante o desenvolvemento de cada Sprint, realízanse probas de forma continua sobre as funcionalidades desenvolvidas para comprobar e corrixir o funcionamento do traballo realizado. Mencionar que Node.js (citado no apartado 4.3.4) permite realizar cambios no código mentres se está executando, dado que no momento de gardar, volve a recompilar o frontend. Isto facilita a programación en TypeScript (apartado 4.1.2), xa que se poden ir realizando cambios e probándoos de maneira practicamente simultánea, o que fai diminuír o número de erros do código, debido a que non se deixan as probas para o final da implementación. Tamén se fai uso do inspector que ofrece Google Chrome (mencionado no apartado 4.2.2) para comprobar os erros que poidan saír pola consola e revisar que as peticións ó servidor se lanzan correctamente.

7.3 Validación de ficheiros xerados

Para comprobar que os ficheiros XML xerados cumpren co formato que establece a FUN-DAE, empregouse un validador online (Free Online XML Validator [30]) no cal se introduce o esquema que debe seguir o documento xunto co contido do propio documento, mostrando os erros que presenta no caso de que os teña ou amosando unha mensaxe satisfactoria. Isto serviu para percatarse de erros que a simple vista, ollando os ficheiros, non se atoparían.

7.4 Probas de aceptación

Tras o remate de cada Sprint, amósase o resultado logrado ó Product Owner e permítese que probe as funcionalidades desenvolvidas para recibir a súa aprobación e os comentarios sobre aqueles aspectos que se deben modificar ou corrixir. Non todas as observacións feitas polo cliente puideron ser realizadas, debido a que se saían do alcance do TFG, pero quedan rexistradas para ser realizadas como liñas futuras de desenvolvemento.

Conclusións e Liñas Futuras

8.1 Conclusións

Tras finalizar o desenvolvemento e as probas do produto, obtense unha aplicación que conta cas funcionalidades obtidas durante a análise de requisitos e que cumpre cos obxectivos propostos. O alumno puido mellorar a súa experiencia e as súas competencias naqueles procesos involucrados no ciclo de vida deste proxecto. Na análise de requisitos foise perfeccionando a comunicación co cliente tras cada reunión con el, conseguindo obter os aspectos principais que debería cumprir o proxecto. Na planificación foi de utilidade aplicar coñecementos sobre o ciclo de vida do software, para poder estimar un traballo equilibrado en cada Sprint e poder conseguir un proxecto abarcable. Na implementación do software, aplicáronse moitas leccións obtidas durante os anos da titulación e puidéronse reforzar moitas delas. Ademais, tamén se obtivo experiencia programando con Angular (mencionado no apartado 4.3.5), que está sendo unha das plataformas de desenvolvemento web máis demandadas actualmente.

8.2 Liñas futuras

Para continuar co traballo desenvolvido neste proxecto, poderíase engadir a funcionalidade correspondente para rexistrar accións formativas con modalidade de teleformación e mixta. Isto supón ter en conta, tamén, o número de horas impartidas de forma online, así como as titorías impartidas polos formadores destes tipos de modalidade. Outro aspecto a engadir é a internacionalización da aplicación, xa que agora se atopa en inglés, e sería de interese que tamén estivese en castelán e en galego. Tamén se poderían almacenar na base de datos do proxecto todos os grupos do tipo de formación (con código e nome) posibles para ser asignados a unha acción formativa, para evitar que o usuario teña que saír fóra da aplicación para consultar o código ó que pertence a acción que está rexistrando. É unha tarefa sinxela pero custosa, xa que existen moitos tipos de grupos (en total son 910). Outra funcionalidade

a engadir sería que, en lugar de xerar os ficheiros XML para dar de alta participantes, accións formativas e o inicio e o fin dos grupos formativos, ditos ficheiros se enviaran directamente dende a aplicación cara a FUNDAE, aforrándolle esa tarefa intermedia ó usuario de ter que comunicarllos unha vez os obtén. Finalmente, crese que podería ser de utilidade engadir ás entidades de empregados, accións formativas e grupos formativos un atributo de estado, que informe se xa se xeraron ficheiros XML con elas e, no caso dos grupos formativos, que documentos se crearon. Isto sería para informar ó usuario, non para impedir que poida engadir ningunha entidade a un ficheiro.

Apéndices

Manual de usuario

MANUAL no que se detalla o comportamento que deberá seguir o usuario na aplicación para facer uso das funcionalidades ofrecidas.

A.1 Inicio de sesión

Na figura A.1 móstrase a pantalla de inicio de sesión, onde o usuario que empregue a aplicación pode autenticarse introducindo o seu login e contrasinal nos espazos indicados e logo premer o botón "Login".

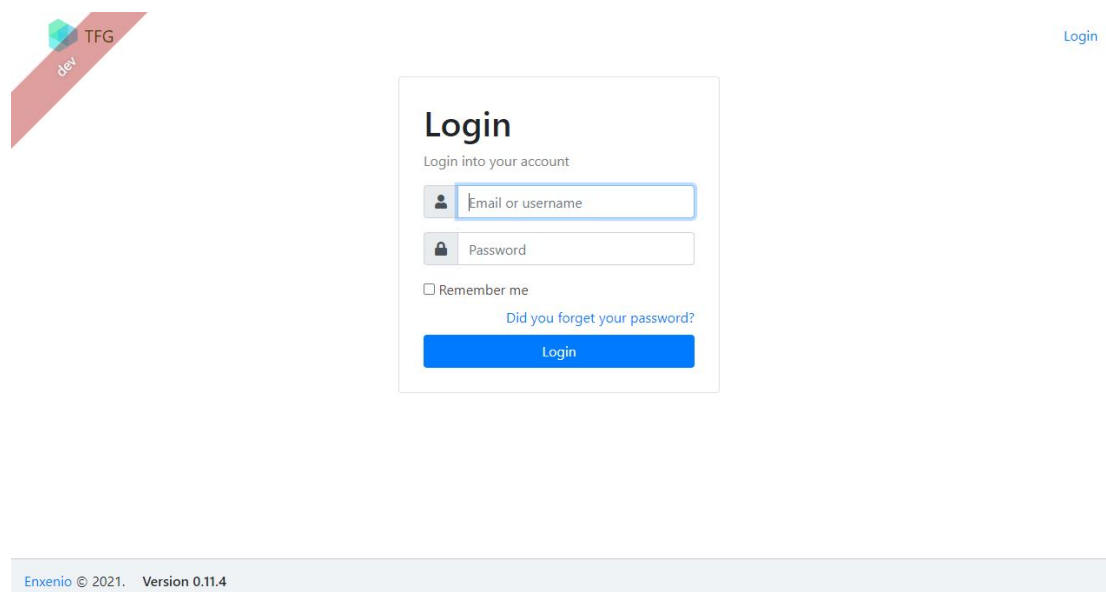


Figura A.1: Pantalla de inicio de sesión

A.2 Menú administrador

Tras realizar o inicio de sesión, se o usuario autenticado ten o rol Administrador accederá a pantalla representada na figura A.2, onde terá a posibilidade de acceder á funcionalidade de listar usuarios e empresas bonificadas premendo nas correspondentes opcións do menú que se atopa no lateral esquerdo ("Users" e "Companies").

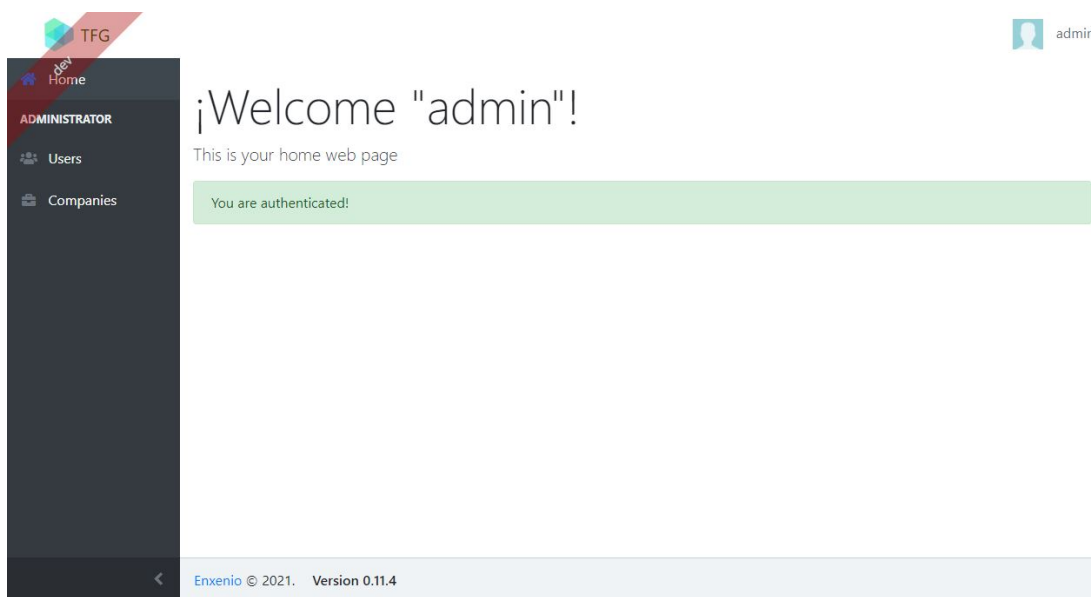


Figura A.2: Pantalla de menú do usuario Administrador

A.3 Listado de usuarios

Tras seleccionar a opción "Users" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.3, onde pode visualizar o listado de usuarios, crear un novo premendo o botón "New user" e visualizar o detalle de calquera deles pulsando sobre cadanseu código.

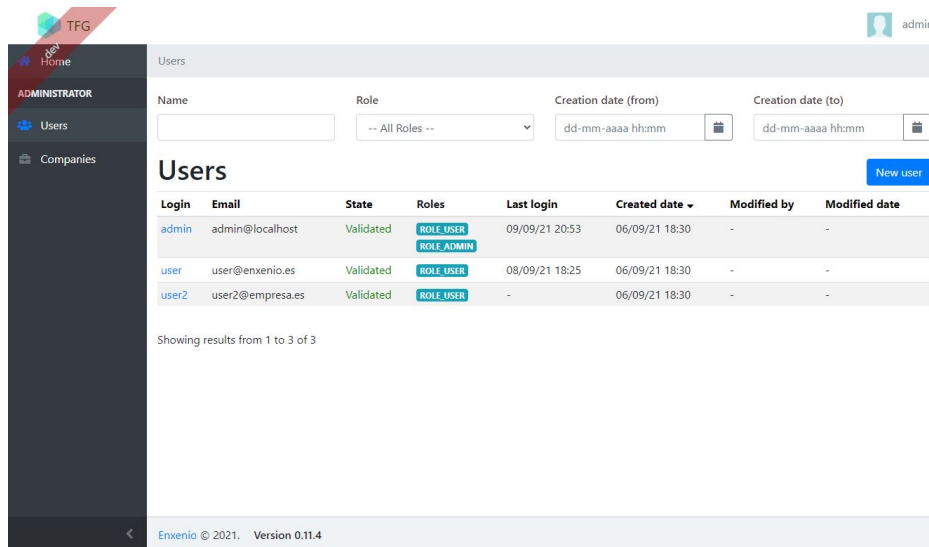


Figura A.3: Pantalla de listado de Usuarios

A.4 Formulario de usuario

Tras seleccionar un usuario do listado de usuarios, o usuario accederá á pantalla mostrada na figura A.4, onde terá a opción de editar información do usuario premendo no botón "Edit" ou eliminalo pulsando o botón "Delete". No caso de que o usuario pulse o botón "New user" na pantalla representada na figura A.3, accederá a unha pantalla similar á mostrada na figura A.4 pero en modo edición e sen ningún campo informado para que poida indicar a información do novo usuario a engadir.

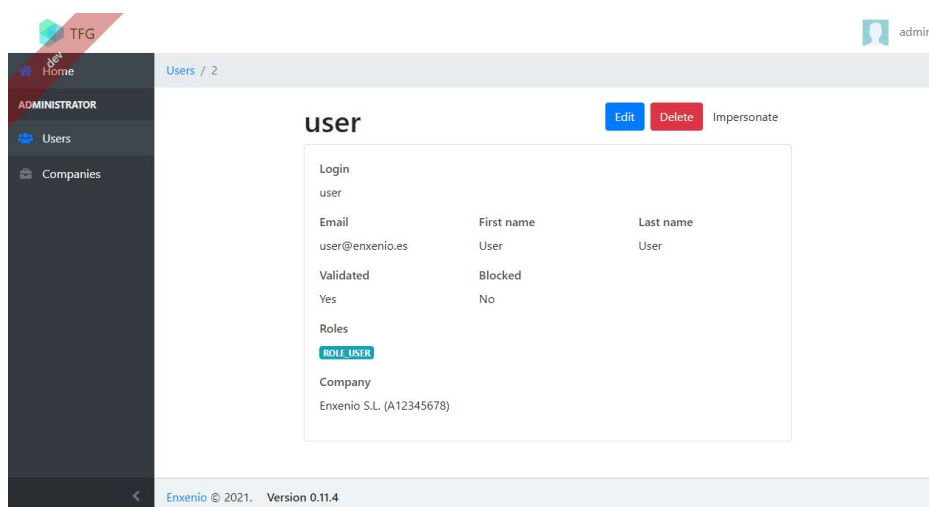


Figura A.4: Pantalla de formulario de Usuario

A.5 Listado de empresas bonificadas

Tras seleccionar la opción "Companies" del menú lateral izquierdo, el usuario accederá a la pantalla representada en la figura A.5, donde puede visualizar el listado de empresas bonificadas, crear una nueva presionando el botón "+Add" e visualizar el detalle de cualquiera de las empresas pulsando sobre su código.

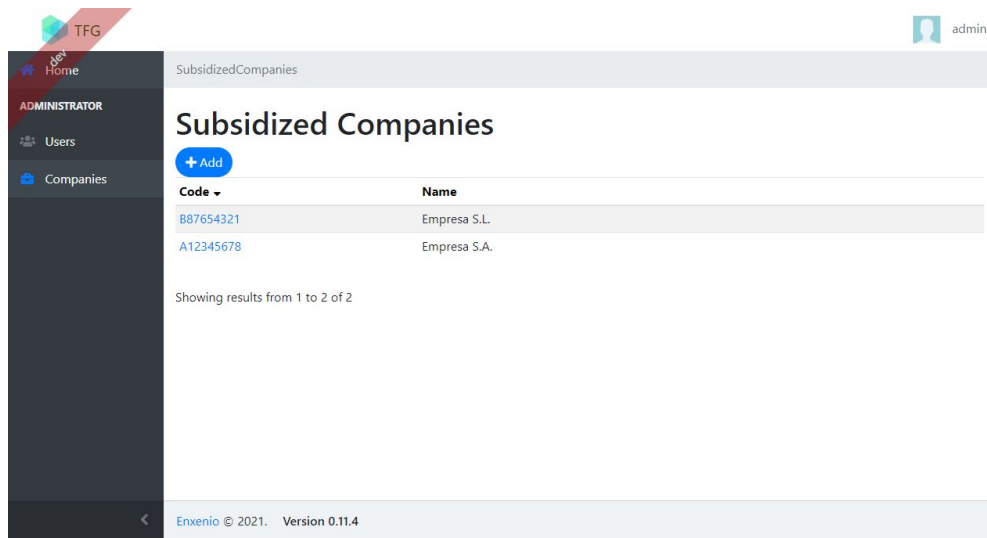


Figura A.5: Pantalla de listado de Empresas bonificadas

A.6 Formulario de empresa bonificada

Tras seleccionar una empresa bonificada del listado de empresas, el usuario accederá a la pantalla mostrada en la figura A.6, donde tendrá la opción de editar información de la empresa presionando el botón "Edit". No caso de que el usuario pulse el botón "+Add" en la pantalla representada en la figura A.5, accederá a una pantalla similar a la mostrada en la figura A.6 pero en modo edición e sin ningún campo informado para que pueda indicar la información de la nueva empresa a engadir.

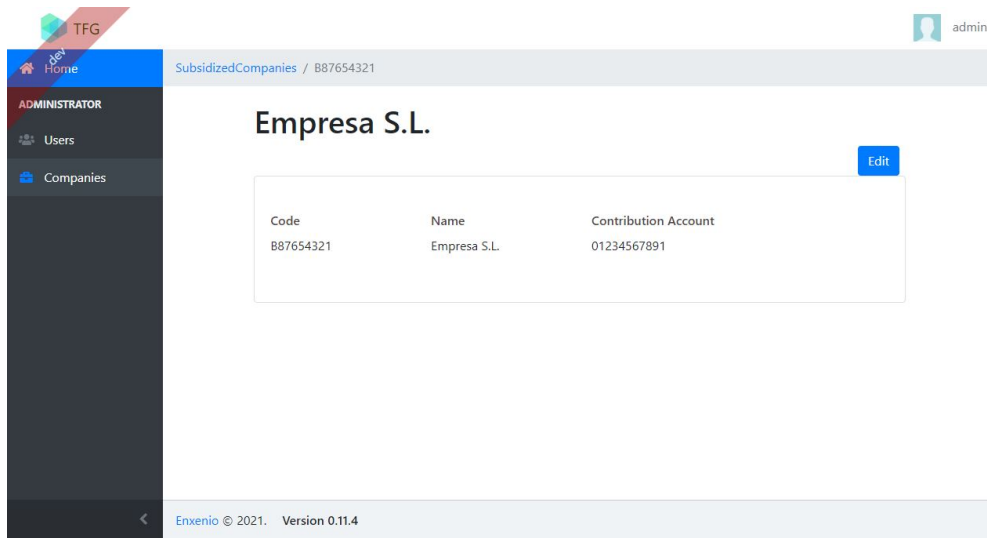


Figura A.6: Pantalla de formulario de Empresa bonificada

A.7 Menú xestor

Tras realizar o inicio de sesión, se o usuario autenticado ten o rol Xestor accederá a pantalla representada na figura A.7, onde terá a posibilidade de acceder á funcionalidade de listar empregados, centros de formación, formadores, accións formativas e grupos formativos (asociados todos eles á empresa que xestiona o usuario) premendo nas correspondentes opcións do menú que se atopa no lateral esquerdo ("Employees", "Trainig Centers", "Trainers", "Formative Actions" e "Formative Groups").

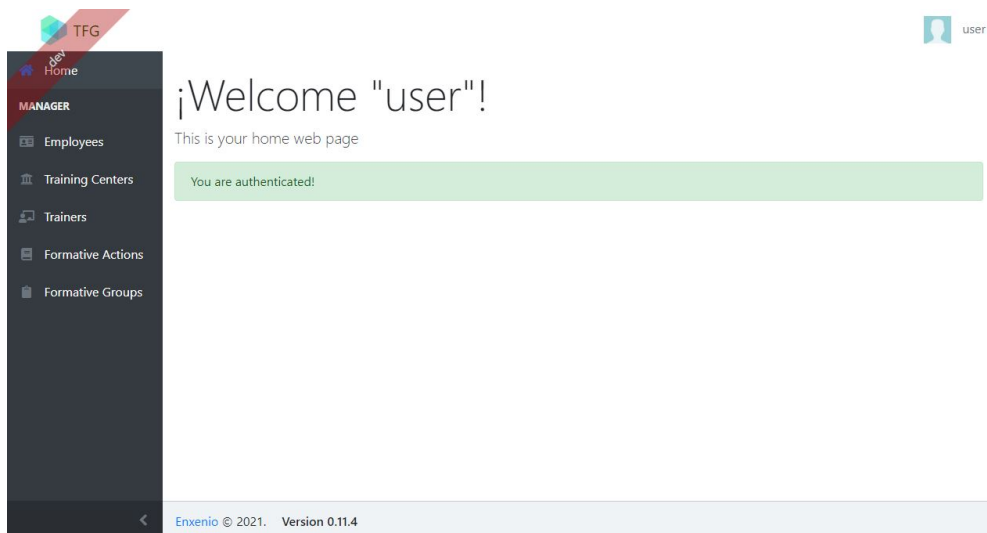


Figura A.7: Pantalla de menú do usuario Xestor

A.8 Listado de empregados

Tras seleccionar a opción "Employees" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.8, onde pode visualizar o listado de empregados; crear un novo premendo o botón "+Add"; visualizar o detalle de calquera dos empregados pulsando sobre cadanseu código; importar empregados dende un ficheiro Excel, tras escoller o documento premendo no botón "Choose File" e logo pulsar o botón "Import"; e xerar o ficheiro XML tras seleccionar nos checkboxes da columna "To XML" os empregados que se queiran incluír en dito documento e logo premer o botón "Generate XML".

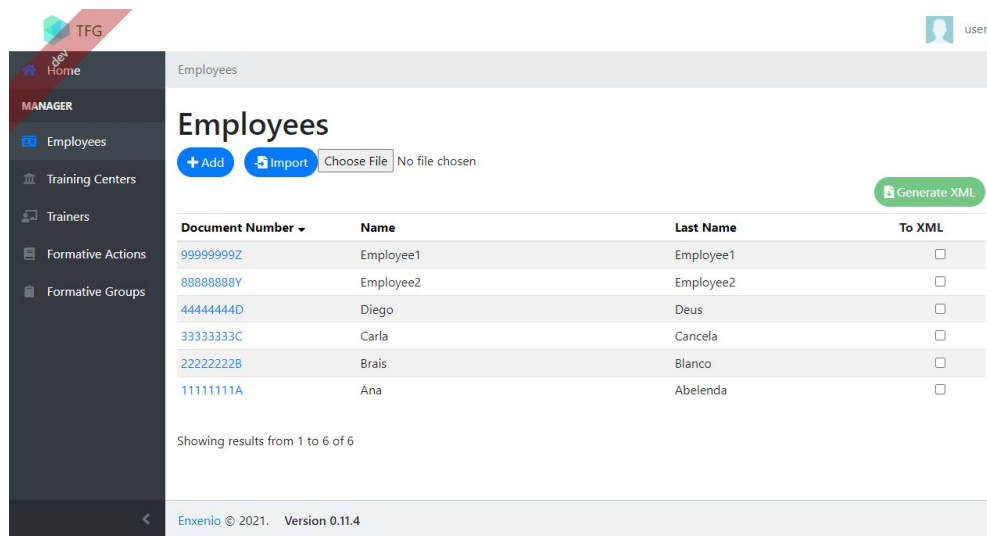


Figura A.8: Pantalla de listado de Empleados

A.9 Formulario de empregados

Tras seleccionar un empregado do listado, o usuario accederá á pantalla mostrada na figura A.9, onde terá a opción de editar información do empregado premendo no botón "Edit". No caso de que o usuario pulse o botón "+Add" na pantalla representada na figura A.8, accederá a unha pantalla similar á mostrada na figura A.9 pero en modo edición e sen ningún campo informado para que poida indicar a información do novo empregado a engadir.

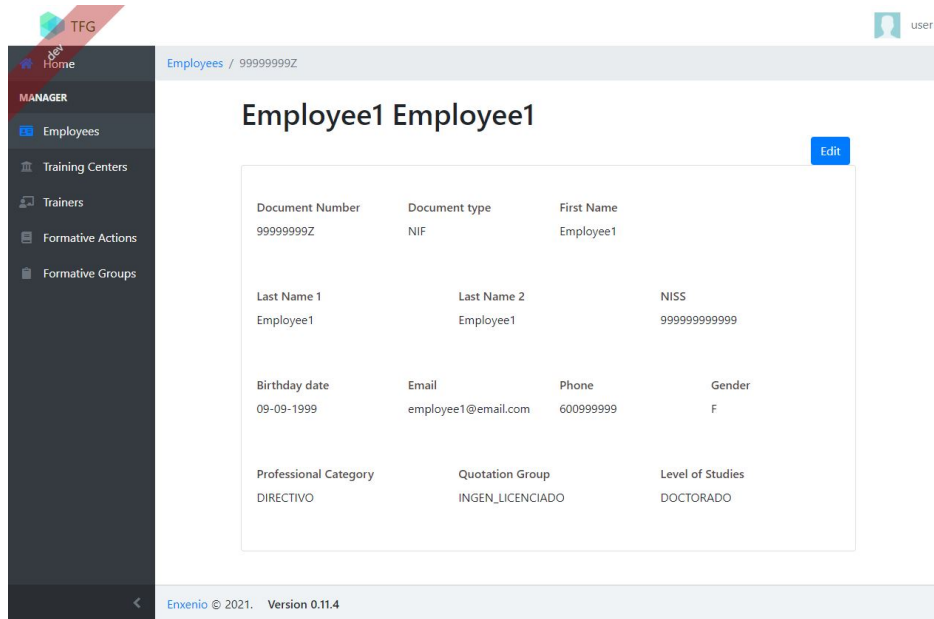


Figura A.9: Pantalla de formulario de Empleado

A.10 Listado de centros de formación

Tras seleccionar a opción "Training Centers" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.10, onde pode visualizar o listado de centros formativos, crear un novo premendo o botón "+Add" e visualizar o detalle de calquera dos centros pulsando sobre cadanseu código.

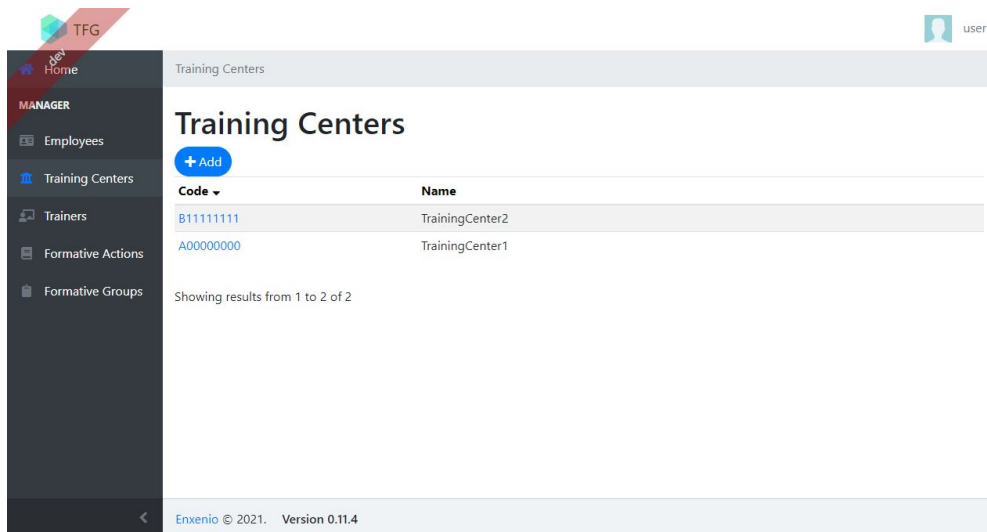


Figura A.10: Pantalla de listado de Centros de formación

A.11 Formulario de centros de formación

Tras seleccionar un centro de formación do listado, o usuario accederá á pantalla mostrada na figura A.11, onde terá a opción de editar información do centro premendo no botón "Edit". No caso de que o usuario pulse o botón "+Add" na pantalla representada na figura A.10, accederá a unha pantalla similar á mostrada na figura A.11 pero en modo edición e sen ningún campo informado para que poida indicar a información do novo centro a engadir.

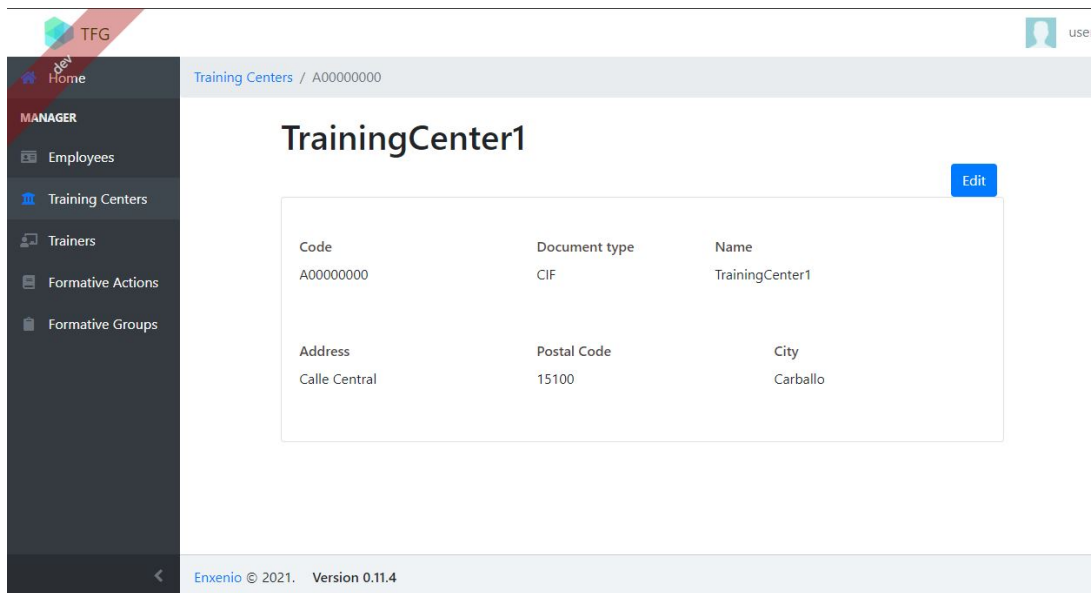


Figura A.11: Pantalla de formulario de Centro de formación

A.12 Listado de formadores

Tras seleccionar a opción "Trainers" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.12, onde pode visualizar o listado de formadores, crear un novo premendo o botón "+Add" e visualizar o detalle de calquera deles pulsando sobre cadanseu código.

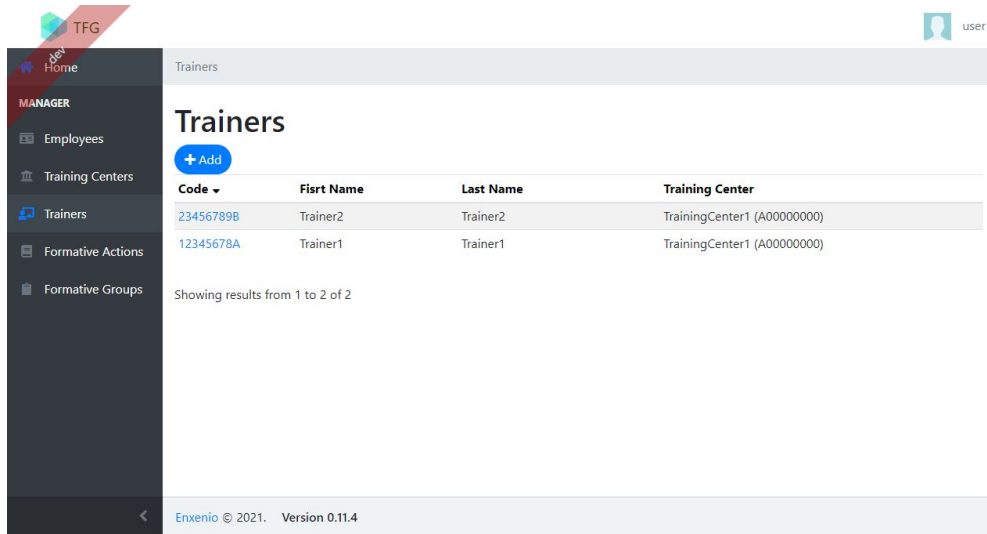


Figura A.12: Pantalla de listado de Formadores

A.13 Formulario de formador

Tras seleccionar un formador do listado, o usuario accederá á pantalla mostrada na figura A.13, onde terá a opción de editar información do formador premendo no botón "Edit". No caso de que o usuario pulse o botón "+Add" na pantalla representada na figura A.12, accederá a unha pantalla similar á mostrada na figura A.13 pero en modo edición e sen ningún campo informado para que poida indicar a información do novo formador a engadir.

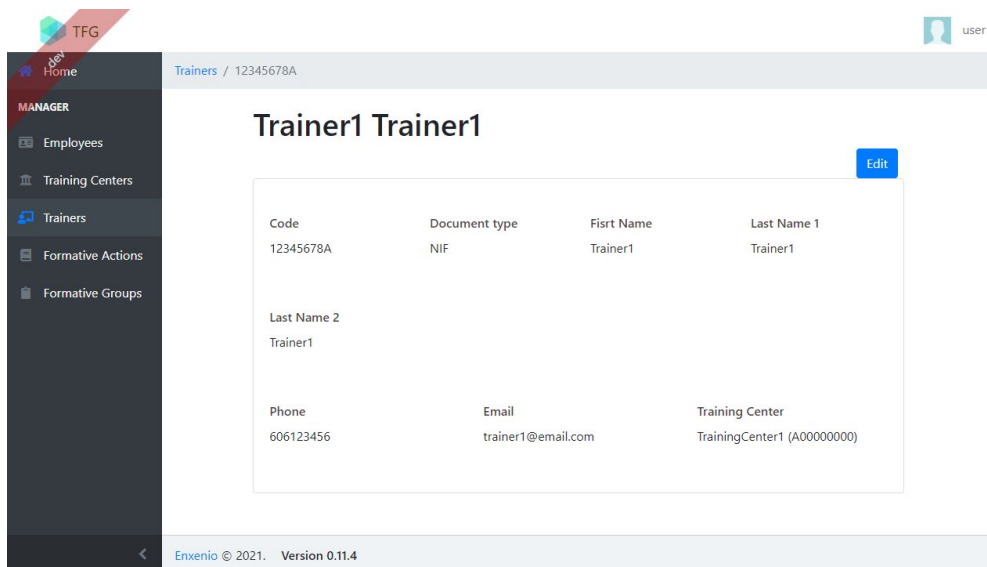


Figura A.13: Pantalla de formulario de Formadores

A.14 Listado de accións formativas

Tras seleccionar a opción "Formative Actions" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.14, onde pode visualizar o listado de accións; crear unha nova premendo o botón "+Add"; visualizar o detalle de calquera delas pulsando sobre cadanseu código; e xerar o ficheiro XML tras seleccionar nos checkboxes da columna "To XML" aquelas que se queiran incluír en dito documento e logo premer o botón "Generate XML".

The screenshot displays the 'Formative Actions' management page. On the left is a dark sidebar menu with 'MANAGER' at the top and several menu items: 'Home', 'Employees', 'Training Centers', 'Trainers', 'Formative Actions' (highlighted), and 'Formative Groups'. The main content area has a header 'FormativeActions' and a title 'Formative Actions' with a '+ Add' button. Below the title is a table with the following data:

Code	Name	Modality	Level	To XML
00002	FormativeAction2	PRESENIAL	BASICO	<input type="checkbox"/>
00001	FormativeAction1	PRESENIAL	SUPERIOR	<input type="checkbox"/>

Below the table, it says 'Showing results from 1 to 2 of 2'. In the top right corner of the table area, there is a green 'Generate XML' button. At the bottom of the page, the footer reads 'Enxenio © 2021. Version 0.11.4'.

Figura A.14: Pantalla de listado de Accións formativas

A.15 Formulario de acción formativa

Tras seleccionar unha acción formativa do listado, o usuario accederá á pantalla mostrada na figura A.15, onde terá a opción de editar información da acción premendo no botón "Edit". No caso de que o usuario pulse o botón "+Add" na pantalla representada na figura A.14, accederá a unha pantalla similar á mostrada na figura A.15 pero en modo edición e sen ningún campo informado para que poida indicar a información da nova acción a engadir.

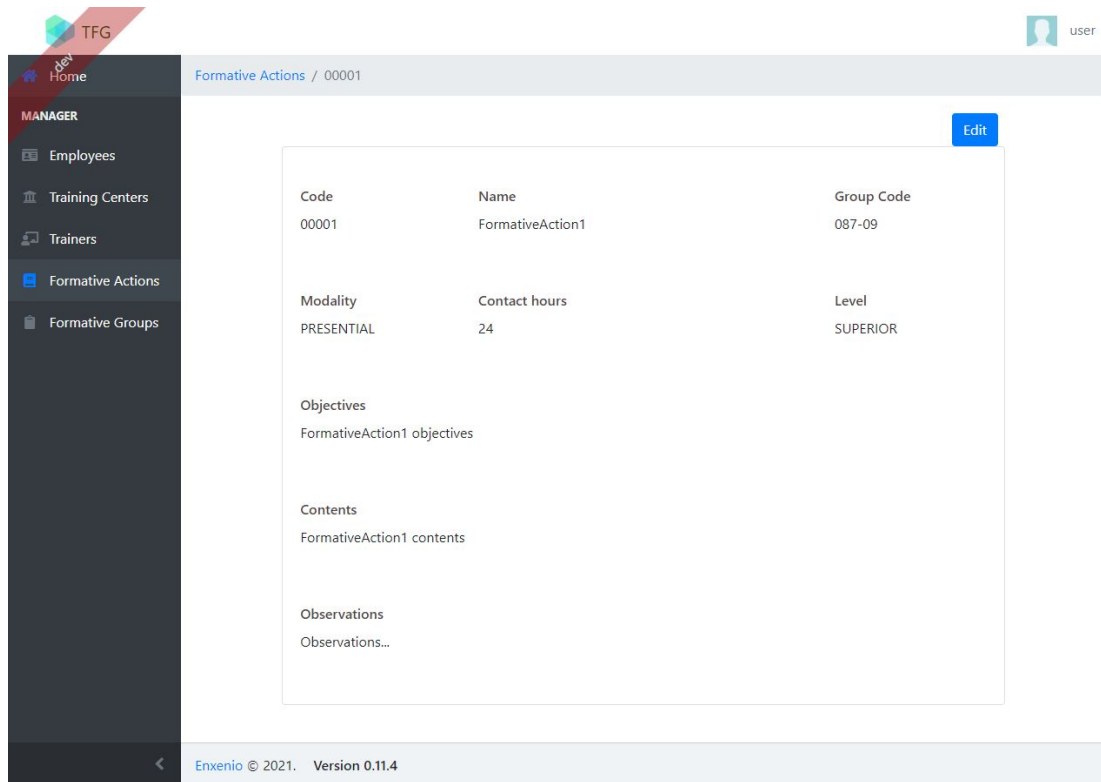


Figura A.15: Pantalla de formulario de Acción formativa

A.16 Listado de grupos formativos

Tras seleccionar a opción "Formative Groups" do menú lateral esquerdo, o usuario accederá a pantalla representada na figura A.16, onde pode visualizar o listado de grupos; crear un novo premendo o botón "+Add"; visualizar o detalle de calquera dos grupos formativos pulsando sobre cadanseu código; xerar os ficheiro XML tras seleccionar nos checkboxes da columna "To XML" os grupos que se queiran incluír en dito documento e logo premendo o botón "Generate Start XML" para xerar o ficheiro de inicio ou "Generate End XML" para o de fin.

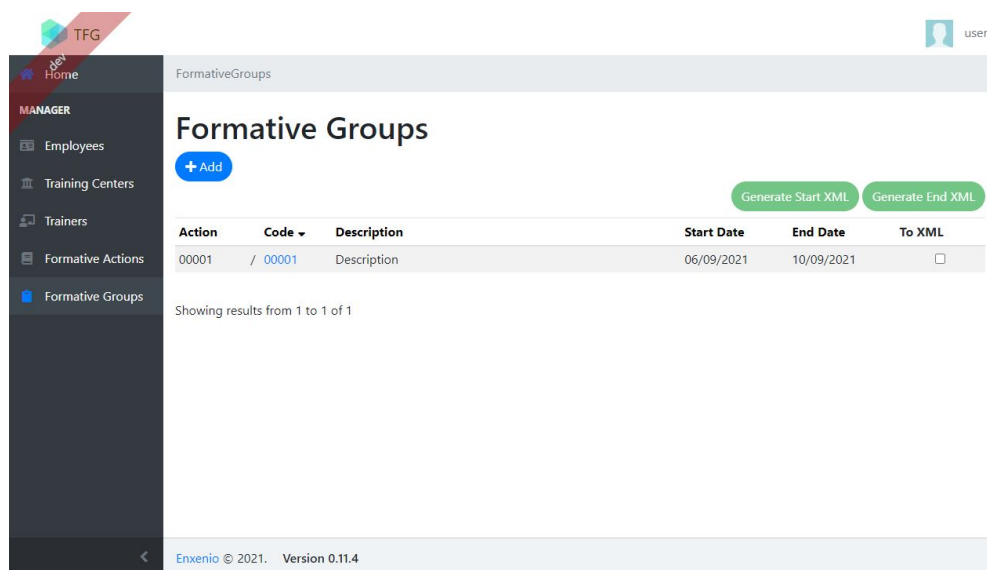


Figura A.16: Pantalla de listado de Grupos formativos

A.17 Formulario de grupo formativo

Tras seleccionar un grupo formativo do listado, o usuario accederá á pantalla mostrada na figura A.17, onde terá a opción de editar información do grupo premendo no botón "Edit" e xerar un ficheiro PDF para os partes de asistencia dos participantes do grupo premendo no botón "Generate PDF". Ademais, tamén se presentan uns botóns a modo de pestanas para mostrar a información do grupo formativo organizada en información xeral (mostrada na figura A.17) premendo en "General", información sobre os participantes (mostrada na figura A.18) premendo en "Participants" e información sobre os costes do grupo (mostrada na figura A.19) premendo en "Costs". No caso de que o usuario pulse o botón "+Add" na pantalla representada na figura A.16, accederá a unha pantalla similar á mostrada na figura A.17 pero en modo edición e sen ningún campo informado para que poida indicar a información do novo grupo a engadir e podendo moverse por todas as pestanas.

TFG

Home

Formative Groups / 00001

MANAGER

Generate PDF

Edit

General Participants Costs

Code	Formative Action	Start date
00001	FormativeAction1 (00001)	06-09-2021

Description	End date
Description	10-09-2021

Person in charge	Phone
Pedro Lamas Pazos	600000000

Presential Session

Presential Center
TrainingCenter1 (A00000000)

Delivery Place

CIF	Delivery Place Name	Address
F00000000	Auditorio	Calle Principal
Postal Code	City	Country
15007	A Coruña	Spain

Class schedule

Total hours	Days
12	LXV

Monday
 Tuesday
 Wednesday
 Thursday
 Friday
 Saturday
 Sunday

Start Time Section 1	End Time Section 1	Start Time Section 2	End Time Section 2
09:00	12:00	12:30	13:30

Presential Trainers

Presential Trainer	Total Hours
Trainer2 Trainer2 (234567898)	5
Trainer1 Trainer1 (12345678A)	7

Virtual Class

Environment	Connection
moodle	www.moodle.es
Contact	Phone
Manuel Mira	600112233

Enxenio © 2021. Version 0.11.4

Figura A.17: Pantalla de formulario para a primeira pestana de Grupo formativo

Formative Groups / 00001

Generate PDF

General Participants Costs

Nº of Participants: 4

Participants:

- Ana Abelenda (11111111A)
- Carla Cancela (33333333C)
- Brais Blanco (22222222B)
- Diego Deus (44444444D)

Exxenio © 2021. Version 0.11.4

Figura A.18: Pantalla de formulario para a segunda pestana de Grupo formativo

Formative Groups / 00001

Generate PDF

General Participants Costs

Direct Cost (€)	Direct Cost Accepted (€)	
1500	1500	
Indirect Cost (€)	Indirect Cost Accepted (€)	Wage Cost (€)
200	170	2000
Total Cost (€)	Total Cost Accepted (€)	Cost Limit (€)
1700	1670	468

Monthly bonus

Nº Month	Amount (€)	Total (€)
11	585	585

Exxenio © 2021. Version 0.11.4

Figura A.19: Pantalla de formulario para a terceira pestana de Grupo formativo

Relación de Acrónimos

BOE Boletín Oficial do Estado. 17

CSS Cascading Style Sheets. 11

DAO Data Access Object. 13, 20, 21, 26

DTO Data Transfer Object. 21, 23, 24

FUNDAE Fundación Estatal para la Formación en el Empleo. 1–3, 5, 33, 34, 36, 40, 42

HTML HyperText Markup Language. 11, 12, 21

HTTP HyperText Transfer Protocol. 13

JSON JavaScript Object Notation. 13

REST Representational State Transfer. 13

SSD Solid State Drive. 12

UML Unified Modeling Language. 14

XML Extensible Markup Language. 5, 7, 8, 13, 14, 17, 33–36, 40, 42

Bibliografía

- [1] “What is scrum?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.scrum.org/resources/what-is-scrum>
- [2] “¿qué es java?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: https://www.java.com/es/about/whatis_java.jsp
- [3] “What is typescript?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.typescriptlang.org/>
- [4] R. A. A. Altamirano, *HTML: Desarrollo web*. Independently published, 2021.
- [5] “Css with superpowers,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://sass-lang.com/>
- [6] “An introduction to latex,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.latex-project.org/about/>
- [7] “Getting started,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://code.visualstudio.com/docs>
- [8] “El navegador creado por google,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: https://www.google.com/intl/es_es/chrome/
- [9] “Accelerate developer productivity,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://gradle.org/>
- [10] “Spring makes java productive,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://spring.io/>
- [11] “Everything data.” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://hibernate.org/>

-
- [12] “About node.js,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://nodejs.org/en/about/>
- [13] “The modern web developer’s platform,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://angular.io>
- [14] “What are restful web services?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html>
- [15] D. Gourley, M. Sayer, A. Aggarwal, and S. Reddy, *HTTP: The Definitive Guide*. O’Reilly Media, 2002.
- [16] “What is postgresql?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.postgresql.org/about/>
- [17] “What is pgadmin 4?” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.pgadmin.org/faq/#1>
- [18] “–fast-version-control,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://git-scm.com/>
- [19] “Gitlab is the devops platform,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://about.gitlab.com/>
- [20] “Legendary git gui client for windows, mac and linux,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.gitkraken.com/>
- [21] “Documentation,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://es.overleaf.com/learn>
- [22] “Whitestaruml home page,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <http://whitestaruml.sourceforge.net/>
- [23] “Microsoft teams,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.microsoft.com/es-ww/microsoft-teams/group-chat-software>
- [24] M. D. E. Y. S. SOCIAL, “Xvii convenio colectivo estatal de empresas de consultoría y estudios de mercado y de la opinión pública.” consultado o 12 de setembro de 2021. [En liña]. Disponible en: [https://www.boe.es/eli/es/res/2018/02/22/\(3\)/dof/spa/pdf](https://www.boe.es/eli/es/res/2018/02/22/(3)/dof/spa/pdf)
- [25] W. Crawford and J. Kaplan, *J2EE Design Patterns: Patterns in the Real World*. O’Reilly Media, 2003.
- [26] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison Wesley, 1994.

BIBLIOGRAFÍA

- [27] “Apache poi - the java api for microsoft documents,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://poi.apache.org/>
- [28] “Package org.w3c.dom,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://docs.oracle.com/javase/7/docs/api/org/w3c/dom/package-summary.html>
- [29] “Discover itext pdf,” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://itextpdf.com/en>
- [30] “Free online xml validator (xsd),” consultado o 12 de setembro de 2021. [En liña]. Disponible en: <https://www.liquid-technologies.com/online-xsd-validator>

