

Gestor y visualizador web de datos hidráulicos

Héctor Gonzaga Andrés

Grado de Ingeniería Informática
JavaEE

Vicenç Font Sagrista
Santi Caballe Llobet

08/01/2019



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Gestor y visualizador web de datos hidráulicos</i>
Nombre del autor:	<i>Héctor Gonzaga Andrés</i>
Nombre del consultor/a:	<i>Vicenç Font Sagrista</i>
Nombre del PRA:	<i>Santi Caballe Llobet</i>
Fecha de entrega (mm/aaaa):	01/2019
Titulación::	<i>Grado de Ingeniería Informática</i>
Área del Trabajo Final:	<i>JavaEE</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>Spring JPA Rest</i>
<p>Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.</i></p>	
<p>El objetivo del proyecto es diseñar una herramienta web para gestionar y hacer accesible la información de una antigua base de datos compuesta al recopilar la información obtenida por sensores ubicados en una infraestructura de tuberías de abastecimiento de agua potable para unos equipos hidráulicos.</p> <p>Para ello se realiza un plan de trabajo para desarrollar una plataforma de gestión útil para manejar datos en un entorno web dotando al aplicativo con distintos roles de usuario y funcionalidades bien definidas para cada uno de ellos.</p> <p>La estrategia seguida para implementar la solución se basa en la arquitectura cliente-servidor, por ser un entorno web, y se desarrolla utilizando el Modelo-Vista-Controlador (MVC). En este contexto se usa una tecnología lo más actual posible como es el caso del framework Spring que permitirá hacer futuras modificaciones al aplicativo.</p> <p>La implementación con Spring, tecnología muy versátil y modular al estar diseñado con interfaces, permite al desarrollador la reutilización de código y la inserción de nuevas funcionalidades de forma relativamente sencilla. Además, la información que se extrae de la base datos en muchos casos es en formato JSON lo que facilitará la integración con otro tipo de tecnologías de desarrollo y componentes al tratarse de un estándar.</p> <p>Durante la elaboración del proyecto se han sopesado otras tecnologías de desarrollo finalmente descartadas pues eran menos reutilizables y versátiles que las finalmente implementadas. El producto final puede ser fácilmente extrapolable a otros ámbitos que también requieran la visualización y gestión de datos almacenados en una base de datos.</p>	

Abstract (in English, 250 words or less):

The aim of this project is to design a web tool to manage and make accessible the information of an old compound database by collecting this information obtained through sensors located in an infrastructure of potable water supply pipes for hydraulic equipment.

In order to do so, a work plan is put into practice to develop a useful management platform to manage data in a web environment providing the application with different user roles and well-defined functionalities for each of them.

As it is a web environment the strategy followed to implement the solution is based on the client-server architecture, and it is developed using the Model-View-Controller (MVC). In this context, the most up-to-date technology is used, as is the case with the Spring framework, which will allow future modifications to the application.

The implementation with Spring, which is a very versatile and modular technology, being designed with interfaces, allows the developer to reuse code and insert new functions in a relatively simple way. In addition, the information that is extracted from the database in many cases is in JSON format as it is a standard which will facilitate integration with other types of development technologies and components.

During the elaboration of the project, other development technologies have been weighed, and finally discarded because they were less reusable and versatile than the one finally implemented. The final product can be easily extrapolated to other areas that also require the visualization and management

of data stored in a database.

Índice

1. Introducción.....	8
1.1 Contexto y justificación del Trabajo.....	8
1.2 Requisitos del Trabajo.....	8
1.3 Objetivos del Trabajo.....	9
1.4 Enfoque y método seguido.....	10
1.6 Aproximación de herramientas a utilizar.....	14
1.7 Breve resumen de productos obtenidos.....	15
1.8 Breve descripción de los otros capítulos de la memoria.....	15
2. Punto de vista de la empresa.....	17
2.1 Comunidades.....	17
2.2 Objetos y roles.....	17
2.3 Acciones y procesos.....	17
2.4 Reglas de negocio y asignación de objetos a roles.....	18
2.5 Diagrama de casos de uso.....	19
2.6 Casos de uso definidos.....	20
2.7 Fichas de los casos de uso.....	21
2.8 Capturas de las funcionalidades de la aplicación.....	35
3. Punto de vista de la información.....	46
3.1 Esquemas invariantes.....	46
3.2 Esquemas dinámicos.....	46
3.3 Esquemas estáticos.....	47
4. Punto de vista de la computación.....	47
4.1 Diseño de alto nivel.....	47
4.2 Arquitectura del software detallada.....	49
5. Punto de vista de la ingeniería.....	55
6. Punto de vista de la tecnología.....	56
7. Conclusiones.....	58
8. Glosario.....	60
9. Bibliografía.....	62

Lista de figuras

Ilustración 1. Plan de Trabajo.....	12
Ilustración 2. Acciones y procesos de consultor.....	17
Ilustración 3. Acciones y procesos de administrador.....	18
Ilustración 4. Diagrama de casos de uso	19
Ilustración 5. CU_1_Login	35
Ilustración 6. CU_2_Logout.....	35
Ilustración 7. Ventana principal usuario consultor	36
Ilustración 8. Ventana principal usuario administrador.....	36
Ilustración 9. CU_3_RegisterUser y CU_3.4_ListadoUsuarios.....	36
Ilustración 10. CU_3.1_AltaUsuario	37
Ilustración 11. CU_3.2_BajaUsuario	37
Ilustración 12. CU_3.3_ModificarUsuario y Cu_4_ChangePassword.....	37
Ilustración 13. CU_5_EditEquipment.....	38
Ilustración 14. CU_5.1_EditName y CU_5.2_EditCoordenates.....	38
Ilustración 15. CU_6_EditChannel	38
Ilustración 16. CU_6.1_EditName y CU_6.2_EditEngineerinUnit.....	39
Ilustración 17. Ventana System Management usuario consultor.....	39
Ilustración 18. Ventana System Management usuario administrador.....	39
Ilustración 19. CU_7_ManagementAccounts y CU_7.4_ListAccounts	40
Ilustración 20. CU_7.1_AddAccount.....	40
Ilustración 21. CU_7.2_DeleteAccount.....	40
Ilustración 22. CU_7.3_ModifyAccount	41
Ilustración 23. CU_8_ManagementEquipments y CU_8.4_ListEquipments.....	41
Ilustración 24. CU_8.1_AddEquipment	41
Ilustración 25. CU_8.2_DeleteEquipment	42
Ilustración 26. CU_8.3_ModifyEquipment	42
Ilustración 27. CU_9_DisplayMap	42
Ilustración 28. CU_10_DisplayData Paso1	43
Ilustración 29. CU_10_DisplayData Paso2	43
Ilustración 30. CU_11_DisplayTreeOfAccountsAndEquipments	43
Ilustración 31. CU_12_Associations Management.....	44
Ilustración 32. CU_12.1_AssociateAccounts Paso1.....	44
Ilustración 33. CU_12.1_AssociateAccounts Paso2.....	45
Ilustración 34. CU_12.2_AssociateEquipmentToAccount	45
Ilustración 35. CU_12.3_AssociateAccountToUser.....	45
Ilustración 36. Esquemas invariantes.....	46
Ilustración 37. Funcionamiento Modelo-Vista-Controlador en Spring.....	48
Ilustración 38. Diagrama de componentes	49
Ilustración 39. Capa de Presentación Primer Nivel	51
Ilustración 40. Capa de Presentación Segundo Nivel	52
Ilustración 41. Capa de negocio	53
Ilustración 42. Capa de integración	54

1. Introducción

1.1 Contexto y justificación del Trabajo

Se dispone de una base de datos con registros antiguos de equipos hidráulicos compuesta al recopilar la información obtenida por sensores ubicados en una infraestructura de tuberías de abastecimiento de agua potable.

Actualmente los datos almacenados sólo pueden ser explotados mediante un gestor de base de datos o con una hoja de cálculo, por lo que se pretende facilitar el acceso a esta información en un entorno cómodo y accesible.

El objetivo final del proyecto será diseñar una herramienta web que gestione los datos existentes y que se pueda utilizar en un entorno web que no requiera más que el acceso a una URL por parte de los usuarios.

1.2 Requisitos del Trabajo

El proyecto a realizar consiste en el desarrollo de una aplicación web para la explotación gráfica y tabular de los datos, registrados por unos equipos hidráulicos, almacenados en una base de datos SQL.

El objetivo de la aplicación web será la de proporcionar acceso a los usuarios para comprobar el estado de los equipos hidráulicos y asociar los datos obtenidos a las cuentas de los clientes. La información se presentará utilizando gráficas y podrá ser ampliada por los usuarios añadiendo datos extra.

Cada usuario tendrá una cuenta que permitirá “visualizar” los equipos integrados. Cada cuenta podrá, a su vez, tener subcuentas anidadas pertenecientes a la misma región para que los usuarios puedan acceder a la información contenida en los equipos hidráulicos en dichas subcuentas.

La información que almacenarán los equipos será el nombre y la localización, teniendo asociados unos canales y unos datos. Los canales contienen un nombre de canal, un número de canal y una medida de unidad de ingeniería. Los datos estarán compuestos por un número de canal, una fecha, una hora y un valor.

Un usuario supervisor, por defecto, será el encargado de administrar la base de datos. Después de un primer acceso a la web modificará su contraseña y será el encargado de dar de alta a los demás usuarios de la aplicación, indicando un nombre y una contraseña, crear nuevas cuentas y asociar los equipos hidráulicos a éstas.

El acceso a la información de la aplicación web solo será posible para los usuarios registrados en el sistema por el supervisor. El acceso se realizará mediante la indicación de un nombre y una contraseña.

Cuando un usuario realice el “login” en el sistema podrá ver las cuentas a las que está asociado y tendrá la opción de seleccionar una de ellas para acceder a los datos de los equipos hidráulicos que contenga. Además se mostrarán las gráficas con los valores de los canales que tengan definidos. Los canales hacen referencia a las medidas registradas por los equipos y que serán el caudal y la presión.

La información de las cuentas y de los equipos se presentará en forma de árbol para facilitar la selección. Los usuarios podrán modificar los equipos y los canales que pertenezcan a su cuenta pudiendo editar, en el primer caso, el nombre y la ubicación y en el segundo caso, mediante la edición del nombre del canal y la unidad de ingeniería en la que se desea obtener los canales de éstos que pertenezcan a su cuenta. Las gráficas y tablas obtenidas mostrarán todos los datos asociados a los canales del equipo seleccionado por el usuario de rol consultor.

1.3 Objetivos del Trabajo

- Desarrollar una aplicación web para la explotación gráfica y tabular de los datos.
- Diseñar una base de datos relacional a partir de los datos existentes que permita cubrir los requisitos del proyecto.
- Crear una tabla de usuarios para permitir el acceso web.
- Creación de una tabla de equipos hidráulicos a los que asociar los registros antiguos. Esta tabla deberá permitir dar un nombre y una ubicación geográfica en forma de coordenadas (latitud y longitud).
- Crear una tabla de canales relacionada con los datos existentes y con los equipos que permita dotar de un nombre al canal y una unidad de medida para dotar de significado a los datos.
- Disponer de un árbol de cuentas a los que asociar los distintos usuarios y los datos de los equipos mediante tablas y relaciones.
- Creación de una tabla con perfiles de usuarios para asociar a los mismos.
- Crear una interfaz web para el “login” de los usuarios según su perfil (Administrador o Consultor) mediante la introducción de un nombre de usuario y una contraseña.
- Crear un perfil de usuario “Administrador” que gestione la base de datos y un perfil de usuario “Consultor” que pueda visualizar los datos que el administrador ponga a su disposición.
- Crear una interfaz web que permita el manejo de la base de datos por parte del usuario Administrador.
- Crear un componente en una interfaz web de usuario Administrador que permita la selección de las subcuentas y sus equipos en forma de árbol.
- Crear un componente en una interfaz web para mostrar en forma tabular la información de un equipo seleccionado por un usuario Consultor.

- Crear un componente gráfico en una interfaz web que permita la visualización del equipo seleccionado por el usuario Consultor.
- Crear un componente integrado en la interfaz web del usuario Consultor que permita la visualización de la ubicación de los equipos en un mapa.
- Crear un componente integrado en la interfaz web del usuario Consultor que permita la introducción y la edición de los campos nombre y ubicación (latitud y longitud) de los equipos.
- Crear un componente integrado en la interfaz web del usuario Consultor que permita la introducción y la edición de los campos nombre de canal y unidad de ingeniería de los canales de los equipos.

1.4 Enfoque y método seguido

Se han evaluado dos posibles estrategias para llevar a cabo el desarrollo del proyecto:

En primer lugar, se ha planteado la opción de adaptar la base de datos para ser gestionada mediante una arquitectura cliente-servidor de manera que la base de datos estuviera accesible en “la parte servidora” y se accediera a la misma a través de una aplicación local que realizará consultas vía web para obtener la información necesaria.

En segundo lugar, se ha planteado también una arquitectura cliente-servidor pero en este caso se generará una nueva base de datos de tipo DBMS para que pueda ser accesible desde una aplicación web.

Se ha optado por la segunda opción ya que se desea que la información de los equipos hidráulicos sea accesible por diferentes tipos de cliente y lo más rápido y eficiente para su desarrollo es hacerlo vía web. De esta manera sólo es necesario disponer de un navegador web para poder ejecutar la aplicación web y no sería necesaria la instalación de ningún software específico en el puesto cliente.

En definitiva, el proyecto según los requisitos anteriores, se desarrollará en cuatro apartados principales:

- Administración del sistema: “autenticación” y otras operaciones que puede hacer el administrador del sistema. Esto incluye la creación de nuevos usuarios y nuevas cuentas así como la asociación de los equipos a las cuentas existentes.
- Edición de los equipos y los canales: un usuario registrado podrá editar los distintos campos que contienen los equipos referentes al nombre y su ubicación. Del mismo modo, podrá editar el nombre de los canales y la unidad de ingeniería en la que se mostrarán.
- Visualización gráfica y tabular de los canales asociados a un equipo.

- Visualización y ubicación, sobre un mapa, de los equipos de una cuenta usando la API de Google Maps. Aunque actualmente se exige la creación una cuenta de Google y la posesión de una clave para poder usar la API, se puede seguir usando (sólo) en modo “desarrollador”. Esto implica que el mapa aparece sombreado y muestra una marca de agua en la que se puede leer “sólo desarrollo”.

1.5 Planificación del Trabajo

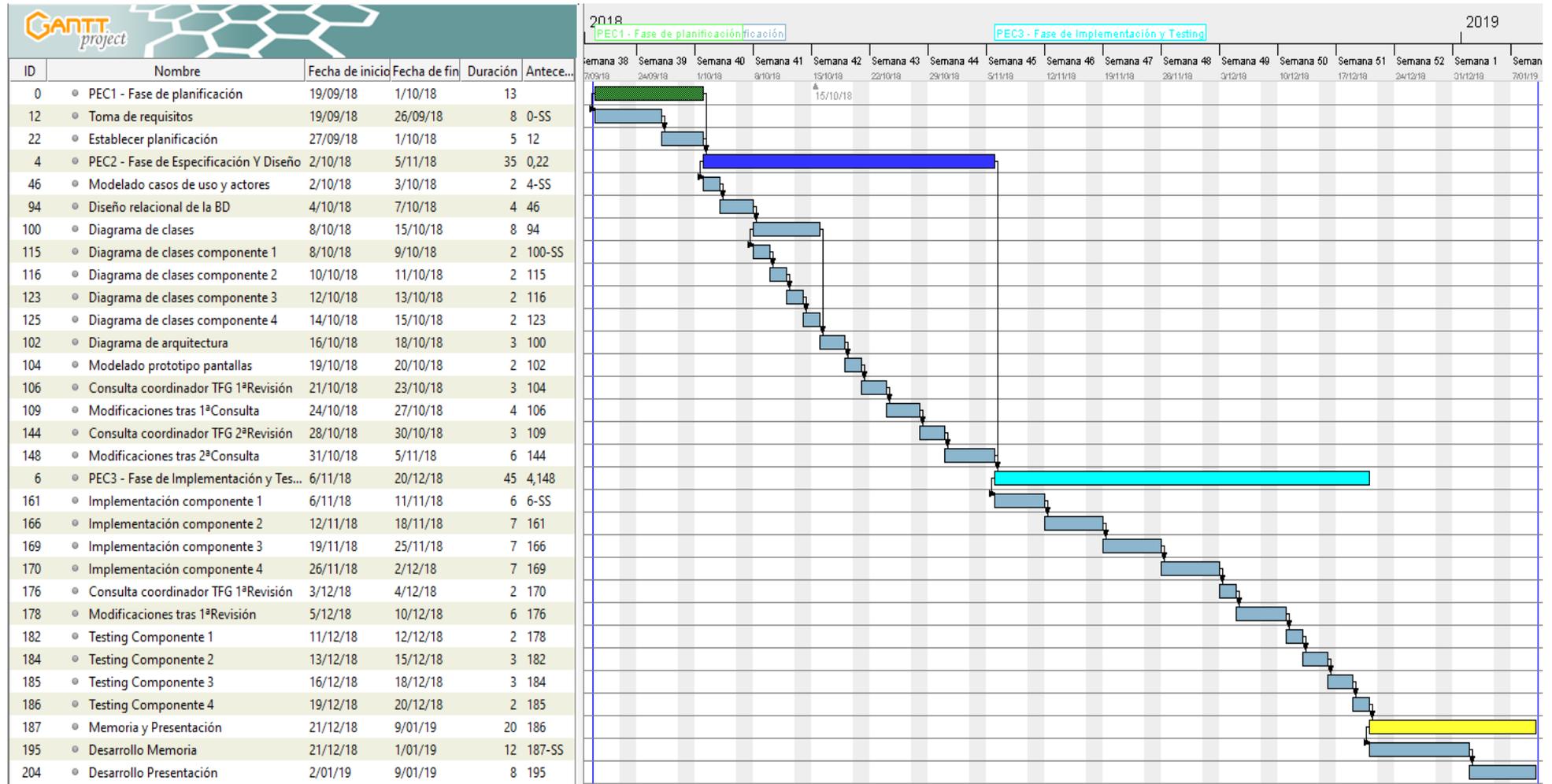


Ilustración 1. Plan de Trabajo

PEC1: Fase de planificación

- Toma de requisitos → redactar qué es lo que hay que implementar para llevar a cabo la solución al problema planteado.
- Establecer Planificación → establecer los tiempos y las tareas que se van a realizar en función de los requisitos planteados.

PEC2: Fase de especificación y diseño

- Modelado casos de uso y actores → realizar un diagrama de casos de uso que defina las funcionalidades del proyecto.
- Diseño relacional de la BD → diseñar la base de datos con sus tablas y sus relaciones que pueda dotar al proyecto de las funcionalidades requeridas.
- Diagrama de clases → crear el diagrama de clases que se usarán en el proyecto. Esta tarea se subdivide en la creación del diagrama de clases de los distintos componentes.
 - Diagrama de clases componente 1 → diagrama de clases del componente 1 correspondiente al módulo de Administración.
 - Diagrama de clases componente 2 → diagrama de clases del componente 2 correspondiente al módulo de Edición de los equipos hidráulicos y los canales habilitados.
 - Diagrama de clases componente 3 → diagrama de clases del componente 3 correspondiente al módulo de Visualización gráfica y tabular.
 - Diagrama de clases componente 4 → diagrama de clases del componente 4 correspondiente al módulo de Visualización y ubicación.
- Diagrama de arquitectura,
- Modelado prototipo pantallas → realización del prototipado de las pantallas de la aplicación web que formarán el proyecto.
- Consulta coordinador TFG 1ª revisión → mandar al coordinar el trabajo realizado así como las dudas que hayan surgido durante la realización del mismo.
- Modificaciones tras 1ª consulta → modificar el trabajo realizado hasta el momento en función de las respuestas obtenidas del coordinador.
- Consulta coordinador TFG 2ª revisión → mandar al coordinar el trabajo modificado siguiendo las indicaciones que haya sugerido en la primera revisión.
- Modificaciones tras 2ª consulta → adecuar el contenido del trabajo realizado en función de los cambios sugeridos por el coordinador.

PEC3: Fase de implementación y testing

- Implementación componente 1 → implementación del módulo de Administración.
- Implementación componente 2 → implementación del módulo de Edición de los equipos y los canales.
- Implementación componente 3 → implementación del módulo de Visualización gráfica y tabular.

- Implementación componente 4→ implementación del módulo de Visualización y ubicación.
- Consulta coordinador TFG 1ª revisión→mostrar la implementación realizada del TFG.
- Modificaciones tras 1ª revisión→ realizar las modificaciones sugeridas en la implementación de los componentes que integran el TFG.
- Testing Componente 1→ realización de las pruebas necesarias para la validación del componente de Administración en función de la especificación de requerimientos.
- Testing Componente 2→ realización de las pruebas necesarias para la validación del componente de Edición de los equipos y los canales en función de la especificación de requerimientos.
- Testing Componente 3→ realización de las pruebas necesarias para la validación del componente de Visualización gráfica y tabular en función de la especificación de requerimientos.
- Testing Componente 4→ realización de las pruebas necesarias para la validación del componente de Visualización y ubicación en función de la especificación de requerimientos.

Memoria y Presentación:

- Desarrollo Memoria →crear documento que detalla la metodología seguida para desarrollar el proyecto TFG y los resultados de los procesos de planificación, análisis, diseño e implementación.
- Desarrollo Presentación → generación de la presentación en Powerpoint y del vídeo que muestra la funcionalidad del proyecto TFG desarrollado.

1.6 Aproximación de herramientas a utilizar

Se ha decidido utilizar el framework Spring y en concreto el módulo Spring Boot que proporciona una gestión automatizada de la infraestructura software que necesita la aplicación para funcionar. Esta tecnología se centra en la configuración e integración de las dependencias que necesita de forma genérica el repositorio Maven para que sin esfuerzo de comienzo el desarrollo con el IDE escogido, en este caso Eclipse y además integra (configura de forma embebida) un servidor Tomcat o Jetty para facilitar el despliegue de la aplicación.

Bajo esta tecnología se ha realizado el acceso a los datos mediante JPA a una base de datos MySQL para lo cual es necesario modificar el archivo pom.xml agregando las dependencias de JPA y MySQL.

Para el aplicativo web se empleará HTML5 y JavaEE junto con un motor de plantillas llamado Thymeleaf, también se utilizará un componente de licencia BSD D3.js basado en Javascript para la presentación gráfica y se hará uso de la Api de Google Maps para mostrar los equipos hidráulicos de una cuenta en un mapa. Como también se desea que el usuario administrador disponga de

una visualización en modo árbol, de las cuentas y de los equipos se usará el componente `TreeNode.js`.

Para devolver la información desde la base de datos al componente Javascript se utilizará el formato de intercambio de datos JSON la cual es facilitada por framework Spring seleccionado ya que nos abstrae de la creación de un servicio web en “la parte servidora” para ofrecer los datos al cliente mediante el uso de RestComponents. Este mismo formato se utilizará para modificar la base de datos y su gestión por parte del usuario supervisor.

De esta manera se ha llevado a cabo el proyecto mediante el uso combinado de Spring, JPA y REST.

1.7 Breve resumen de productos obtenidos

El resultado final del proyecto está formado por los siguientes entregables:

La presente memoria en la que se detalla la metodología seguida para el desarrollo del proyecto TFG, así como los resultados de los procesos de planificación, análisis, diseño e implementación.

El software desarrollado, diseñado en JavaEE que incluye la implementación de las funcionalidades detectadas y analizadas.

El vídeo `PresentacionFuncionalidadTFG.avi`, donde se muestra el funcionamiento de la aplicación resultante con la finalidad de facilitar la evaluación del proyecto desarrollado.

La presentación `PresentaciónTFG.pps` destinada a servir de síntesis del desarrollo total del proyecto.

El vídeo `PresentacionPowerPointTFG.avi`, donde se muestra el funcionamiento de la aplicación resultante con la finalidad de facilitar la evaluación del proyecto desarrollado.

1.8 Breve descripción de los otros capítulos de la memoria

Para el desarrollo del proyecto he seguido el modelo de referencia para el procesamiento abierto y distribuido RM-ODP (Reference Model – Open Distributed Processing) que define cinco puntos de vista a la hora de definir un sistema.

El capítulo dos se basa en el punto de vista de la empresa que se centra en los aspectos propios del negocio en donde se va a integrar el sistema como son el objetivo, alcance, fichas de casos de uso, prototipo de pantallas y campo de aplicación del proyecto.

El capítulo tres se basa en el punto de vista de la información que describe un modelo conceptual de los datos que va a manejar el sistema implementado.

El capítulo cuatro se basa en el punto de vista de la computación que se centra en la funcionalidad y estructuración del sistema en función de una arquitectura software determinada.

El capítulo cinco se basa en el punto de vista de la ingeniería que se centra en los aspectos relativos a la distribución y procesamiento de los datos por parte del sistema.

El capítulo seis se basa en el punto de vista de la tecnología que describe de forma concreta el software, hardware y comunicaciones que se van a utilizar para implementar el sistema.

El capítulo siete hace referencia a las conclusiones obtenidas al desarrollar el sistema planteado y los objetivos conseguidos.

El capítulo ocho se definen los términos y acrónimos más relevantes utilizados dentro de la memoria.

El capítulo nueve se listan las referencias bibliográficas consultadas para llevar a cabo el proyecto incluidas en la memoria.

2. Punto de vista de la empresa

2.1 Comunidades

Se ha identificado una comunidad, que puede ser una compañía de aguas o una empresa dedicada a la explotación hidráulica que desee visualizar los datos registrados por los equipos hidráulicos instalados vía on-line.

2.2 Objetos y roles

Los objetos definidos serán: usuario, cuenta, equipo, canal y datos.

El objeto usuario puede desempeñar los roles de consultor y administrador.

El objeto cuenta toma el rol de cuenta. Tiene asociados usuarios y equipos. Esta asignación determinará los usuarios que pueden acceder a determinados equipos.

El objeto equipo tendrá el rol de equipo, teniendo asociados unos canales y los datos que se recogen.

El objeto canal desempeña el rol de canal y tendrá asociados los datos almacenados correspondientes a las medidas tomadas.

El objeto datos toma el rol de datos y constituye el conjunto de valores de las medidas hidráulicas recogidas.

2.3 Acciones y procesos

El rol de usuario consultor puede editar el nombre y las coordenadas de los equipos hidráulicos pertenecientes a su cuenta, editar el nombre y unidad de ingeniería de los canales de los equipos de su cuenta y visualizar los equipos en un mapa y los datos que almacenan los canales de los mismos por lo que interactúa con el rol de equipo, de canal y de datos.

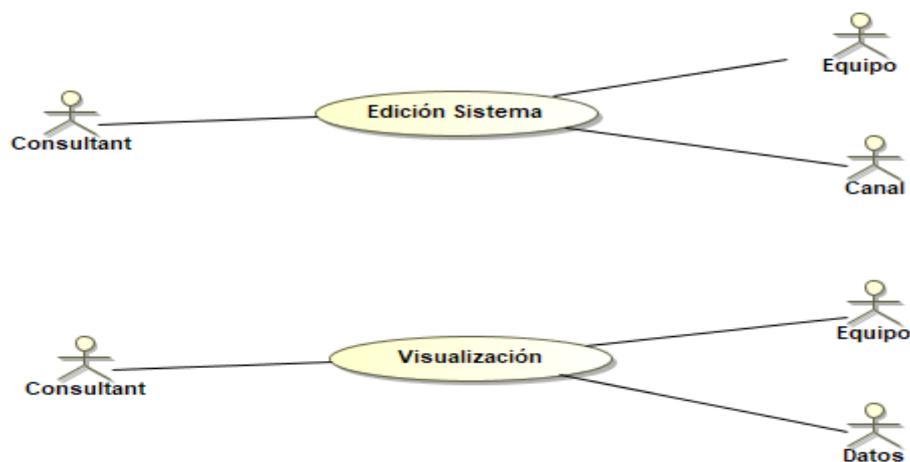


Ilustración 2. Acciones y procesos de consultor

El rol de usuario administrador gestiona el sistema mediante la gestión de las cuentas, los equipos y los usuarios y también puede visualizar el árbol de cuentas y equipos asociados junto con los usuarios que disponen de acceso a las mismas, por lo que interactúa con el rol de cuenta, de equipo y de usuario.

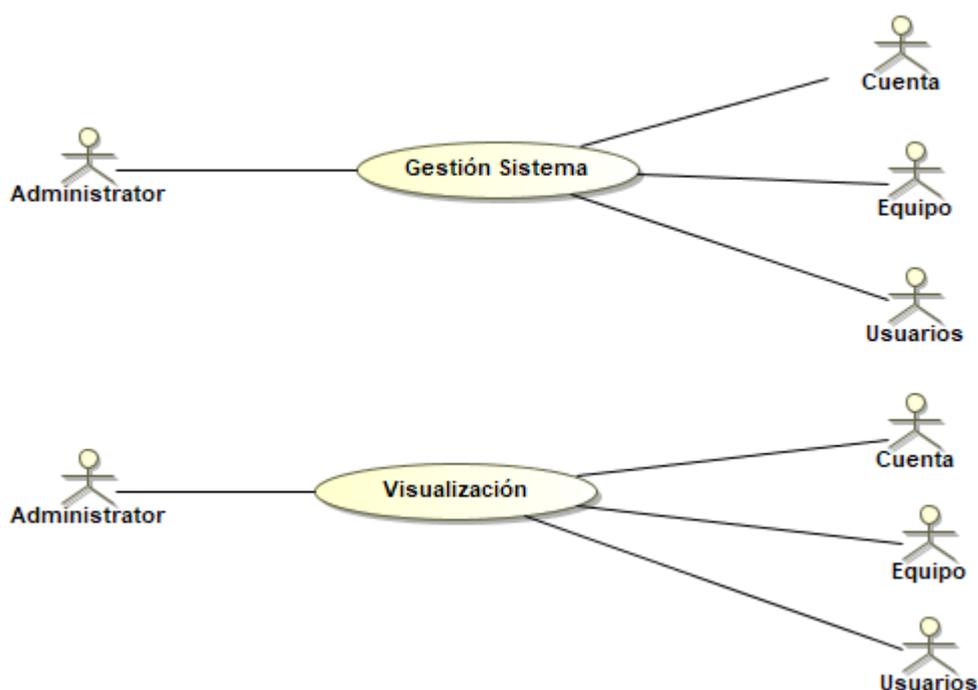


Ilustración 3. Acciones y procesos de administrador

2.4 Reglas de negocio y asignación de objetos a roles

El objeto usuario-consultor tiene permisos para:

- editar el nombre y las coordenadas de los equipos definidos para su cuenta.
- editar el nombre y la unidad de ingeniería de los canales de los equipos pertenecientes a su cuenta.
- visualizar los equipos asociados a su cuenta en un mapa.
- visualizar los datos recogidos por los distintos canales de los equipos asociados a su cuenta.

El objeto usuario-administrador tiene permisos para:

- gestionar las cuentas (alta, baja, modificación y consulta).
- gestionar los equipos hidráulicos (alta, baja, modificación y consulta).
- gestionar los usuarios (alta, baja, modificación y consulta).
- gestionar las asociaciones entre cuentas (asociar y desasociar).

- gestionar las asociaciones entre los equipos definidos y las cuentas (asociar y desasociar).
- gestionar las asociaciones de las cuentas definidas en el sistema con los usuarios (asociar y desasociar).
- visualización de todas las cuentas definidas en el sistema.
- visualización de los equipos hidráulicos asociados a cada usuario.

2.5 Diagrama de casos de uso

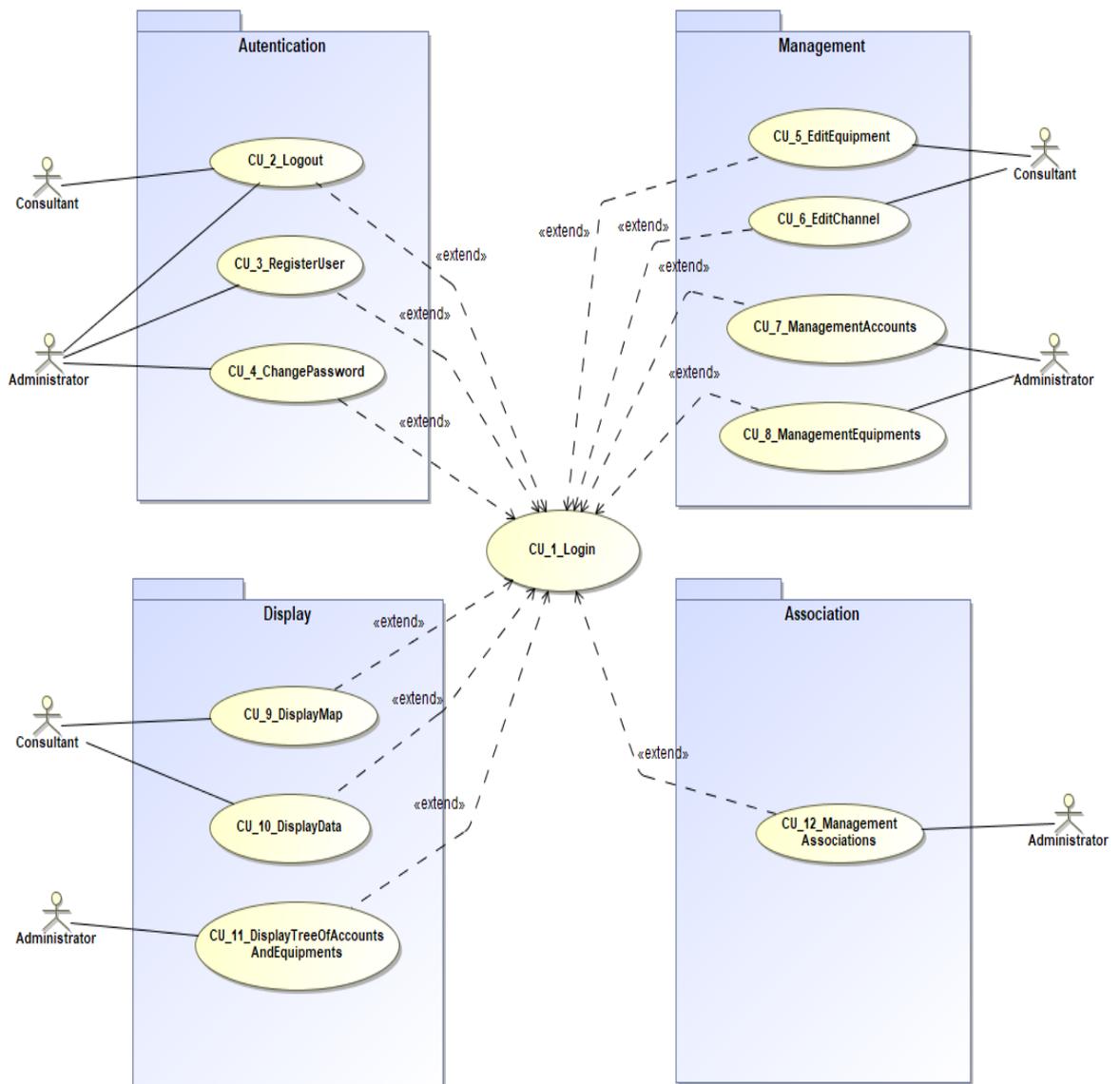


Ilustración 4. Diagrama de casos de uso

2.6 Casos de uso definidos

CU_1_Login

CU_2_Logout

CU_3_RegisterUser

CU_3.1_AddUser

CU_3.2_DeleteUser

CU_3.3_ModifyUser

CU_3.4_ListUsers

CU_4_ChangePassword

CU_5_EditEquipment

CU_5.1_EditName

CU_5.2_EditCoordinates

CU_6_EditChannel

CU_6.1_EditName

CU_6.2_EditEngineeringUnit

CU_7_ManagementAccounts

CU_7.1_AddAccount

CU_7.2_DeleteAccount

CU_7.3_ModifyAccount

CU_7.4_ListAccounts

CU_8_ManagementEquipments

CU_8.1_AddEquipment

CU_8.2_DeleteEquipment

CU_8.3_ModifyEquipment

CU_8.4_ListEquipments

CU_9_DisplayMap

CU_10_DisplayData

CU_11_DisplayTreeOfAccountsAndEquipments

CU_12_ManagementAssociations

CU_12.1_AssociateAccounts

CU_12.2_AssociateEquipmentToAccount

CU_12.3_AssociateAccountToUser

2.7 Fichas de los casos de uso

RF- 01	Login	
Objetivos asociados	CU_1_Login	
Requisitos asociados	Acceso al sistema	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario solicite al acceso al sistema.	
Precondición	El solicitante tiene que ser un usuario registrado.	
Secuencia Normal	Paso	Acción
	1	El usuario ingresa su nombre y contraseña en el formulario de acceso a la aplicación web y pulsa el botón de acceso.
	2	El sistema valida que el nombre y la contraseña corresponden con los datos de un usuario registrado.
	3	El sistema muestra la ventana principal de la aplicación donde se facilitará el acceso a las funcionalidades que tiene activas el usuario en función del rol consultor o administrador que disponga.
Postcondición	El usuario accede al sistema para realizar las acciones que tengan habilitadas en función de su rol de usuario.	
Excepciones	Paso	Acción
	3	Si la información aportada por el usuario en el formulario de ingreso a la aplicación web no es correcto el sistema devolverá un mensaje de error indicando el problema detectado y mostrará la ventana de login de nuevo.
Rendimiento	Paso	Cota de tiempo
	2	3 segundos
Frecuencia esperada	Cada vez que el usuario quiera acceder al sistema.	
Estabilidad	alta	
Comentarios	La frecuencia dependerá del uso que realicen los usuario de la aplicación.	

RF- 02	Logout	
Objetivos asociados	CU_2_Logout	
Requisitos asociados	Salir del sistema.	
Descripción	Esta funcionalidad estará activa desde cualquier ventana de la aplicación web.	
Precondición	El solicitante es un usuario registrado y está logado en el sistema.	
Secuencia	Paso	Acción

Normal	1	El usuario pulsa el botón de cerrar sesión.
	2	El sistema cierra la sesión del usuario y le presenta la ventana de login.
Postcondición	El usuario sale de la aplicación web.	
Excepciones	Paso	Acción
Rendimiento	Paso	Cota de tiempo
	2	4 segundos
Frecuencia esperada	Cada vez que el usuario quiera salir de la aplicación web.	
Estabilidad	Alta	
Comentarios	La frecuencia dependerá del uso que realicen los usuarios de la aplicación.	

RF- 03	RegisterUser	
Objetivos asociados	CU_3_RegisterUser – Alta Usuario	
Requisitos asociados	El usuario ya se ha logado en el sistema para consultar la información del registro de usuarios.	
Descripción	Desde esta funcionalidad se podrá gestionar el alta, baja, modificación y listado de los usuarios registrados.	
Precondición	El solicitante es un usuario registrado y logado en el sistema con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú principal la opción de gestión de usuarios.
	2	El sistema muestra en el menú desplegable las opciones de registrar usuario y cambiar contraseña.
	3	El usuario selecciona la opción de registrar usuario.
	4	El sistema muestra un listado con los usuarios dados de alta en una tabla donde se incluyen las funcionalidades de edición y borrado del usuario. Encima del listado se muestra un botón de Nuevo para añadir un nuevo usuario al sistema.
	5	El usuario pulsa el botón de Nuevo usuario.
	6	El sistema muestra un formulario con los campos nombre, contraseña, rol e idCuenta para que el usuario los cumplimente y después pulse el botón aceptar para que el sistema lo almacene.
	7	El usuario rellena el formulario y pulsa el botón aceptar.
	8	El sistema valida la información introducida en el formulario y en caso de ser correcta le presentará la ventana principal de funcionalidades.

Postcondición	El solicitante añade un nuevo usuario al sistema.	
Excepciones	Paso	Acción
	8	La información introducida por el usuario es incorrecta se muestra un mensaje de error con el problema encontrado y se seguirá mostrando la ventana de alta de usuario.
Rendimiento	Paso	Cota de tiempo
	4	3 segundos
	8	5 segundos
Frecuencia esperada	Cada vez que se quiera dar de alta un usuario en la aplicación web.	
Estabilidad	Alta	
Comentarios		

RF- 04	ChangePassword	
Objetivos asociados	CU_4_ChangePassword	
Requisitos asociados	Cambio de la contraseña de acceso a la aplicación web.	
Descripción	El usuario de rol administrador modifica la contraseña de acceso a la aplicación web.	
Precondición	El solicitante es un usuario registrado y de rol administrador.	
Secuencia Normal	Paso	Acción
	1	El usuario selecciona en el menú principal la opción de gestión de usuarios.
	2	El sistema muestra en el menú desplegable las opciones de registrar usuario y cambio de contraseña.
	3	El usuario selecciona la funcionalidad de cambio de contraseña.
	4	El sistema le muestra una ventana con un formulario con dos campos para que introduzca dos veces la nueva contraseña y pulse el botón aceptar.
	5	El usuario rellena el formulario y pulsa el botón aceptar.
	6	El sistema valida que las contraseñas introducidas son iguales y válidas para presentar seguidamente la ventana principal de su rol de usuario administrador.
Postcondición	El usuario de rol administrador modifica la contraseña de acceso a la aplicación web.	
Excepciones	Paso	Acción
	6	Las contraseñas introducidas por el usuario no son válidas y el sistema informa al usuario del problema encontrado y sigue mostrando la ventana de cambio de contraseña.

Rendimiento	Paso	Cota de tiempo
	4	3 segundos
	6	5 segundos
Frecuencia esperada		
Estabilidad		
Comentarios	El usuario de rol administrador cambiará al menos una vez la contraseña que se le otorga por defecto al darle de alta en el sistema.	

RF- 05	EditEquipment	
Objetivos asociados	CU_5_EditEquipment – Edición del nombre del equipo	
Requisitos asociados	Edición de los equipos hidráulicos	
Descripción	El solicitante desea modificar el nombre o las coordenadas de longitud y latitud del equipo hidráulico.	
Precondición	El solicitante es un usuario registrado con el rol de consultor.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el usuario de rol consultor el usuario selecciona en el menú principal la funcionalidad de edición de sistema.
	2	El sistema presenta desde el menú desplegable las opciones de edición de los equipos y de los canales.
	3	El usuario selecciona la opción de editar equipos.
	4	El sistema presenta una nueva ventana con un listado de los equipos asociados a la cuenta del usuario con las funcionalidades de edición de nombre y edición de coordenadas.
	5	El usuario selecciona la opción de editar el nombre del equipo hidráulico.
	6	El sistema presenta una nueva ventana con un formulario con un campo lista para seleccionar el equipo a modificar, un campo de inserción para el nuevo nombre y un botón aceptar.
	7	El usuario selecciona el equipo a modificar, inserta un nuevo nombre y pulsa el botón de aceptar.
8	El sistema valida que el nombre sea alfanumérico y muestra un mensaje indicando que la información ha sido almacenada correctamente.	

Postcondición	El nombre del equipo hidráulico es modificado	
Excepciones	Paso	Acción
	8	El sistema valida que el nombre introducido por el usuario no es un valor alfanumérico o que ya existe un nombre de equipo con dicho nombre por lo que informa al usuario con un mensaje de error especificando el problema encontrado y sigue mostrando la misma ventana de edición de nombre de equipo hidráulico.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	2 segundos
	6	2 segundos
	8	4 segundos
Frecuencia esperada	A voluntad del usuario.	
Estabilidad	Alta	
Comentarios		

RF- 06	EditChannel	
Objetivos asociados	CU_6_EditChannel – Edición de la unidad de ingeniería	
Requisitos asociados	Edición de los canales de los equipos hidráulicos	
Descripción	El solicitante desea modificar la unidad de ingeniería del canal de un equipo hidráulico determinado.	
Precondición	El solicitante es un usuario registrado con rol de consultor.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el usuario de rol consultor el usuario selecciona la funcionalidad de edición de sistema.
	2	El sistema presenta desde el menú desplegable las opciones de edición de los equipos y de los canales.
	3	El usuario selecciona la opción de editar canales.
	4	El sistema presenta una nueva ventana con un listado de los canales de los equipos asociados a la cuenta del usuario donde puede seleccionar la funcionalidad de modificar el nombre del canal y la funcionalidad de modificar la unidad de ingeniería en la que medirá el canal.
	5	El usuario selecciona la opción de modificar la unidad de ingeniería.

	6	El sistema presenta una nueva ventana con un formulario compuesto por una lista desplegable de los equipos asociados a la cuenta del usuario para que seleccione el que desea modificar y un botón de aceptar.
	7	El usuario selecciona el equipo del cual quiere modificar la unidad de ingeniería de los canales que contiene y pulsa el botón aceptar.
	8	El sistema presenta un formulario donde se especifica en forma de tabla los canales que contiene el equipo hidráulico indicando su nombre y la unidad de ingeniería junto con unos enlaces para editar el campo que desee modificar.
	9	El usuario selecciona el link de edición de la unidad de ingeniería de un canal concreto.
	10	El sistema presenta una nueva ventana donde se indica el nombre del canal y del equipo seleccionado junto con una lista desplegable de las posibles medidas a aplicar y un botón aceptar. Tras seleccionar la unidad que desee se pulsa el botón aceptar.
	11	El sistema valida la opción seleccionada y almacena la información para dicho canal. A continuación se presenta de nuevo la ventana que incluye la tabla de canales disponibles para el equipo seleccionado.
Postcondición	La unidad de ingeniería del canal del equipo seleccionado es modificada.	
Excepciones	Paso	Acción
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	2 segundos
	6	2 segundos
	8	2 segundos
	10	4 segundos
Frecuencia esperada	A voluntad del usuario.	
Estabilidad	Alta	
Comentarios		

RF- 07	ManagementAccounts
Objetivos asociados	CU_7_ManagementAccounts – Añadir Cuenta
Requisitos asociados	Añadir una cuenta en el sistema
Descripción	Desde esta funcionalidad se podrá gestionar el alta, baja, modificación y listado de las cuentas disponibles.

Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el rol de administrador el usuario selecciona en el menú principal la opción de gestionar sistema.
	2	El sistema muestra en un menú desplegable las opciones de gestión de cuentas, gestión de equipos y la gestión de asociaciones sólo disponibles para el rol de administrador.
	3	El usuario selecciona la opción de gestión de cuentas.
	4	El sistema muestra en una nueva ventana un listado de las cuentas disponibles en una tabla con las opciones baja y modificación y en un botón aparte la funcionalidad de añadir un nueva cuenta al sistema.
	5	El usuario pulsa el botón para dar de alta una nueva cuenta.
	6	El sistema presenta un formulario con un cuadro de texto que corresponde al nombre de la cuenta y un botón aceptar.
	7	El usuario introduce un nombre de cuenta y pulsa el botón aceptar.
	8	El sistema valida que el valor introducido es alfanumérico y que no existe ninguna otra cuenta con el mismo nombre y en caso afirmativo, muestra la ventana de gestión de cuentas.
Postcondición	La nueva cuenta es dada de alta en el sistema.	
Excepciones	Paso	Acción
	8	El sistema valida que el nombre introducido por el usuario no es un valor alfanumérico o que ya existe una cuenta con dicho nombre por lo que informa al usuario con un mensaje de error especificando el problema encontrado y sigue mostrando la misma ventana de alta de cuenta.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	2 segundos
	6	3 segundos
	8	4 segundos
Frecuencia esperada	Cada vez que se requiera introducir en el sistema una nueva cuenta.	
Estabilidad	Alta	
Comentarios		

RF- 08	Management Equipment	
Objetivos asociados	CU_8_ManagementEquipment – Alta equipo	
Requisitos asociados	Añadir equipo hidráulico en el sistema.	
Descripción	Desde esta funcionalidad se podrá gestionar el alta, baja, modificación y listado de los equipos hidráulicos.	
Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el rol de administrador el usuario selecciona en el menú principal la opción de gestionar sistema.
	2	El sistema muestra en un menú desplegable las opciones de gestión de cuentas, gestión de equipos y la gestión de asociaciones sólo disponibles para el rol de administrador.
	3	El usuario selecciona la opción de gestión de equipos.
	4	El sistema muestra en una nueva ventana un listado de las equipos disponibles en una tabla con las opciones baja y modificación y en un botón aparte la funcionalidad de añadir un nuevo equipo al sistema.
	5	El usuario selecciona la opción de alta de equipo.
	6	El sistema presenta un formulario con varios cuadros de texto que corresponden al nombre del equipo, la latitud y la longitud junto con un botón aceptar.
	7	El usuario rellena el formulario y pulsa el botón aceptar.
	8	El sistema valida que el valor introducido para el nombre es alfanumérico y que no existe ningún otra equipo con el mismo nombre y que las coordenadas sean de tipo float y en caso afirmativo, muestra la ventana de gestión de equipos de nuevo.
Postcondición	Se incorpora un nuevo equipo hidráulico al sistema.	
Excepciones	Paso	Acción
	8	En caso negativo, el sistema informa al usuario mediante un mensaje de la anomalía detectada y se sigue mostrando la ventana de alta de equipos.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	2 segundos
	6	3 segundos
	8	4 segundos

Frecuencia esperada	Cada vez que se requiera introducir en el sistema un nuevo equipo hidráulico.	
Estabilidad	Alta	
Comentarios		
RF- 09	DisplayMap	
Objetivos asociados	CU_9_DisplayMap	
Requisitos asociados	Mostrar los equipos en un mapa.	
Descripción	Desde esta funcionalidad se presentarán en un mapa todos los equipos hidráulicos asociados a la cuenta del usuario.	
Precondición	El solicitante es un usuario registrado con el rol de consultor.	
Secuencia Normal	Paso	Acción
	1	Desde el menú principal disponible para el rol consultor el usuario selecciona la funcionalidad de visualización.
	2	El sistema presenta en el menú desplegable las funcionalidades de visualización de mapa y visualización de datos.
	3	El usuario selecciona la opción de visualizar mapa.
	4	El sistema presenta una nueva ventana con un mapa con todos los equipos hidráulicos asociados a la cuenta del usuario y un botón para volver a la ventana anterior.
Postcondición	Se presenta un mapa con todos los equipos asociados a la cuenta del usuario.	
Excepciones	Paso	Acción
	4	En caso de que el usuario no disponga de equipos hidráulicos asociados a la cuenta se mostrará al usuario que no hay equipos definidos para esa cuenta y se volverá a la ventana de selección de equipo.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	5 segundos
Frecuencia esperada	Cada vez que el usuario desee visualizar los equipos hidráulicos asociados a su cuenta.	
Estabilidad	Alta	
Comentarios		

RF- 10	DisplayData
Objetivos asociados	CU_10_DisplayData
Requisitos asociados	Muestra los datos asociados a un equipo hidráulico.
Descripción	Desde esta funcionalidad se presentará de forma gráfica todos los datos relacionados con un equipo hidráulico.

Precondición	El solicitante es un usuario registrado con el rol de consultor.	
Secuencia Normal	Paso	Acción
	1	Desde el menú principal disponible para el rol consultor el usuario selecciona la funcionalidad de visualización.
	2	El sistema presenta en el menú desplegable las funcionalidades de visualización de mapa y visualización de datos.
	3	El usuario selecciona la opción de visualizar datos.
	4	El sistema presenta un formulario con una lista desplegable con los equipos hidráulicos asociados a la cuenta del usuario y un botón para mostrar los datos.
	5	El usuario selecciona el equipo a consultar y pulsa el botón de mostrar datos.
	6	El sistema presenta una nueva ventana con una gráfica donde se muestran los datos asociados al equipo hidráulico y una opción de zoom para seleccionar y ampliar una parte de la gráfica para un mejor estudio por parte del usuario.
Postcondición		
Excepciones	Paso	Acción
	6	El sistema devuelve un mensaje de error indicando que no hay equipos asociados a la cuenta del usuario y vuelve a la ventana principal de su rol de consultor.
Rendimiento	Paso	Cota de tiempo
	2	3 segundos
	4	3 segundos
	6	5 segundos
Frecuencia esperada	Cada vez que el usuario desee visualizar los datos que almacena un equipo hidráulico asociado a su cuenta.	
Estabilidad	Alta	
Comentarios		

RF- 11	DisplayTreeOfAccountsAndEquipments
Objetivos asociados	CU_11_DisplayTreeOfAccountsAndEquipments
Requisitos asociados	Mostrar todas las cuentas y equipos asociados almacenados en el sistema.
Descripción	Desde esta funcionalidad el administrador puede visualizar en modo árbol todas las cuentas junto con los equipos que tienen asociados. Además al seleccionar una cuenta podrá visualizar los usuarios que disponen de acceso a dicha cuenta.

Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde el menú principal disponible para el rol administrador el usuario selecciona la funcionalidad de visualización.
	2	El sistema presenta en el menú desplegable las funcionalidad de visualizar el árbol de cuentas y equipos.
	3	El usuario selecciona la opción de visualizar el árbol de cuentas y equipos.
	4	El sistema presenta una nueva ventana con un formulario donde se muestran en modo de árbol todas las cuentas y los equipos asociados a cada una de ellas.
	5	El usuario pulsa sobre una cuenta.
	6	El sistema presenta en la misma ventana los usuarios que tienen acceso a la cuenta seleccionada por el usuario.
Postcondición	El usuario visualiza todas las cuentas y equipos hidráulicos almacenados en el sistema y tiene la posibilidad de consultar los usuarios que disponen de acceso a cada cuenta.	
Excepciones	Paso	Acción
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	2 segundos
	6	5 segundos
Frecuencia esperada	Cada vez que el usuario desee consultar las cuentas y equipos almacenados en el sistema y los usuarios que disponen de acceso a las mismas.	
Estabilidad	Alta	
Comentarios		

RF- 12	AssociateAccounts	
Objetivos asociados	CU_12.1_AssociateAccounts	
Requisitos asociados	Asociar cuenta padre e hija	
Descripción	Desde esta funcionalidad se asocian entre sí las cuentas padres e hijas.	
Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el rol de administrador el usuario selecciona en el menú principal la opción de gestionar sistema.

	2	El sistema muestra en un menú desplegable las opciones de gestión de cuentas, gestión de equipos y la gestión de asociaciones sólo disponibles para el rol de administrador.
	3	El usuario selecciona la opción de gestión de asociaciones.
	4	El sistema presenta una nueva ventana con un formulario con una lista desplegable para seleccionar una cuenta y además en otro menú en la misma ventana cuatro pestañas que hacen referencia a la cuenta padre, los equipos asociados, las cuentas hijas y los usuarios.
	5	El usuario selecciona una cuenta.
	6	El sistema muestra en la misma ventana en la pestaña de cuenta padre una lista desplegable con el valor de la cuenta padre asociada a la cuenta seleccionada por el usuario y en su defecto para que seleccione la cuenta padre a la que quiere asociar la cuenta seleccionada en primera instancia.
	7	El usuario selecciona la cuenta padre.
	8	El sistema valida la información introducida y en caso de ser correcta, la almacena y presenta la misma ventana con la información actualizada.
Postcondición	Dos cuentas quedan asociadas y se almacenan en el sistema.	
Excepciones	Paso	Acción
	8	La validación no es satisfactoria porque se ha introducido la misma cuenta padre e hija y se presenta un mensaje de error indicando el problema detectado.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	3 segundos
	6	3 segundos
	8	5 segundos
Frecuencia esperada	Cada vez que se desee asociar o desasociar una cuenta hija a una cuenta padre.	
Estabilidad	Alta	
Comentarios		

RF- 13	AssociateEquipmentToAccount
Objetivos asociados	CU_12.2_AssociateEquipmentToAccount
Requisitos asociados	Asociar equipos hidráulicos a una cuenta.
Descripción	Desde esta funcionalidad se asocian equipos hidráulicos a una cuenta.

Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el rol de administrador el usuario selecciona la opción de gestionar sistema.
	2	El sistema muestra una nueva ventana donde se presentan botones con acceso a las distintas funcionalidades que tiene el rol de administrador como son la gestión de cuentas, de equipos y de asociaciones.
	3	El usuario selecciona la opción de gestión de asociaciones.
	4	El sistema presenta una nueva ventana con un formulario con una lista desplegable para seleccionar una cuenta y además en otro menú en la misma ventana cuatro pestañas que hacen referencia a la cuenta padre, los equipos asociados, las cuentas hijas y los usuarios. Además en otro menú en la misma ventana se mostrarán los equipos y canales asociados a la cuenta hija seleccionada mediante botones checkBox que en caso de ser marcados asociarán el equipo a la cuenta y desmarcándolos desasociarán el equipo a la cuenta.
	5	El usuario selecciona una cuenta.
	6	El sistema muestra en la misma ventana en la pestaña de equipos una tabla con botones checkBox todos los equipos del sistema donde aparecerán marcados aquellos equipos que están asociados a la cuenta seleccionada y un botón asociar para que realice los ajustes pertinentes y valide los cambios.
	7	El usuario asocia y desasocia los equipos a la cuenta seleccionada y pulsa el botón de asociar.
	8	El sistema valida la información introducida y en caso de ser correcta, la almacena y presenta la misma ventana con la información actualizada.
Postcondición	El equipo hidráulico queda asociado a una cuenta y es almacenado en el sistema.	
Excepciones	Paso	Acción
	8	La validación no es satisfactoria porque se ha seleccionado un equipo que ya está asociado a la cuenta indicada y se presenta un mensaje de error que muestra el problema detectado.

Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	3 segundos
	6	3 segundos
	8	5 segundos
Frecuencia esperada	Cada vez que se desee asignar a una cuenta un equipo hidráulico sin cuenta asociada.	
Estabilidad	Alta	
Comentarios		

RF- 14	AssociateAccountToUser	
Objetivos asociados	CU_12.3_ AssociateAccountToUser	
Requisitos asociados	Asociar una cuenta a un usuario del sistema.	
Descripción	Desde esta funcionalidad se asocia una cuenta a un usuario.	
Precondición	El solicitante es un usuario registrado con el rol de administrador.	
Secuencia Normal	Paso	Acción
	1	Desde la ventana principal disponible para el rol de administrador el usuario selecciona en el menú principal la opción de gestionar sistema.
	2	El sistema muestra en un menú desplegable las opciones de gestión de cuentas, gestión de equipos y la gestión de asociaciones sólo disponibles para el rol de administrador.
	3	El usuario selecciona la opción de gestión de asociaciones.
	4	El sistema presenta una nueva ventana con un formulario con una lista desplegable para seleccionar una cuenta y además en otro menú en la misma ventana cuatro pestañas que hacen referencia a la cuenta padre, los equipos asociados, las cuentas hijas y los usuarios.
	5	El usuario selecciona una cuenta.
	6	El sistema muestra en la misma ventana en la pestaña de usuarios una tabla con botones checkBox todos los usuarios del sistema donde aparecerán marcados aquellos usuarios que están asociados a la cuenta seleccionada y un botón asociar para que realice los ajustes pertinentes y valide los cambios.
7	El usuario asocia y desasocia los usuarios a la cuenta seleccionada y pulsa el botón de asociar.	

	8	El sistema valida la información introducida y en caso de ser correcta, la almacena y presenta la misma ventana con la información actualizada.
Postcondición	El usuario asocia una cuenta a un usuario del sistema.	
Excepciones	Paso	Acción
	8	En caso de no seleccionar un usuario o una cuenta el sistema informa al usuario mediante un mensaje del problema encontrado y se sigue mostrando la misma ventana para que repita la acción.
Rendimiento	Paso	Cota de tiempo
	2	2 segundos
	4	3 segundos
	6	3 segundos
	8	5 segundos
Frecuencia esperada	Cada cuenta debe tener al menos un usuario asignado.	
Estabilidad	Alta	
Comentarios		

2.8 Capturas de las funcionalidades de la aplicación

- **CU_1_Login**

Ilustración 5. CU_1_Login

- **CU_2_Logout**



Ilustración 6. CU_2_Logout

- Ventana principal usuario consultor:

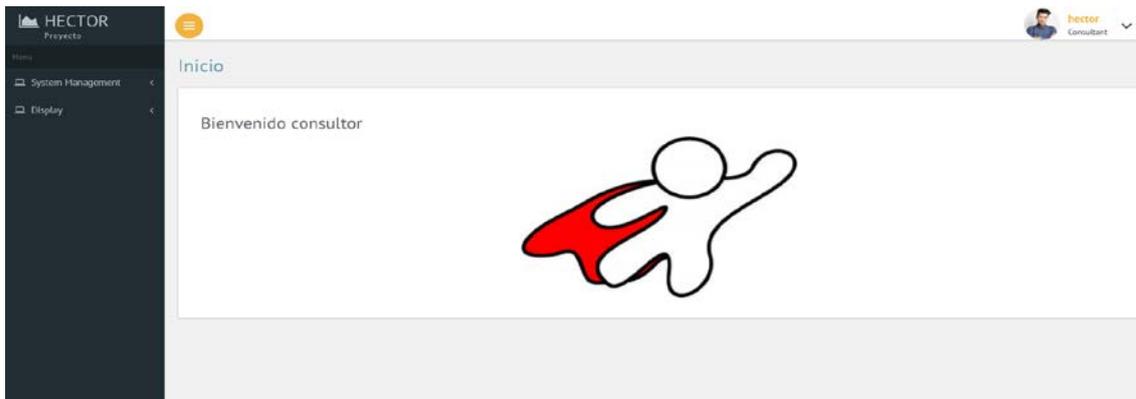


Ilustración 7. Ventana principal usuario consultor

- Ventana principal usuario administrador:



Ilustración 8. Ventana principal usuario administrador

- CU_3_RegisterUser y CU_3.4_ListadoUsuarios

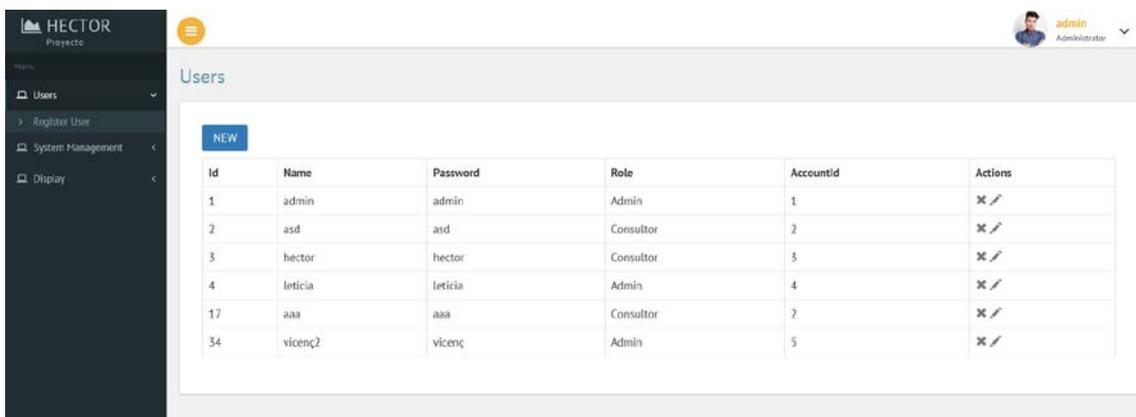


Ilustración 9. CU_3_RegisterUser y CU_3.4_ListadoUsuarios

- **CU_3.1_AltaUsuario**

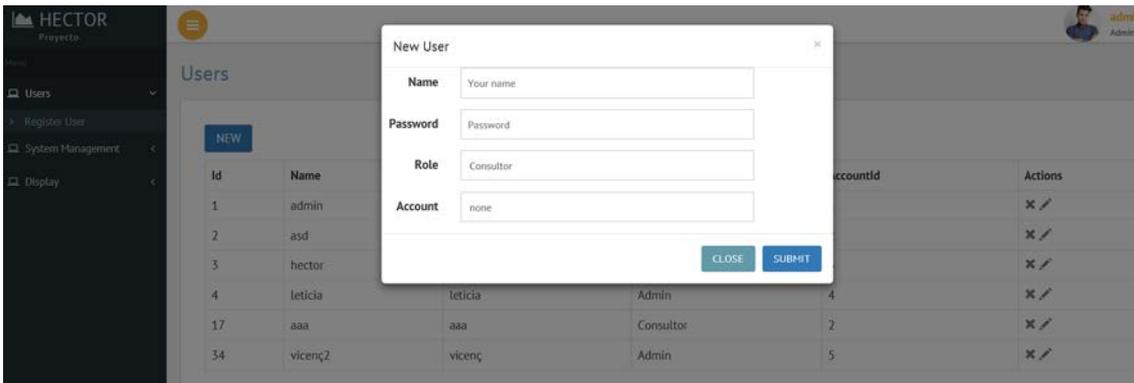


Ilustración 10. CU_3.1_AltaUsuario

- **CU_3.2_BajaUsuario**

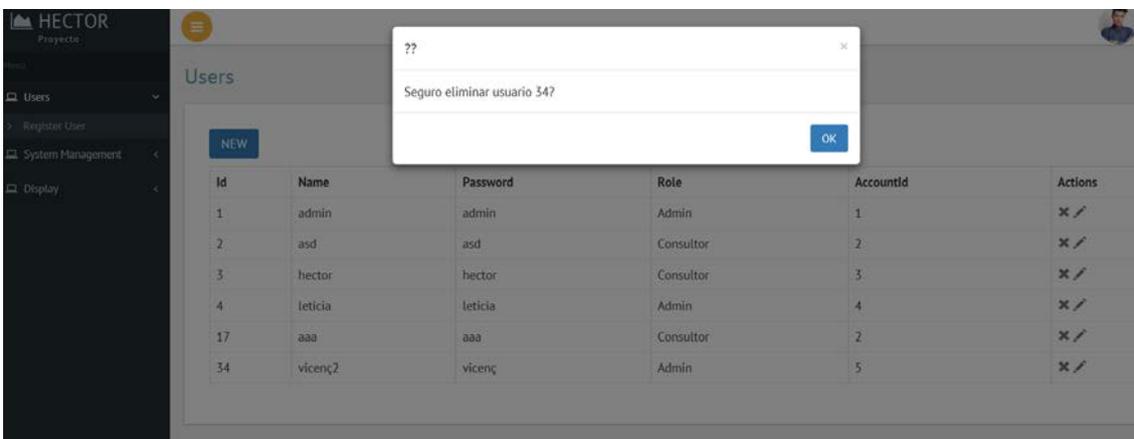


Ilustración 11. CU_3.2_BajaUsuario

- **CU_3.3_ModificarUsuario y CU_4_ChangePassword**

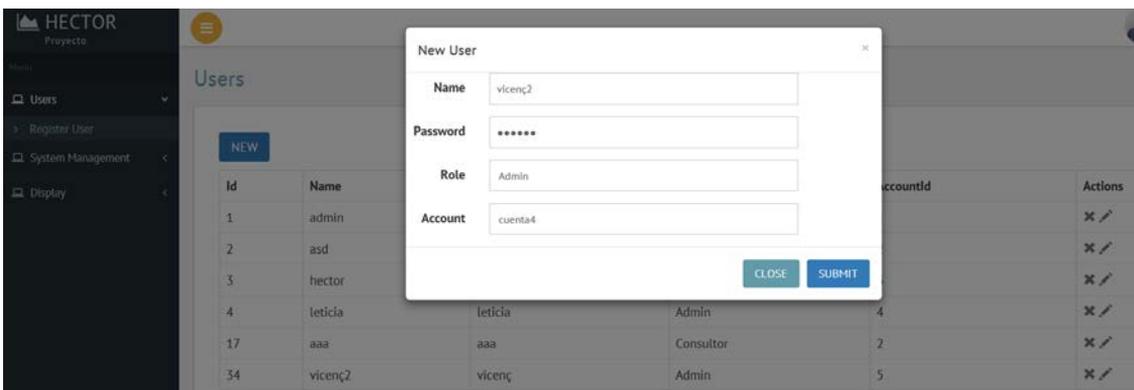


Ilustración 12. CU_3.3_ModificarUsuario y Cu_4_ChangePassword

- **CU_5_EditEquipment**

Id	Name	Latitude	Longitude	Accountid	Actions
3	equip3	40.3	-3.2	3	
4	equip4	40.3	-3.2	3	
5	equipo5	23.13	-4.23	3	
6	equipo6	36.12	5.3	3	
8	equipo8	21.2	-3.29	3	

Ilustración 13. CU_5_EditEquipment

- **CU_5.1_EditName y CU_5.2_EditCoordinates**

Id	Name	Latitude	Longitude	Accountid	Actions
3	equip3				
4	equip4				
5	equipo5				
6	equipo6	36.12	5.3	3	
8	equipo8	21.2	-3.29	3	

Edit equipment

Name:

Latitude:

Longitude:

Ilustración 14. CU_5.1_EditName y CU_5.2_EditCoordinates

- **CU_6_EditChannel**

Id	Name	Channel Number	Unit	equipmentid	Actions
7	channel0	0	m.ca	3	
8	channel1	1	l/s	3	
9	channel2	2	m.ca	3	
10	channel0	0	l/s	4	
11	channel1	1	m.ca	4	
12	channel2	2	l/s	4	
13	channel0	0	m.ca	5	
14	channel1	1	l/s	5	
15	channel2	2	m.ca	5	
16	channel0	0	m.ca	6	

Ilustración 15. CU_6_EditChannel

- CU_6.1_EditName y CU_6.2_EditEngineeringUnit

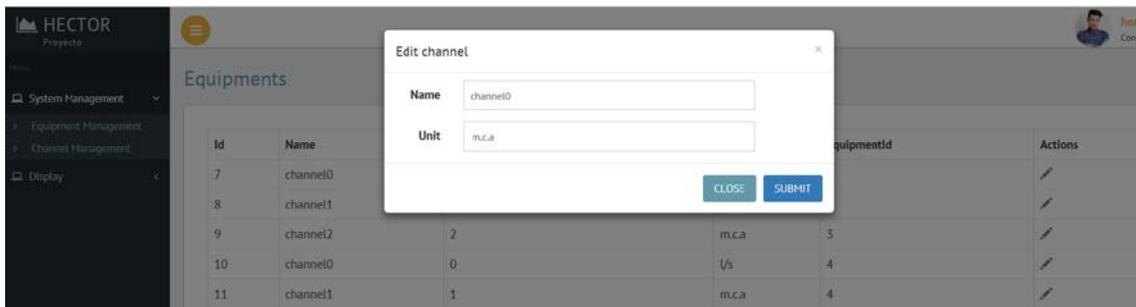


Ilustración 16. CU_6.1_EditName y CU_6.2_EditEngineerinUnit

- Ventana System Management usuario consultor:

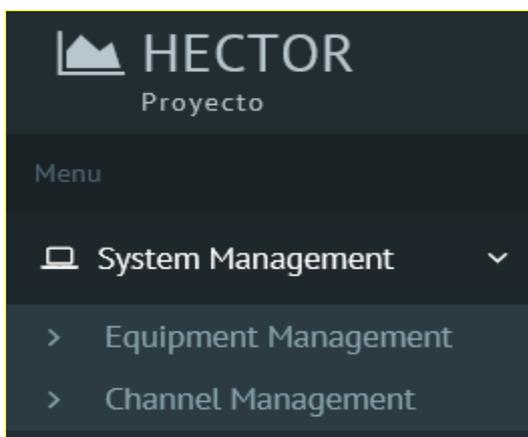


Ilustración 17. Ventana System Management usuario consultor

- Ventana System Management usuario administrador:

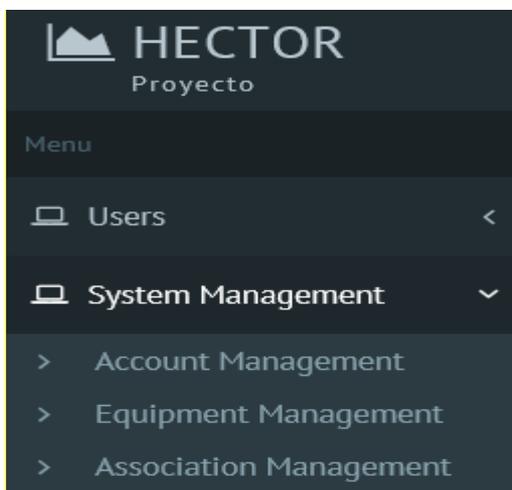


Ilustración 18. Ventana System Management usuario administrador

- **CU_7_ManagementAccounts y CU_7.4_ListAccounts**

Id	Name	TreePosition	FatherAccount	Actions
1	none	1	1	✕ ✎
2	cuenta1	1	4	✕ ✎
3	cuenta2	2	2	✕ ✎
4	cuenta3	1	1	✕ ✎
5	cuenta4	2	4	✕ ✎
6	cuenta5	2	4	✕ ✎
7	cuenta6	1	1	✕ ✎
8	cuenta7	4	10	✕ ✎
9	cuenta8	1	1	✕ ✎
10	cuenta9	3	3	✕ ✎

Ilustración 19. CU_7_ManagementAccounts y CU_7.4_ListAccounts

- **CU_7.1_AddAccount**

Ilustración 20. CU_7.1_AddAccount

- **CU_7.2_DeleteAccount**

Ilustración 21. CU_7.2_DeleteAccount

- CU_7.3_ModifyAccount

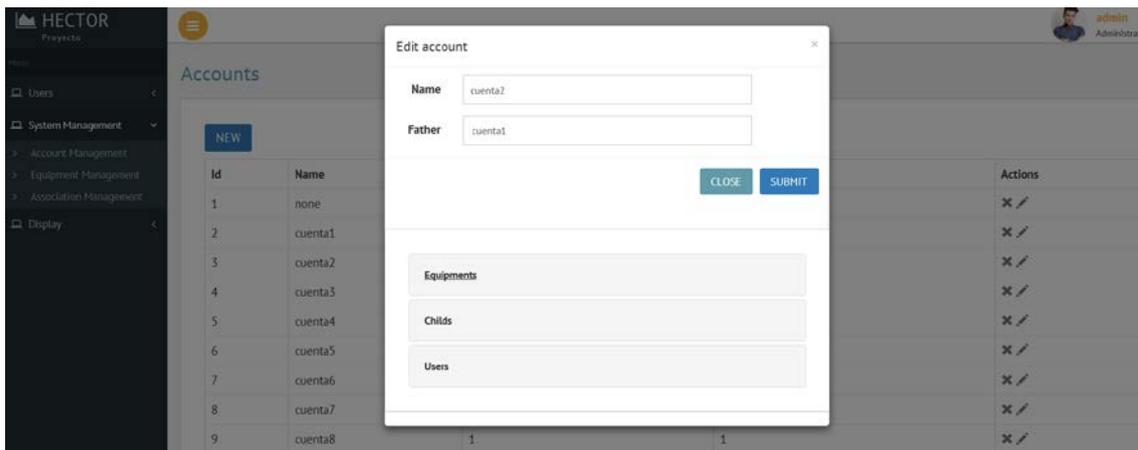


Ilustración 22. CU_7.3_ModifyAccount

- CU_8_ManagementEquipments y CU_8.4_ListEquipments

Id	Name	Latitude	Longitude	AccountId	Actions
1	equip1	40.3	-3.2	2	✕ / ✎
2	equip2	40.6091	-3.69549	2	✕ / ✎
3	equip3	40.5	-3.2	3	✕ / ✎
4	equip4	40.3	-3.2	3	✕ / ✎
5	equipo5	23.13	-4.23	3	✕ / ✎
6	equipo6	36.12	5.3	3	✕ / ✎
7	equipo7	45.12	78.32	1	✕ / ✎
8	equipo8	21.2	-3.29	3	✕ / ✎
9	equipo9	13.31	31.13	1	✕ / ✎

Ilustración 23. CU_8_ManagementEquipments y CU_8.4_ListEquipments

- CU_8.1_AddEquipment

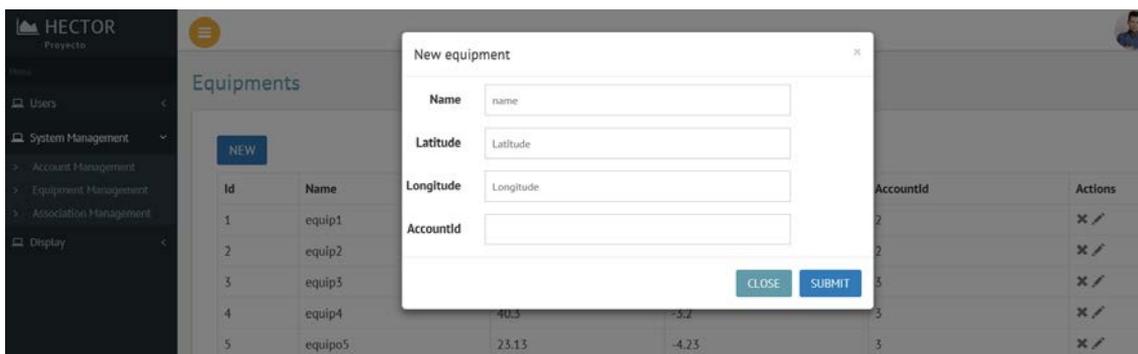


Ilustración 24. CU_8.1_AddEquipment

- CU_8.2_DeleteEquipment

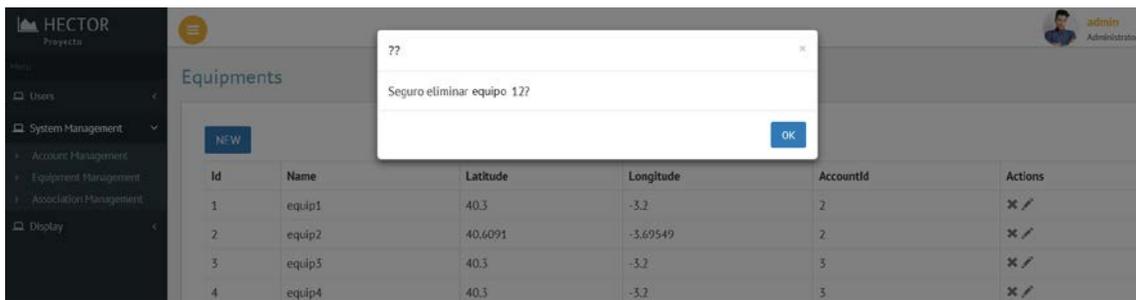


Ilustración 25. CU_8.2_DeleteEquipment

- CU_8.3_ModifyEquipment

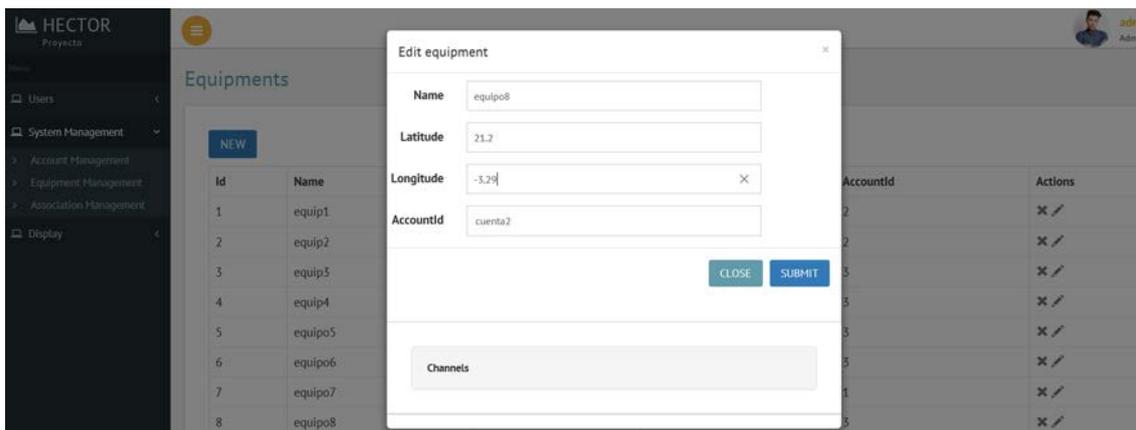


Ilustración 26. CU_8.3_ModifyEquipment

- CU_9_DisplayMap

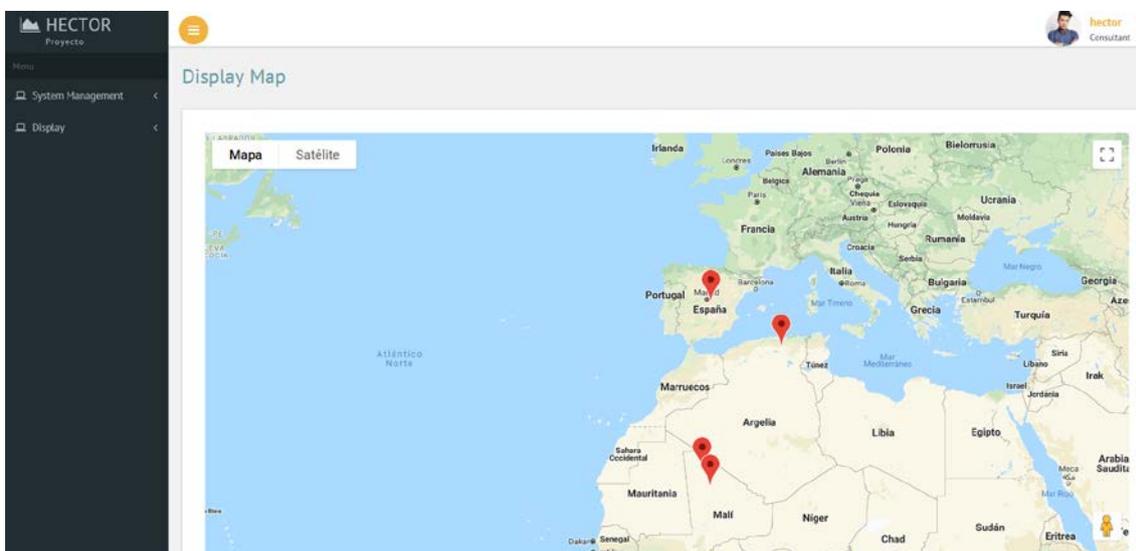


Ilustración 27. CU_9_DisplayMap

- CU_10_DisplayData

Paso1:

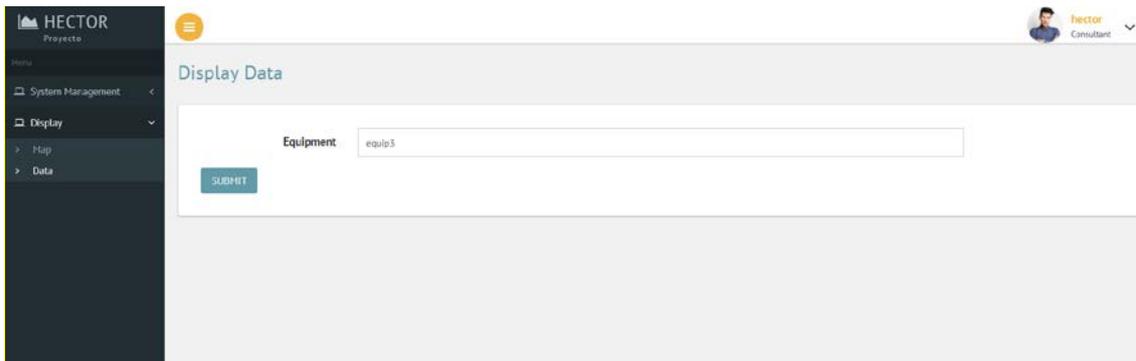


Ilustración 28. CU_10_DisplayData Paso1

Paso2:

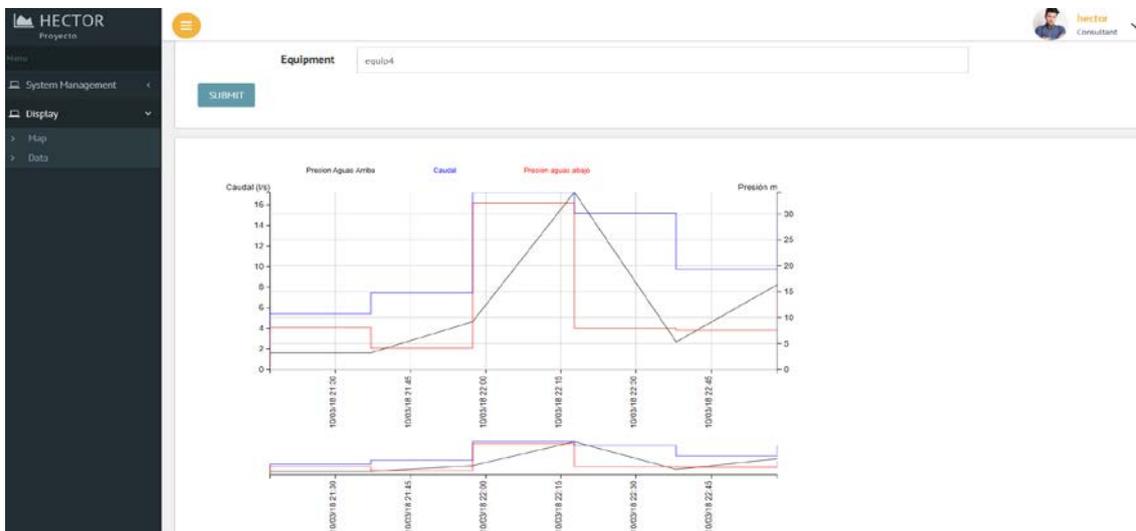


Ilustración 29. CU_10_DisplayData Paso2

- CU_11_DisplayTreeOfAccountsAndEquipments



Ilustración 30. CU_11_DisplayTreeOfAccountsAndEquipments

- **CU_12_Associations Management**

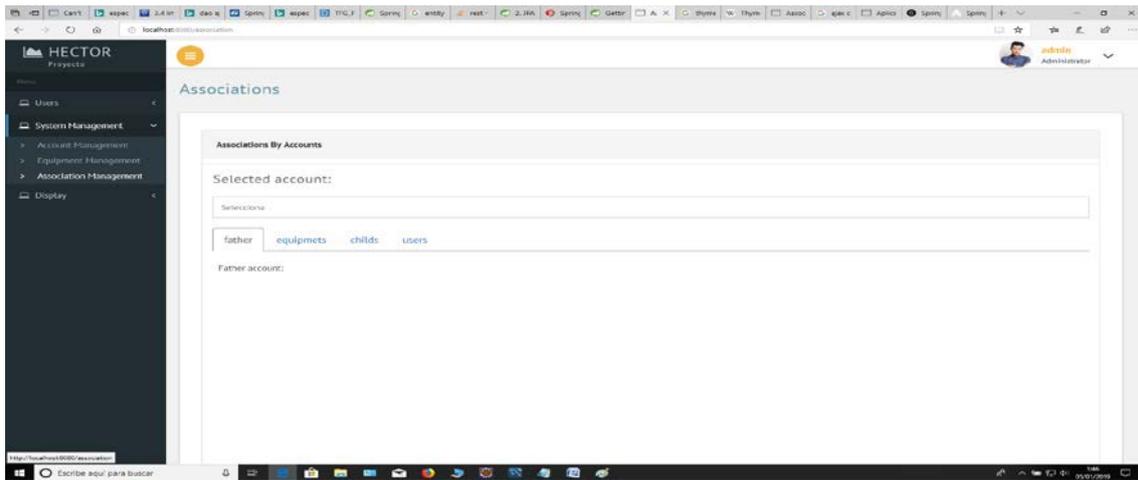


Ilustración 31. CU_12_Associations Management

- **CU_12.1_AssociateAccounts**

Paso1:

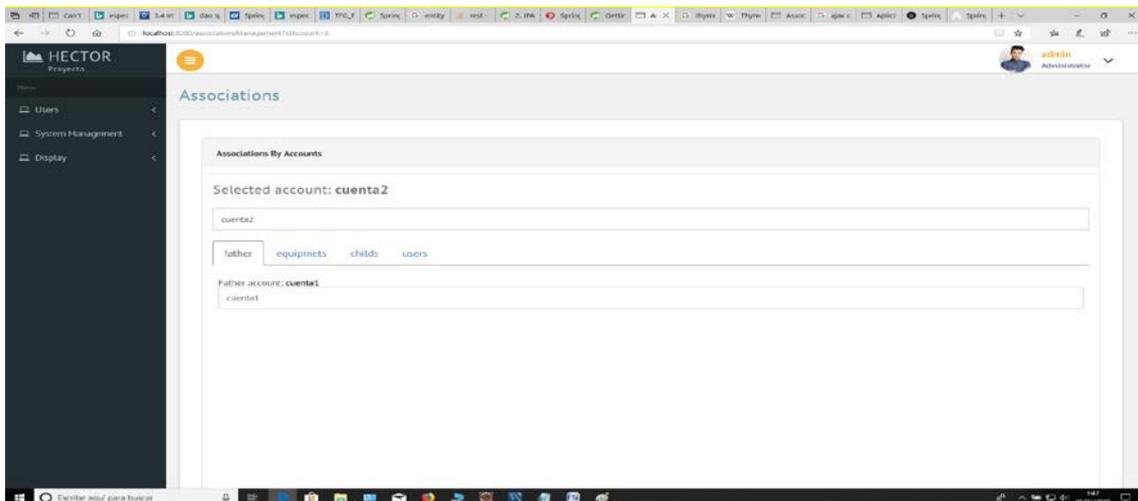


Ilustración 32. CU_12.1_AssociateAccounts Paso1

Paso2:

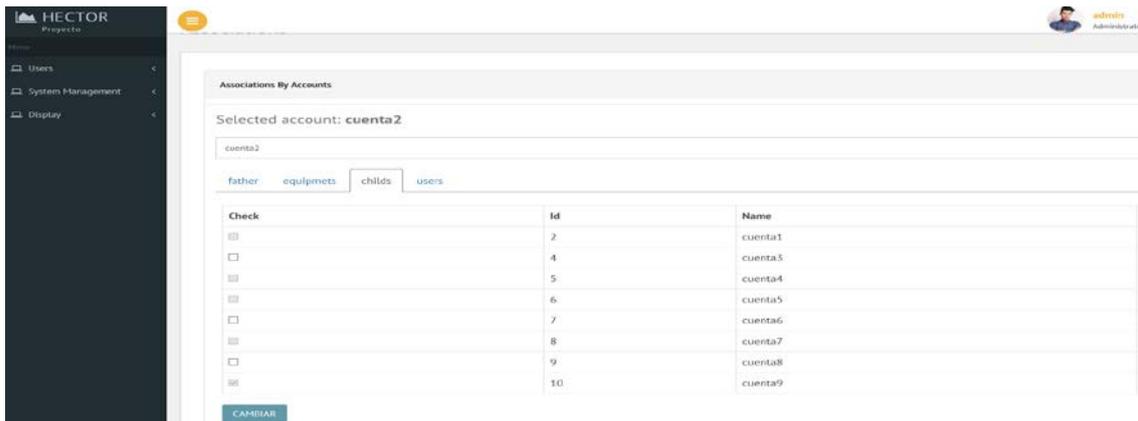


Ilustración 33. CU_12.1_AssociateAccounts Paso2

- **CU_12.2_AssociateEquipmentToAccount**

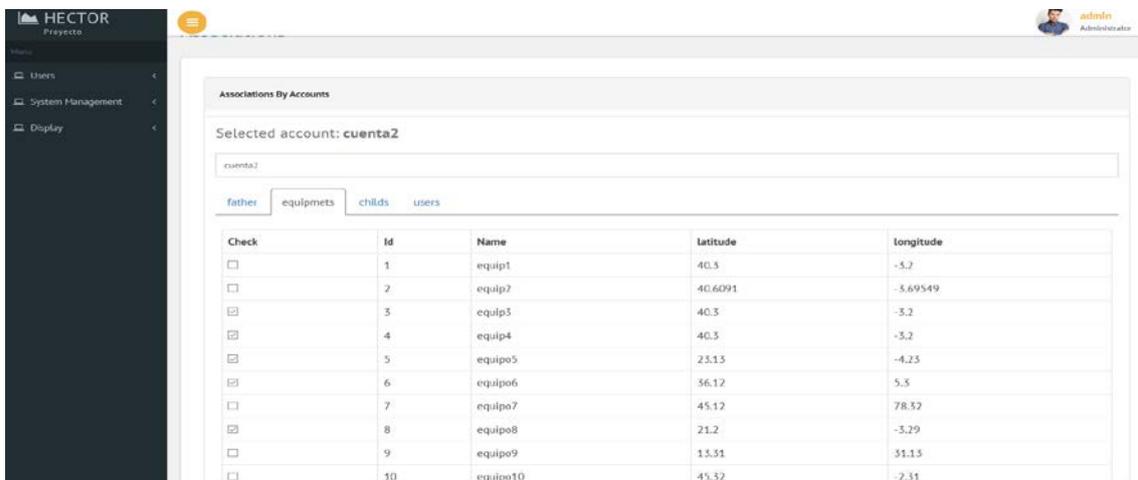


Ilustración 34. CU_12.2_AssociateEquipmentToAccount

- **CU_12.3_AssociateAccountToUser**

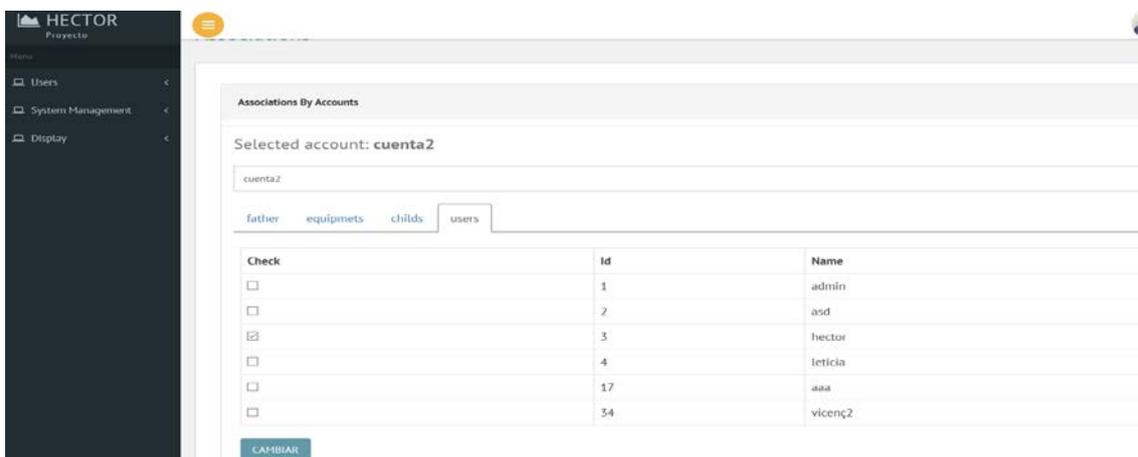


Ilustración 35. CU_12.3_AssociateAccountToUser

3. Punto de vista de la información

3.1 Esquemas invariantes

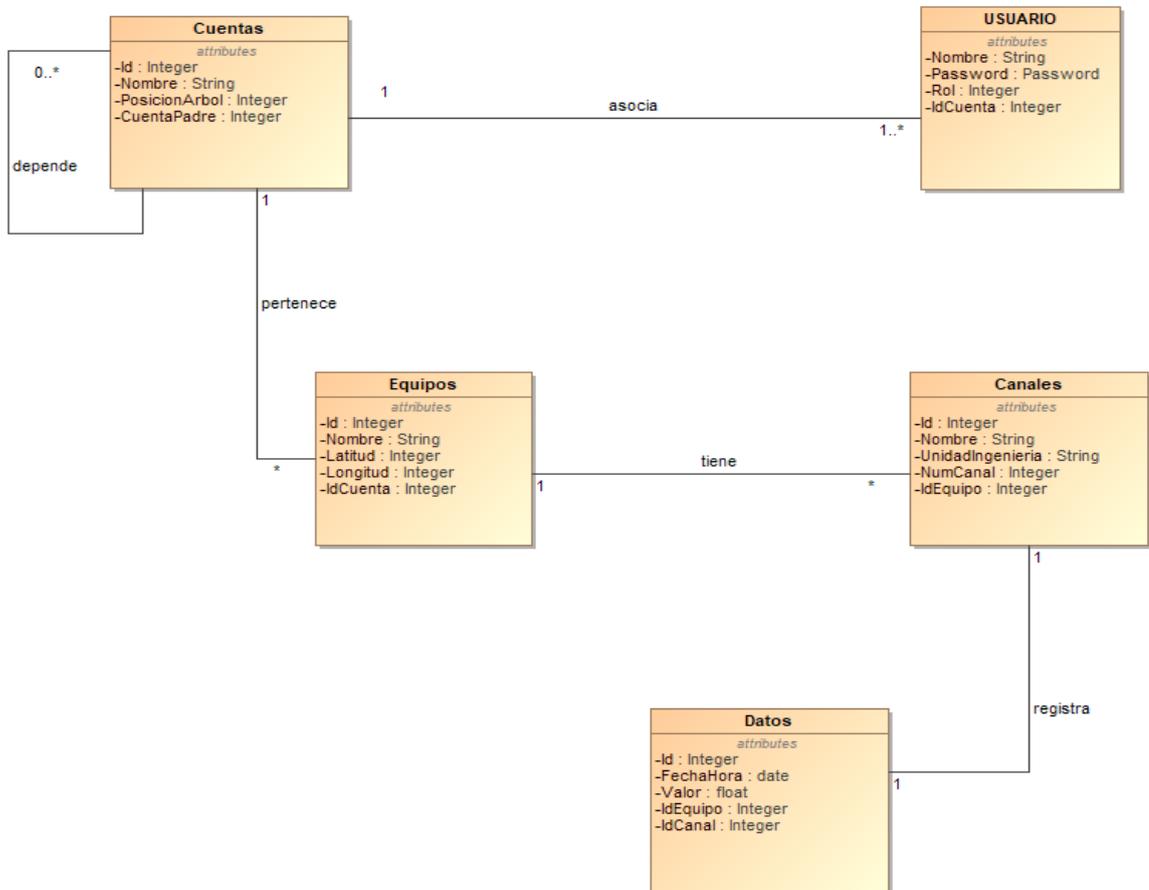


Ilustración 36. Esquemas invariantes

En este diagrama se observa que un usuario está asociado a una cuenta y que una cuenta puede depender a su vez de otra cuenta padre. Por consiguiente los equipos hidráulicos pertenecen a una cuenta y cada equipo tiene unos canales que registran datos.

3.2 Esquemas dinámicos

El objeto usuario dispone de dos roles de usuario (consultor y administrador).

En el caso del rol consultor puede interactuar con el objeto equipo modificando el nombre y las coordenadas longitud y latitud del mismo. Además,

interactúa con el objeto canal mediante la modificación del nombre y la unidad de ingeniería.

Por otro lado cuando el objeto usuario toma el rol de administrador interactúa con el objeto cuenta, el objeto equipo, el objeto canal y con otros objetos usuario puesto que puede añadir, borrar y modificar el valor de todos ellos y es el encargado de asociar las cuentas a los usuarios que tendrán acceso, los equipos hidráulicos a las cuentas a las que pertenecen y los canales que contienen los equipos.

3.3 Esquemas estáticos

El sistema en un estado inicial tendrá los objetos cuenta, equipo vacíos y el objeto canal ya tiene valores definidos. Será el objeto usuario el que sólo contendrá el usuario administrador, encargado de dar de alta las cuentas y los equipos que pertenecen a cada cuenta. Además será el encargado de asociar las cuentas a los usuarios que tendrán acceso, asociar los equipos hidráulicos a cada cuenta y asociar las cuentas entre sí.

4. Punto de vista de la computación

4.1 Diseño de alto nivel

La primera decisión para llevar a cabo el diseño de la aplicación web ha sido utilizar el Modelo-Vista-Controlador (MVC) ya que permite separar los datos, la lógica de negocio y las interfaces de usuario en tres componentes.

En primer lugar, el componente definido como Modelo será la capa encargada del tratamiento de los datos donde se llevarán a cabo los mecanismos para acceder a la información y actualizar su estado. (Como puede ser el acceso a la base de datos y su gestión).

En segundo lugar, el componente definido como Vista corresponderá al código que permitirá presentar la información que proporciona el modelo. (Como puede ser el código HTML a la hora de mostrar la salida de los datos procesados).

Y en tercer lugar, el Controlador será la capa de enlazar la vista y el modelo, mediante el envío de comandos al modelo para actualizar su estado y a la vista correspondiente para cambiar su presentación.

Bajo esta premisa se han definido ocho funcionalidades principales para dividir la aplicación, que serán: Account, Equipment, Channel, User, Home, Authentication, Management, Display y Association. Siguiendo los detalles top-down, se ha propuesto que en la capa de presentación convivan dos componentes esenciales, las vistas y los componentes ControllerWeb y ControllerRest.

Como para realizar el proyecto se ha escogido el framework Spring la aplicación del Modelo-Vista-Controlador en esta tecnología se basa en la utilización de un patrón de diseño FrontController denominado DispatcherServlet que permite disponer de un único punto para llevar a cabo la gestión de las peticiones a los distintos controladores, que para el caso de estudio serán el ControllerWeb y ControllerRest.

Por otro lado, los controladores son los responsables de preparar un modelo y determinar el nombre de la vista a mostrar aunque también pueden responder de forma directa y completar la petición si se desea. El modelo es una interfaz propia de Spring que permite abstraerse del tipo de tecnología que se está utilizando para generar las vistas y que se transforma al formato adecuado cuando es necesario. Para el desarrollo de las vistas que muestran las funcionalidades del proyecto se ha utilizado código HTML junto con un motor de plantillas llamado Thymeleaf, Javascript y estructuras JSON.

La comunicación básica de estos elementos es la que se muestra en la figura siguiente:

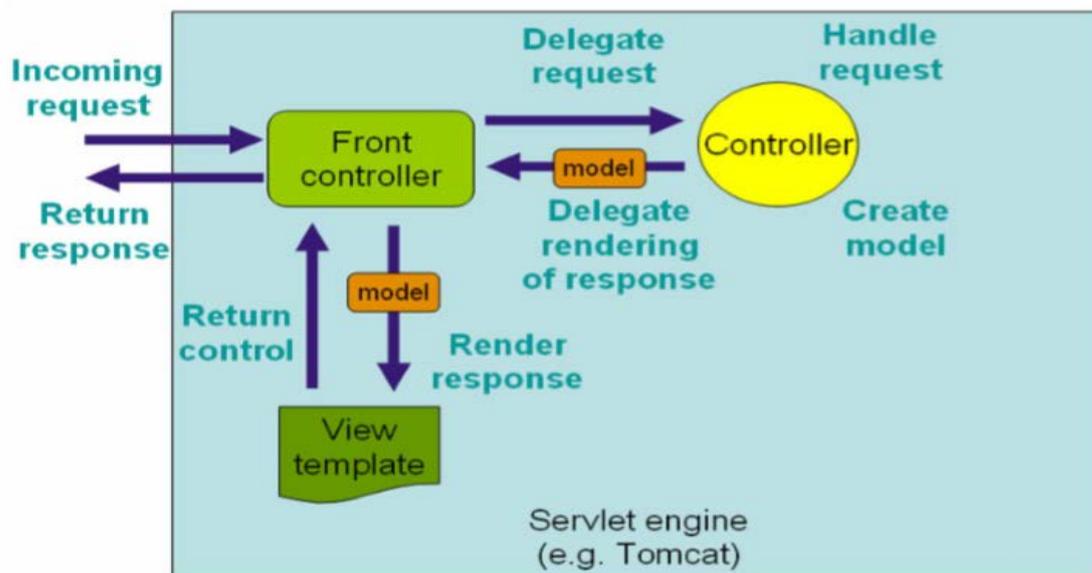


Ilustración 37. Funcionamiento Modelo-Vista-Controlador en Spring

Para la capa de negocio será el componente ControllerServices el que hará uso de las interfaces definidas en el componente ControllerDao para invocar a las clases que incluye el componente ControllerDto para recuperar la información de la capa de persistencia.

Para la capa de integración se dispondrá de una base de datos relacional cuyo acceso será local, de manera que se utilizarán componentes JPA que estarán incluidos en el componente ControllerEntities y que harán uso de las interfaces definidas en el componente ControllerRepository para llevar a cabo las operaciones. Así al aplicar el perfil de JPA tendremos múltiples entidades para cada uno de los objetos que persistan en la base de datos.

4.2 Arquitectura del software detallada

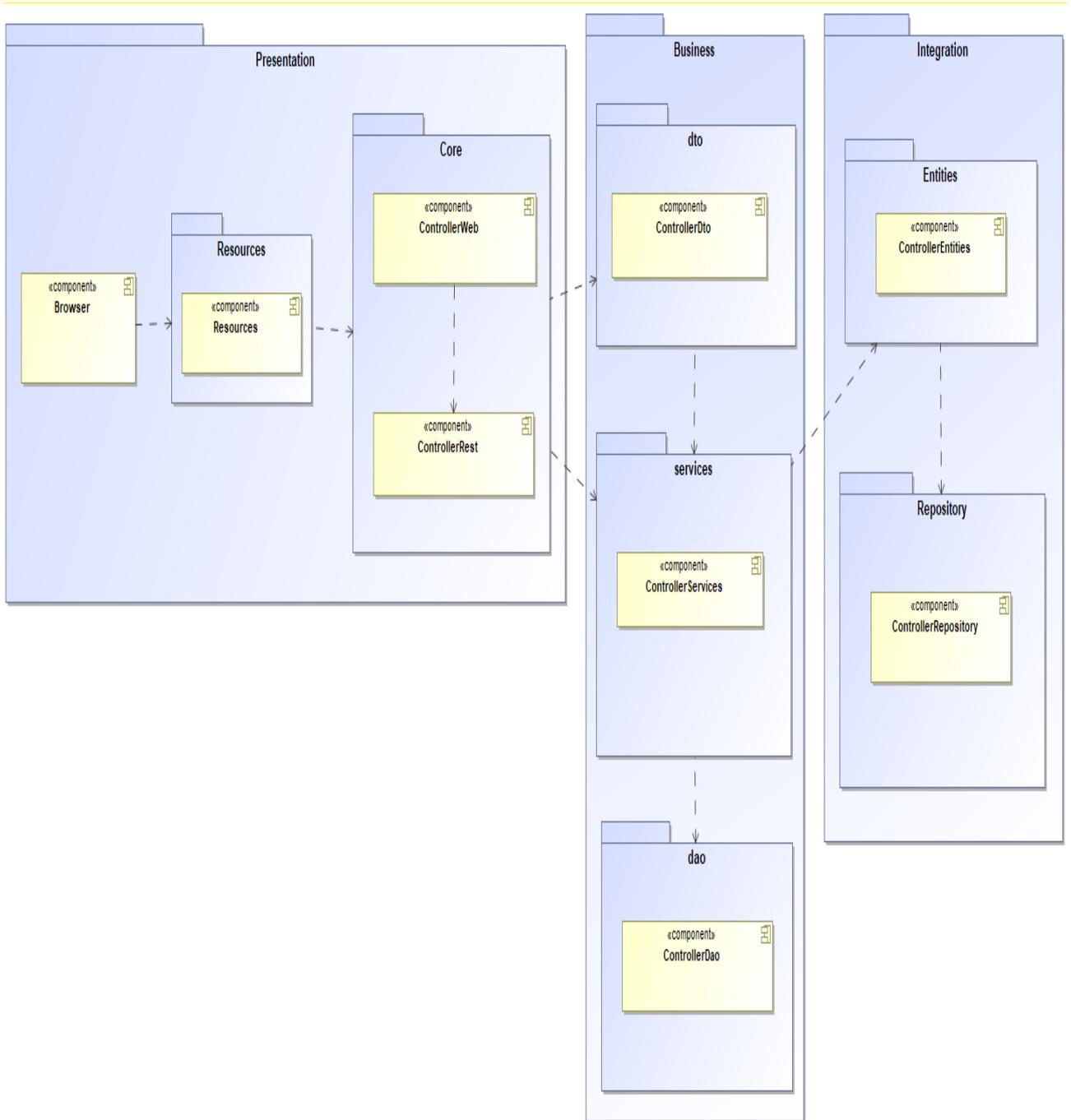


Ilustración 38. Diagrama de componentes

El diagrama anterior muestra los siguientes paquetes:

- Browser: corresponde al navegador donde se mostrarán las vistas HTML con cada funcionalidad de la aplicación para ser visualizadas por los usuarios.
- Resources: contiene las vistas HTML, los ficheros de configuración, estilos (css) y Javascript necesarios para mostrar los templates de cada funcionalidad de la aplicación web.
- Core: contiene los controladores Web y Rest que son la base de la aplicación ya que el DispatcherServlet de Spring recurrirá a ellos para cada petición que se lleve a cabo.
- Dto(Data Transfer Object): contiene las clases necesarias para transportar los datos entre procesos mediante la creación de objetos de transferencia del tipo necesitado por cada petición para llevar a cabo una funcionalidad.
- Dao(Data Access Object): contiene las interfaces necesarias para que los objetos de servicio se abstraigan de la implementación requerida para acceder a los datos y habilitar un acceso transparente a los mismos.
- Services: contiene los componentes que encapsulan la lógica de negocio ya que se encargan de realizar las acciones solicitadas por los controladores.
- Entities: contiene las clases JPA para el manejo de las entidades definidas en la base de datos.
- Repository: contiene las interfaces necesarias para que los objetos de entidad se abstraigan de la implementación requerida para llevar a cabo las operaciones de consulta, búsqueda, actualización, borrado y almacenamiento.

A continuación se muestra la arquitectura de cada una de las capas definidas con un mayor grado de especificación.

Capa de Presentación: Primer Nivel

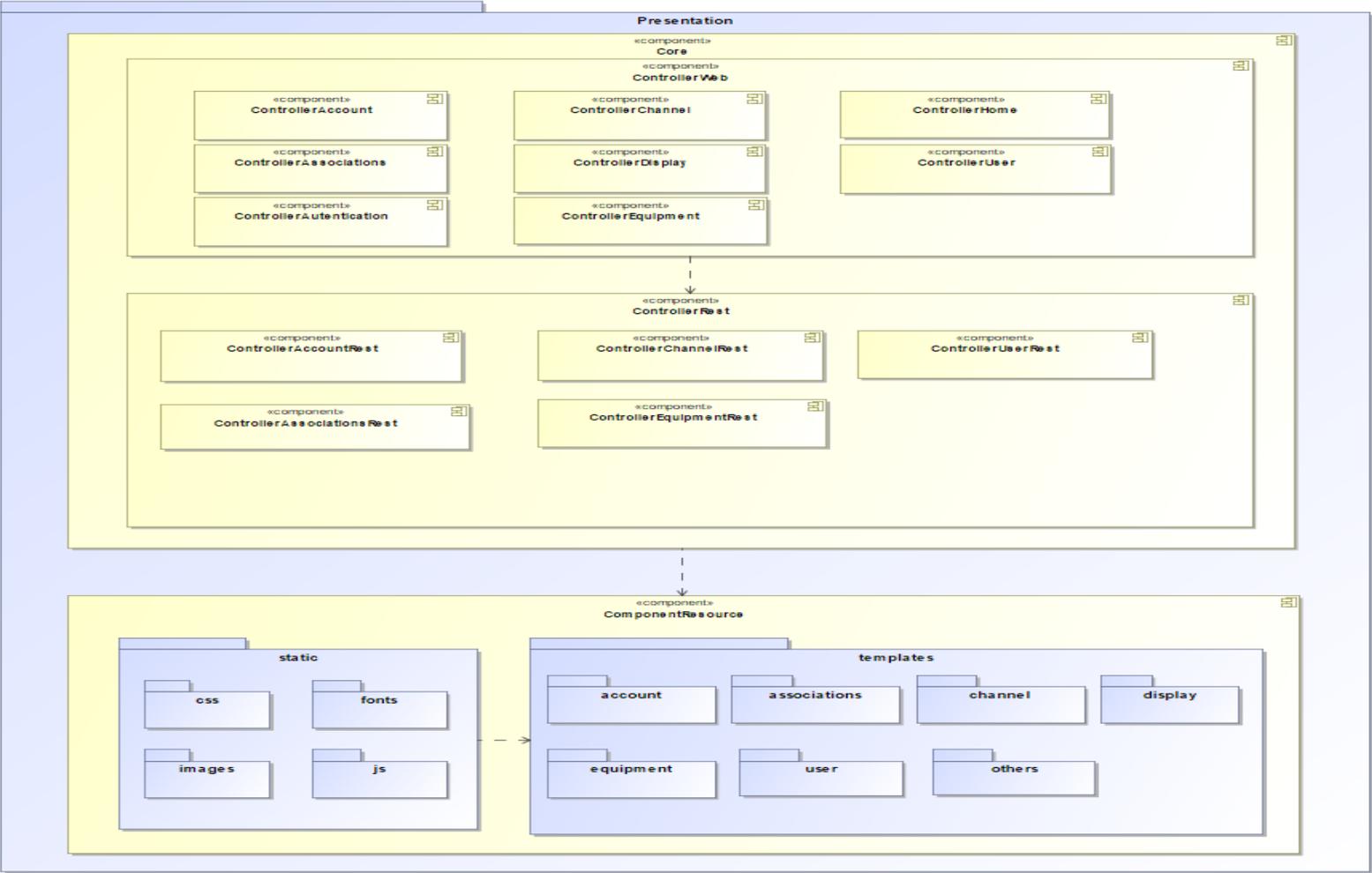


Ilustración 39. Capa de Presentación Primer Nivel

Capa de Presentación: Segundo Nivel

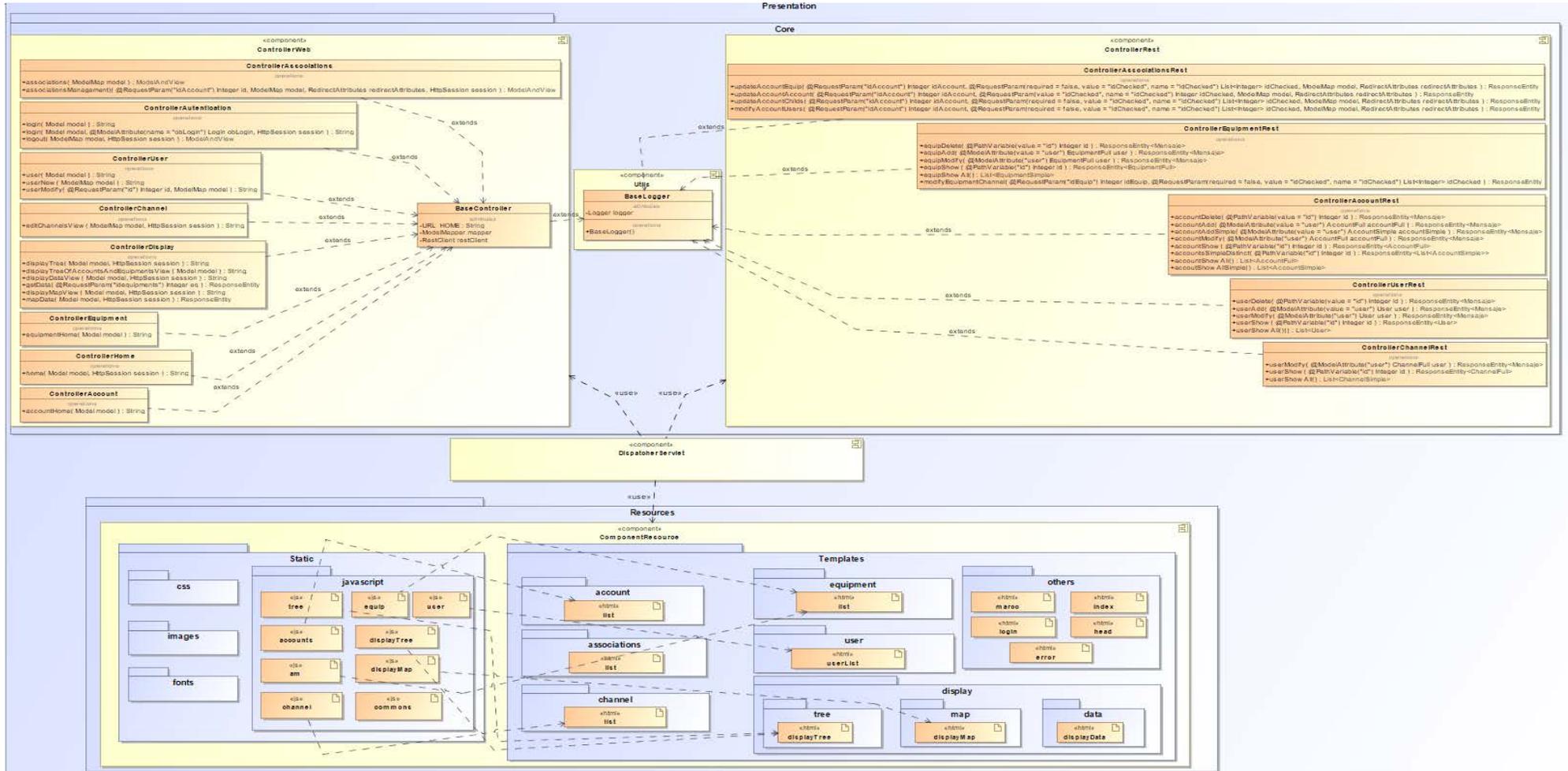


Ilustración 40. Capa de Presentación Segundo Nivel

Capa de Negocio:

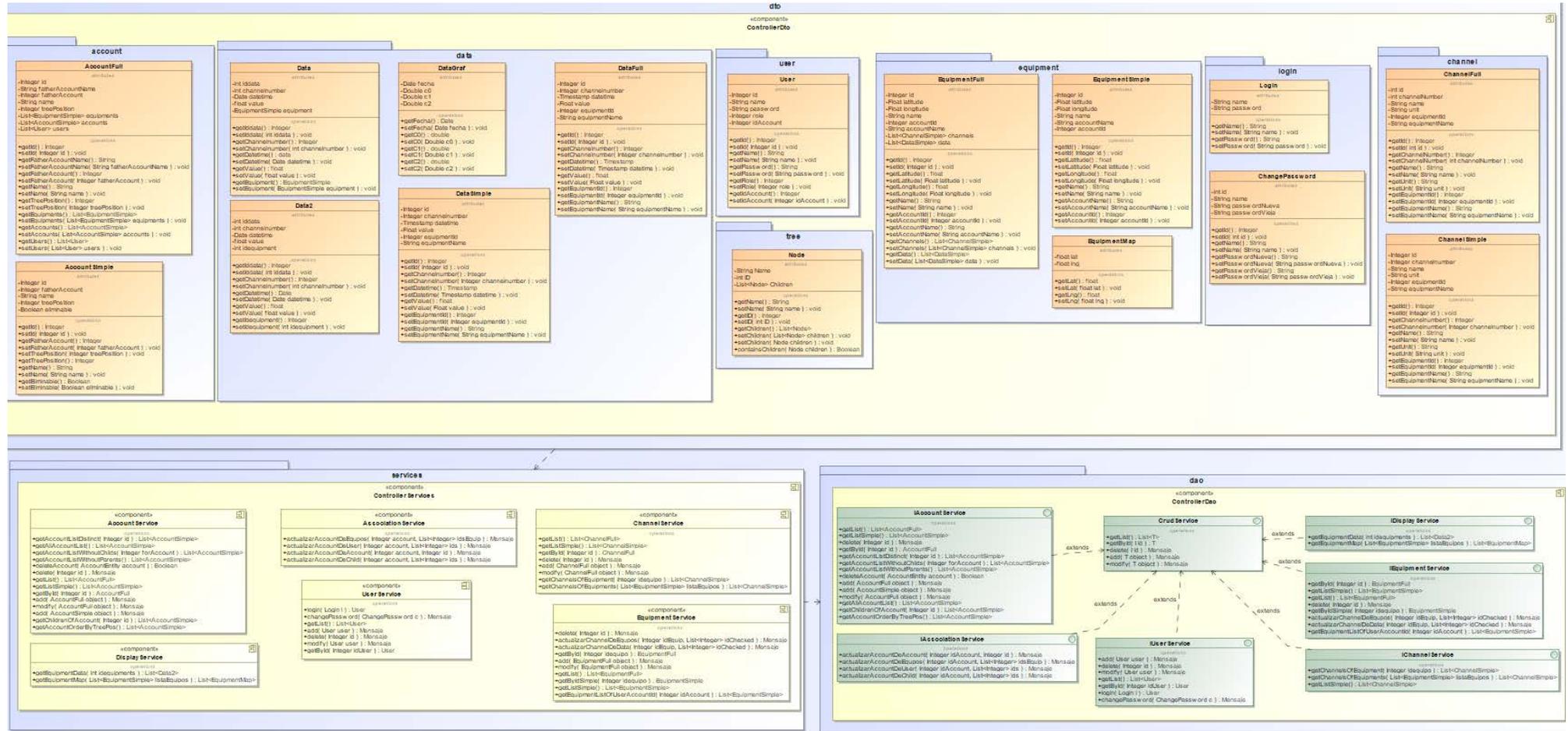


Ilustración 41. Capa de negocio

Capa de Integración:

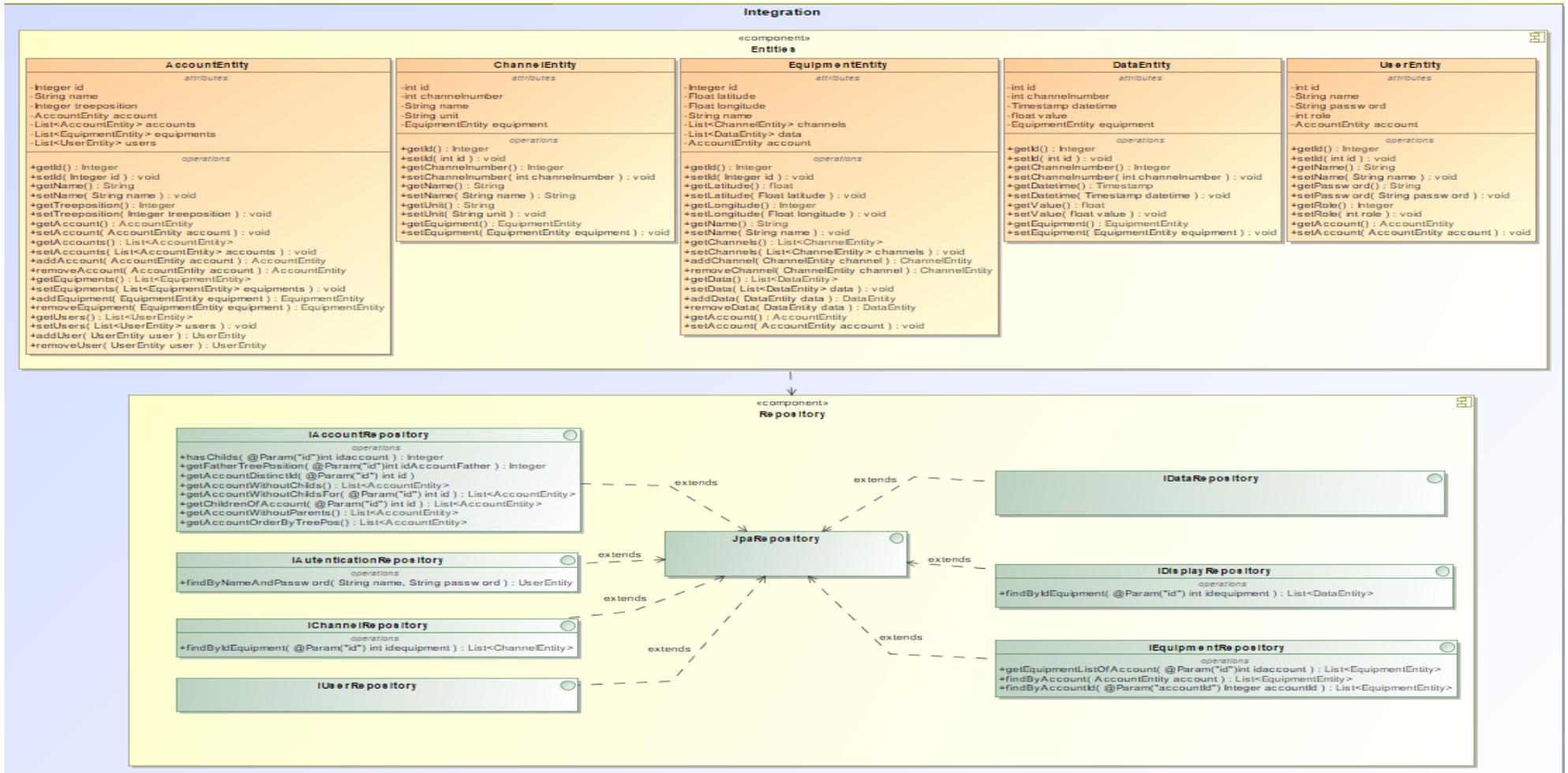


Ilustración 42. Capa de integración

5. Punto de vista de la ingeniería

Los usuarios que harán uso de la aplicación web utilizarán ordenadores personales y accederán al aplicativo vía URL que se aloja en el servidor que será el ordenador central.

Los datos persistentes serán almacenados en una base de datos relacional MySQL.

Según el perfil JEE aplicado los componentes de la capa de presentación serán ejecutados entre los ordenadores de los usuarios y el servidor. Como la curva de aprendizaje y uso del framework Spring la he llevado a cabo mientras realizaba el proyecto me he estado informando sobre la forma de trabajar con esta tecnología en el ámbito laboral. De tal manera, los componentes que se encargan de gestionar las vistas de la capa de presentación serán los componentes incluidos en el controlador `ControllerWeb`(`ControllerAccount`, `ControllerAssociations`, `ControllerAutentication`, `ControllerChannel`, `ControllerDisplay`, `ControllerEquipment`, `ControllerHome` y `ControllerUser`) y en el controlador llamado `ControllerRest`(`ControllerAccountRest`, `ControllerAssociationsRest`, `ControllerChannelRest`, `ControllerEquipmentRest` y `ControllerUserRest`). La diferencia entre ambos radica en que en el primer caso los datos como el listado de usuarios, cuentas, equipos y asociaciones son obtenidos a través de un cliente Rest y en el segundo caso las funcionalidades de alta, baja, y modificación de usuarios, cuentas, equipos y asociaciones se lleva a cabo mediante peticiones Ajax que devuelven la información en formato JSON en el cuerpo de la respuestas Http.

Por otro lado, los componentes incluidos en el controlador denominado como `ControllerServices`(`AccountService`, `AssociationService`, `ChannelService`, `DisplayService`, `EquipmentService` y `UserService`) junto con el controlador que incluye las interfaces que implementan dichos servicios llamado `ControllerDao`(`IAccountService`, `IAssociationService`, `IChannelService`, `IDisplayService`, `IEquipmentService`, `IUserService` y `CRUD`) y las clases definidas en el controlador `ControllerDto`(`AccountFull`, `AccountSimple`, `ChannelFull`, `ChannelSimple`, `Data`, `Data2`, `DataFull`, `DataGraf`, `DataSimple`, `EquipmentFull`, `EquipmentMap`, `EquipmentSimple`, `ChangePassword`, `Login`, `Node` y `User`) conforman la capa de negocio y serán ejecutados en el servidor de forma concurrente, lo que quiere decir que cada componente actuará como un proceso independiente del resto. Su finalidad será el obtener los datos de la capa de persistencia para servirlos a la capa de presentación.

El acceso a los componentes de integración por parte de los componentes de negocio será local por razones de eficiencia, de esta manera el componente identificado como `Entities` hace uso de Java Persistence Api(JPA) que proporciona una forma de manipular la base de datos a través de objetos Entity y permite mapear clases Java contra las tablas de la base de datos mediante anotaciones. Por otro lado el componente `Repository` contiene las interfaces necesarias para llevar a cabo las operaciones que se desea

realizar con cada tipo de entidad donde todas ellas heredan los métodos de la interfaz JPAREpository que define para cada caso la clase de entidad con la que se va a trabajar y el tipo de dato del campo identidad. En conclusión estos dos componentes serán los encargados del acceso a la base de datos para cada funcionalidad de la aplicación.

6. Punto de vista de la tecnología

La definición de requisitos en la tecnología a utilizar para desarrollar la aplicación web serán los siguientes:

Los clientes que hagan uso de la aplicación web podrán hacerlo desde cualquier tipo de ordenador (PC, MAC) y desde los sistemas operativos Windows y Linux.

Las comunicaciones entre el cliente y el servidor se harán por Internet y la invocación e interacción del cliente con la aplicación se hará mediante un navegador web.

Para realizar el proyecto se ha decidido utilizar el framework Spring de código libre que es un framework del lenguaje de programación Java cuya finalidad es la de facilitar el desarrollo de aplicaciones web ya que mediante ficheros xml o anotaciones se encarga de construir todos los objetos que la aplicación necesita. Spring es un framework modular en la que cada programador agrega los módulos que necesita en función de la aplicación a desarrollar.

Con el fin de abstraernos de la configuración necesaria para llevar a cabo el proyecto bajo esta tecnología se ha utilizado Spring Boot que es una herramienta que se centra en la configuración e integración de las dependencias que se suelen utilizar para poder comenzar con el desarrollo de la aplicación sin esfuerzo y facilita el despliegue de la aplicación ya que integra (configura de forma embebida) un servidor Tomcat o Jetty.

Como el IDE de desarrollo escogido es Eclipse ha sido necesario descargar el complemento Spring Tools 4 del marketplace para poder desarrollar el proyecto en Spring.

Para la conexión y explotación de la base de datos relacional MySQL se va a utilizar un componente JDBC que se define como dependencia en el archivo pom.xml de Spring.

Para el aplicativo web se empleará HTML5 y JavaEE junto con un motor de plantillas llamado Thymeleaf, también se utilizará un componente de licencia BSD D3.js basado en Javascript para la presentación gráfica y se hará uso de la Api de Google Maps para mostrar los equipos hidráulicos dados de alta en una cuenta en un mapa. Como también se desea que el usuario administrador

disponga de una visualización en modo árbol, de las cuentas y de los equipos se usará el componente Javascript TreeNode.js.

Para devolver la información desde la base de datos al componente Javascript se utilizará el formato de intercambio de datos JSON la cual es facilitada por framework Spring seleccionado ya que nos abstrae de la creación de un servicio web en “la parte servidora” para ofrecer los datos al cliente mediante el uso de RestComponents. Este mismo formato se utilizará para modificar la base de datos y su gestión por parte del usuario supervisor.

Aunque en principio la aplicación se ha planificado para ser explotada por un cliente web también se plantea utilizar el Framework Cordova para dar acceso a otro tipo de cliente como puede ser un cliente Android.

7. Conclusiones

En este capítulo se dan a conocer las conclusiones obtenidas al realizar este proyecto, lo aprendido durante su desarrollo, la experiencia y aspectos valiosos adquiridos durante su realización.

En primer lugar la utilización de un framework, en concreto Spring, para desarrollar una aplicación puede proporcionar mejoras y beneficios ya que aporta grandes facilidades en todos los procesos de ingeniería del software como son el llevar a cabo una correcta estructuración del sistema, el análisis, diseño, codificación, la realización de pruebas y su mantenimiento.

En segundo lugar he observado que utilizar el módulo Spring MVC para llevar a cabo el proyecto brinda gran variedad de funcionalidades a las aplicaciones web pudiendo destacar la capacidad para la reutilización de código, mediante el uso de anotaciones facilita la implementación proporcionando un mejor diseño y sobre todo hacer hincapié en su alto grado de modularidad. Este framework cuenta con una gran variedad de módulos que se pueden agregar dependiendo de las necesidades del usuario y de la aplicación web a desarrollar. En este aspecto he podido crear un proyecto Maven a través de Spring Boot con todas las dependencias que necesitaba y al mismo tiempo, mediante el módulo Spring Data, simplificar la persistencia de datos contra los distintos repositorios de información de forma muy cómoda y rápida.

En tercer lugar he comprobado que al usar este framework puedo integrar el software desarrollado con otras herramientas enfocadas cada una en un área en particular, en lugar de tener que seleccionar una herramienta nueva para realizar tareas en la cual muchas otras tienen más experiencia y tiempo en el mercado.

Al inicio del proyecto planteé realizar la aplicación utilizando EJB de manera que, para la capa de presentación se utilizarían JSF (Java Servlet Faces) que serían gestionadas por el controlador Faces Servlet, las acciones serían los Commands definidos en los ManagedBeans y las vistas serían implementadas mediante Facelets. Para la capa de negocio decidí utilizar EJB de sesión sin estado y en la capa de integración el uso de componentes JPA que serían desplegados en un servidor de aplicaciones JBOSS pues soporta el uso de contenedores EJB 3.X. Sin embargo, durante la fase de implementación por diversos problemas de compatibilidad con el gestor de base de datos que en un principio quería utilizar (MySQL Server), al usar la última versión de EJB comprobé que gran parte de los métodos que conocía estaban obsoletos (deprecated) y cuya gestión era distinta, unido a que en el mercado actual se considera el uso de esta tecnología casi un atraso, decidí actualizarme aprovechando que el proyecto a desarrollar era JavaEE, e investigar y aprender a desarrollar con las últimas tecnologías que han aparecido en el mercado y que se exigen a nivel laboral por lo que opté por escoger el framework de Spring.

Aunque disponía de poco tiempo para realizar el proyecto y la curva de aprendizaje de esta tecnología no es fácil sobre todo teniendo en cuenta las funcionalidades que quería aportar a la aplicación web, considero que he conseguido realizar casi todas las funcionalidades que había propuesto al planificar el proyecto. Sin embargo, no he logrado obtener el funcionamiento deseado a la hora de mostrar la información en modo de árbol de las cuentas y los equipos desde el rol de administrador ya que muestro el árbol de cuentas pero no los equipos que dispone cada cuenta. Por otro lado, no he podido incorporar el uso del framework Cordova para dar acceso a otro tipo de cliente como puede ser un cliente Android.

A medida que he ido profundizando en esta nueva forma de implementar he caído en la cuenta de que necesitaría agregar más módulos de Spring a mi desarrollo como por ejemplo Spring Security que proporciona un amplio conjunto de funcionalidades para llevar a cabo la autorización destacando la autorización mediante solicitudes web, la autorización basada en los métodos que pueden ser invocados por un rol de usuario determinado y la autorización de acceso a instancias de objetos que pertenecen a un dominio determinado, pero no he llegado a incorporarlo al proyecto al no lograr que funcione como es debido. A nivel de componente no he tenido dificultades para implementarlo pero el problema ha surgido cuando tenía que validar en el objeto sesión el rol del usuario logado en el sistema, a la hora de presentar las funcionalidades disponibles para dicho rol desde las páginas HTML que utilizaban el motor de plantillas Thymeleaf y no conseguí que lo reconocieran.

8. Glosario

Modelo-Vista-Controlador (MVC): es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones para lo que se propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador.

HyperTextMarkup Language (HTML): es un estándar a cargo del World Wide Web Consortium(W3C) que hace referencia al lenguaje de marcado para la elaboración de páginas web definiendo su estructura básica y un código para la definición de contenido como texto, imágenes, vídeos, etc.. en una página web.

Thymeleaf: es una librería Java que implementa un motor de plantillas de XML/XHTML/HTML5 que se acopla muy bien en la capa vista del modelo Modelo-Vista-Controlador(MVC).

Spring: es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.

Javascript Object Notation (JSON): es un formato de texto ligero para el intercambio de datos.

Java Persistence API (JPA): es la API de persistencia desarrollada para la plataforma Java que permite establecer una correlación entre una base de datos relacional y un sistema orientado a objetos.

Data Access Object (DAO): es un componente de software que facilita el acceso a datos almacenados por ejemplo en una base de datos.

Data Transfer Object(DTO): es un objeto que transporta datos entre procesos que se utiliza para reducir el número de llamadas entre cliente y servidor al transferir los datos.

Licencia BSD(Berkeley Software Distribution): es una licencia de software libre permisiva que permite el uso de código fuente en software no libre.

Enterprise JavaBeans (EJB): son unas interfaces de programación de aplicaciones (API) que forman parte del estándar de construcción de aplicaciones empresariales J2EE de Oracle Corporation.

Java Database Connectivity (JDBC): es la especificación JavaSoft de una interfaz de programación de aplicaciones(API) estándar que permite que los programas Java accedan a sistemas de gestión de bases de datos.

Java Servlet Faces (JSF): es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.

JBOSS: es un servidor de aplicaciones JavaEE de código abierto que puede ser utilizado en cualquier sistema operativo que tenga disponible la máquina virtual de Java.

Representational State Transfer (REST): es cualquier interfaz entre sistemas que use HTTP para obtener o generar operaciones sobre datos en todos los formatos posibles, como son XML y JSON.

9. Bibliografía

Web:

https://docs.spring.io/spring/docs/current/spring-framework-reference/	10/11/2018
https://www.thymeleaf.org/documentation.html	15/11/2018
https://www.adictosaltrabajo.com/2016/12/22/aplicaciones-web-con-spring-boot-capa-a-cap/	16/11/2018
https://docs.spring.io/spring-data/jpa/docs/current/reference/html/	17/11/2018
https://www.journaldev.com/14476/spring-mvc-example	18/11/2018
https://medium.com/@gustavo.ponce.ch/spring-boot-spring-mvc-spring-security-mysql-a5d8545d837d	20/11/2018
https://d3js.org/	03/12/2018
https://cloud.google.com/maps-platform/maps/?hl=es	06/12/2018